On Multiscale and Statistical Numerical Methods for PDEs and Inverse Problems

Thesis by
Yifan Chen

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2023
Defended May 17, 2023

Yifan Chen
ORCID: 0000-0001-5494-4435

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my advisors, Thomas Hou, Houman Owhadi, and Andrew Stuart, for their unwavering support and guidance. They have consistently provided me with the freedom to explore and collaborate, allowing me to develop a versatile understanding of applied and computational mathematics and teaching me invaluable life attitudes and lessons. As someone who may overthink and get stuck in self-consciousness, I have gained much inspiration from their integrity, patience, and compassion. Working with Tom, Houman, and Andrew helped me grow professionally and personally, and I am very thankful for their intellectual generosity and humanistic care.

I thank Peter Schröder for serving as my committee chair. I vividly recall attending his discrete geometry classes during my first year, where the beauty of PDEs in graphics was refreshing. I also want to thank all the professors with whom I had the opportunity to interact both at and outside Caltech. In particular, I am grateful to Franca Hoffmann and Joel Tropp for providing me with the teaching assistant opportunity and guidance, sharing their professional and scientific insights, and supporting my career development. I have gained much knowledge from the discussions and cherish these interactions.

I want to thank the fantastic administrative staff in the CMS department and Caltech, especially Diana Bohler, Jolene Brink, and Maria Lopez. They have taken good care of me, and their help has made my graduate life much smoother.

I have had the opportunity to work with many fabulous collaborators, without whom this thesis would not have been possible. In particular, I would like to thank Pau Batlle, Ethan Epperly, Bamdad Hosseini, Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, Florian Schäfer, Yixuan Wang, and Robert Webber for the stimulating discussions and reliable support throughout our projects. I especially would like to acknowledge Florian Schäfer for teaching me Julia programming and sharing many fun ideas in computational math, and Daniel Zhengyu Huang and Yixuan Wang for tremendous freestyle and imaginative discussions about research and life.

I want to thank all the funding resources that support my research, notably the Kortchak scholarship, NSF, and MURI programs. Throughout my Ph.D., I also had the chance to experience industrial research. I thank Pengchuan Zhang for offering

me the opportunity to work on deep learning with Microsoft remotely. Additionally, my summer internship at Citadel Securities exposed me to numerous new ideas and practical research techniques, for which I am grateful.

I am immensely thankful for all my wonderful friends, with whom I have shared countless happy moments and persevered through difficult times. The camaraderie with Tom, Houman, Andrew's group members, and the CMS community has always been a great source of comfort and enthusiasm. I thank Jiajie Chen, De Huang, and Shumao Zhang for helping me navigate life in the U.S. during my early Ph.D., and I appreciate my peers Robert Hsin-Yuan Huang, Matthew Levine, Nicholas Nelsen, Yousuf Soliman, Ziyun Zhang, and many others, with whom I enjoyed plenty of intriguing discussions. I am grateful to many people in Houman's group, notably Pau Batlle and Matthieu Darcy, for organizing interesting group events and helping me socialize and get involved in the community. I would also like to thank all my Chinese cohorts at Caltech for many fun activities and chats that enriched my graduate life. Among them, I am particularly grateful to my roommates, Ruizhi Cao and Tianzhe Zheng, for their companionship during the challenges of the pandemic. I also thank Professor Butian Zhang from Tsinghua University for taking the time to chat with me about philosophy and caring about my personal growth.

Finally, I want to thank my family, especially my parents, Yuansheng Chen, and Maozhu Lei, for their unconditional love and care. They have always supported my pursuits and have given me the best freedom. Without their constant understanding, I would not be who I am today. Thank you, Dad and Mom.

ABSTRACT

This thesis is about *numerical methods* for scientific computing and scientific machine learning, with a focus on solving partial differential equations (PDEs) and inverse problems (IPs). The design of numerical algorithms usually encompasses a spectrum that ranges from *specialization to generality*. Classical approaches based on finite element methods (FEMs) and contemporary scientific machine learning approaches based on neural networks (NNs) can be viewed as lying at relatively opposite ends of this spectrum. In this thesis, we address mathematical challenges associated with both ends of the spectrum through advancing rigorous *multiscale and statistical numerical methods*.

In the first part of the thesis, we study multiscale methods for solving challenging PDEs with rough coefficients and high frequency, where standard FEMs often fail. The key is to construct specialized basis functions to incorporate the microscale structures of the equation. We present the following two contributions in Chapters II and III:

1. We introduce the *exponentially convergent* multiscale finite element method (ExpMsFEM) for solving general elliptic and Helmholtz's equations, which achieves nearly exponentially convergent accuracy regarding the number of basis functions. Notably, ExpMsFEM applies to Helmholtz's equations. Compared to pre-existing approaches, ExpMsFEM does not rely on any partition of unity functions and can lead to more localized basis functions with better "orthogonality" properties.

2. We analyze a multiscale method that reveals mathematical connections between numerical coarse-graining of elliptic PDEs and scattered data approximation. We introduce a new concept of subsampled lengthscale into the coarse variables, and highlight a novel tradeoff between efficiency and accuracy induced by the choice of this lengthscale.

The second part of the thesis explores the interplay between numerical approximation and statistical inference for algorithm design, with a primary focus on *automation of algorithms*, as advocated in the scientific machine learning paradigm. More specifically, we investigate using Gaussian processes (GPs) and kernel methods to build a simple yet flexible and more interpretable (than NNs) computational

pipeline to automate solving general PDEs/IPs. In Chapters IV, V, and VI, we make the following three contributions regarding methodology, efficiency, and adaptivity:

1. We introduce GPs and kernel methods to solve general *nonlinear* PDEs and IPs. By assigning a GP prior to the unknown functions and observing PDE information at certain collocation points, we perform Bayes inference to learn the solution. The method combines the theoretical rigor of traditional numerical algorithms with the flexible design of machine learning solvers, generalizing radial basis function-based approaches.

2. In GP and kernel-based methodology, the presence of dense kernel matrices can limit scalability. In the case of PDE problems, these matrices may also involve partial derivatives of the kernel. Fast algorithms for such matrices are less developed compared to the derivative-free counterparts. We present a *sparse Cholesky factorization* algorithm based on the near-sparsity of the Cholesky factor under a *multiscale reordering* of Diracs and derivative measurements. The algorithm is motivated by a *probabilistic interpretation* of factorization. We rigorously analyze the exponentially convergent accuracy of the algorithm. This enables us to compute approximate inverse Cholesky factors of kernel matrices with a state-of-the-art *near-linear complexity* in space and time.

3. We analyze the use of *hierarchical learning* for enhancing the expressivity and adaptivity of GPs and kernel methods. We investigate two paradigms for learning hierarchical parameters: one from a *probabilistic perspective*, using the empirical Bayes approach, and the other from an *approximation-theoretic view*, using the kernel flow algorithm. We prove the consistency of these paradigms in the large data limit and identify their implicit bias in parameter learning for a Matérn-like model. Our results also highlight the better robustness of approximation-theoretic-based approaches compared to Bayesian approaches in model misspecification.

Finally, in Chapter VII, we discuss other related and prospective works concerning *fast randomized numerical linear algebra* for Gaussian processes/kernel methods in high dimensional scientific problems and *gradient flow based sampling algorithms* for uncertainty quantification, thus enriching rigorous and efficient numerical methods that are based on ideas in probability and statistics.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] Pau Batlle, Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. Error analysis of kernel/GP methods for nonlinear and parametric PDEs. *arXiv preprint arXiv:2305.04962*, 2023.
Y.C. contributed to part of the theoretical analysis, numerical experiments, and writing of this work.

[2] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponentially convergent multiscale finite element method. *Communications on Applied Mathematics and Computation*:1–17, 2023. URL: https://link.springer.com/article/10.1007/s42967-023-00260-2.
Y.C. contributed to the conceptualization and writing of this review paper. Y.C. and Y.W. are joint main contributors to this work.

[3] Yifan Chen, Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Gradient flows for sampling: mean-field models, Gaussian approximations and affine invariance. *arXiv preprint arXiv:2302.11024*, 2023.
Y.C. contributed to the conceptualization, theoretical analysis, and writing of this work. Y.C. is one of the main contributors to this work.

[4] Yifan Chen, Houman Owhadi, and Florian Schäfer. Sparse Cholesky factorization for solving nonlinear PDEs via Gaussian processes. *arXiv preprint arXiv:2304.01294*, 2023.
Y.C. is the main contributor to this work, with the help of F.S. on Julia programming and comprehension of existing algorithms in this field.

[5] Yifan Chen, Ethan N Epperly, Joel A Tropp, and Robert J Webber. Randomly pivoted Cholesky: practical approximation of a kernel matrix with few entry evaluations. *arXiv preprint arXiv:2207.06503*, 2022.
Y.C. contributed to part of the formal analysis and numerical experiments of this work.

[6] Yifan Chen and Thomas Y Hou. Multiscale elliptic PDE upscaling and function approximation via subsampled data. *Multiscale Modeling & Simulation*, 20(1):188–219, 2022. DOI: https://doi.org/10.1137/20M1372214. URL: https://epubs.siam.org/doi/10.1137/20M1372214.
Y.C. is the main contributor to this work.

[7] Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M. Stuart. Solving and learning nonlinear PDEs with Gaussian processes. *Journal of Computational Physics*, 447:110668, 2021. DOI: https://doi.org/10.1016/j.jcp.2021.110668.
Y.C. contributed to all the numerical experiments, and part of the theoretical analysis and writing of this work.

[8] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponential convergence for multiscale linear elliptic PDEs via adaptive edge basis functions. *Multiscale Modeling & Simulation*, 19(2):980–1010, 2021. DOI: https://doi.org/10.1137/20m1352922. URL: https://epubs.siam.org/doi/10.1137/20M1352922.
Y.C. contributed to the conceptualization, theoretical analysis, and writing of this work. Y.C. and Y.W. are joint main contributors to this work.

[9] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponentially convergent multiscale methods for 2d high frequency heterogeneous Helmholtz equations. *To appear in Multiscale Modeling & Simulation, arXiv preprint arXiv:2105.04080*, 2021.
Y.C. contributed to the conceptualization, theoretical analysis, and writing of this work. Y.C. and Y.W. are joint main contributors to this work.

[10] Yifan Chen, Houman Owhadi, and Andrew M Stuart. Consistency of empirical Bayes and kernel flow for hierarchical parameter estimation. *Mathematics of Computation*, 2021. URL: https://www.ams.org/journals/mcom/2021-90-332/S0025-5718-2021-03649-2/.
Y.C. is the main contributor to this work.

[11] Yifan Chen and Thomas Y Hou. Function approximation via the subsampled Poincaré inequality. *Discrete and Continuous Dynamical Systems-A*, 2020. DOI: http://dx.doi.org/10.3934/dcds.2020296. URL: https://www.aimsciences.org/article/doi/10.3934/dcds.2020296.
Y.C. is the main contributor to this work.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Number *Page*

Chapter 1

INTRODUCTION

This thesis is about designing *numerical algorithms* for scientific computing and scientific machine learning, with a particular emphasis on solving partial differential equations (PDEs) and inverse problems (IPs).

A shared challenge in these fields is the numerical approximation of infinite-dimensional objects using finite-dimensional information. In scientific computing, we compute functions that are solutions of continuous PDEs using a finite number of arithmetic operations. In scientific machine learning, we learn and infer a continuous function given a finite amount of data. These challenges are typically addressed through building suitable finite dimensional approximation space of the function, such as by finite element methods (FEMs) in traditional scientific computing and neural networks (NNs) in contemporary machine learning.

When designing numerical approximations, there is a fundamental concern about whether the primary focus should be on *generality or specialization*. Such a focus can result in algorithms with different properties, which, in turn, can lead to diverse paths for developing the algorithms further. FEMs and NNs can be seen as lying at relatively opposite ends of the generality-versus-specialty spectrum. There are challenges associated with both ends. For instance, FEMs may not be specialized enough for problems with rough coefficients and high frequency, as they can perform arbitrarily poorly [17, 19]. NNs, on the other hand, may sometimes be too general for low-dimensional PDEs and IPs that we often encounter, as the existing training tricks are enormous [150, 282, 284, 64, 299] and relevant theories are elusive.

In this thesis, we investigate both ends of the spectrum between generality and specialization, making progress in developing numerical algorithms with enhanced accuracy, efficiency, and robustness. To that end, we study *multiscale and statistical numerical methods*, addressing their associated mathematical challenges. Specifically, we advance exponentially convergent multiscale algorithms for solving challenging PDEs with rough coefficients or high frequency, while developing general yet rigorous and interpretable Gaussian processes and kernel methods for automatic learning of solutions of nonlinear PDEs and IPs. This chapter will discuss multiscale methods in Section 1.1 and statistical numerical methods in Section 1.2.

1.1 Multiscale Numerical Methods

Mathematically, solving PDEs is a function approximation problem. All numerical methods aim to find a finite-dimensional approximation of the solution. Multiscale methods are designed to take advantage of the microscale structure of the equation to construct the approximation space. Such consideration is beneficial because the equation's structure plays a critical role in determining the coarse and fine-scale behaviors of the solution, which, in turn, impacts the accuracy of the approximation. In fact, without taking these structures into account, standard numerical methods such as FEMs struggle to address many challenging problems in rough media [17] and high-frequency wave propagation [19].

Approximating a function involves finding a coarse scale representation of the function, which can be approached from either a *primal or dual perspective*. The former focuses on coarse function spaces, while the latter concerns coarse measurements or variables that belong to the dual space of the function space. The two perspectives lead to different paths of incorporating the microscale information[1].

This thesis contributes to progress in both perspectives. For the former, we aim at coarse spaces that achieve exponentially convergent approximation accuracy even when the solution is rough. Methods with such convergence rates have been pioneered in the work of optimal basis [15] for elliptic equations using local spectral basis and partition of unity functions in an overlapped domain decomposition; see also some recent developments in [253, 35, 41, 16, 243, 178, 179]. We generalize the analysis to *high-frequency Helmholtz's equations* and provide an alternative multiscale framework based on *non-overlapped domain decompositions* that lead to more localized basis functions. In the latter perspective, exemplified by methods such as Local Orthogonal Decomposition (LOD) [181] and Gamblets [203], the selected coarse variables resemble the sampled data in scattered data approximation and machine learning. Building on this connection, we contribute to a line of analysis that characterizes the *accuracy-efficiency tradeoff* exhibited by a new concept of *subsampled lengthscale*, which we introduce into these coarse variables.

In Sections 1.1.1 and 1.1.2, we provide the necessary background on the prototypical equations and how solving PDEs can be viewed as a function approximation problem through Galerkin's methods. In Sections 1.1.3 and 1.1.4, we discuss, at a high level,

[1] We categorize multiscale methods based on primal and dual perspectives, which may be unconventional in the literature. Nevertheless, we found this view to be convenient and insightful; see our detailed discussions in Sections 1.1.3 and 1.1.4. The readers can also refer to [5] for a comprehensive review of history of multiscale methods.

the context and our contributions to the primal and dual perspectives. Section 1.1.5 provides a summary of these discussions.

1.1.1 Prototypical Equations

To provide clear explanations of multiscale methods, we consider a model problem in a bounded domain $\Omega \subset \mathbb{R}^d$ with a Lipschitz boundary. The equation is

$$-\nabla \cdot (A\nabla u) + Vu = f\,. \tag{1.1.1}$$

We ignore boundary conditions for simplicity. Here, A, V are functions in $L^\infty(\Omega)$; they can be rough, leading to oscillations in ∇u and resulting in a difficult-to-solve solution u. *We do not assume any scale separation or periodicity in A, V.*

We assume $f \in L^2(\Omega), 0 < A_{\min} \leq A(x) \leq A_{\max} < \infty$. When $V = 0$, the equation reduces to a standard elliptic equation. On the other hand, if $Vu = -k^2 u$, the boundary conditions are suitably chosen, and u is a complex-valued, we obtain the Helmholtz equation with wavenumber k.

1.1.2 Solving PDEs as Function Approximation

One common approach for solving (1.1.1) is to use Galerkin's method, which involves selecting a finite-dimensional space S of basis functions and combining it with the weak form of the equation [32]. The high-level theory of Galerkin's method implies that once functions in S can approximate the solution well, such that

$$\eta(S) := \sup_{f \in L^2(\Omega)} \inf_{v \in S} \frac{\|u - v\|_{\mathcal{H}(\Omega)}}{\|f\|_{L^2(\Omega)}} \quad \text{is small}, \tag{1.1.2}$$

where u, f satisfies (1.1.1) and $\|\cdot\|_{\mathcal{H}(\Omega)}$ is the energy norm:

$$\|w\|^2_{\mathcal{H}(\Omega)} := (A\nabla w, \nabla w)_\Omega + (|V|w, w)_\Omega\,, \tag{1.1.3}$$

then the Galerkin solution will have quasi-optimality, meaning that its error will be of the same order as the optimal approximation accuracy $\inf_{v \in S} \|u - v\|_{\mathcal{H}(\Omega)}$, which is small thanks to (1.1.2).

The failure of many FEMs in elliptic equations with rough coefficients [17] and high-frequency Helmholtz's equations [19] is due to a large $\eta(S)$ caused either by the roughness of the solution or the indefiniteness of the equation. Multiscale methods aim to find better S that can capture the coarse scale behavior of the solution by incorporating the structure of the equation.

1.1.3 Primal Perspective: Constructing Coarse Spaces

Multiscale methods classified into the primal perspective focus on directly building the coarse approximation space for u. Since localized basis functions are more favorable for the sake of computation, these approaches often involve dividing the domain Ω into smaller subdomains and constructing local approximation spaces that accurately capture the solution behavior within each subdomain. These local spaces are then coupled to form a global approximation space S. Many multiscale methods, including the Generalized Finite Element Method (GFEM) [18], the Multiscale Finite Element Method (MsFEM) [129, 130, 73], and the Generalized Multiscale Finite Element Methods (GMsFEM) [74], can be interpreted in this way. Initially studied in periodic media with a scale separation, these methods were later generalized to handle the case of rough coefficients; see [128, 52, 160, 89] and also the approach based on harmonic coordinates [209]. Most existing results concern approximation accuracy of order comparable to the size of the subdomains; the convergence rate is at most algebraic.

We discuss related results on achieving more remarkable exponential convergence in Section 1.1.3.1 and then present the key of our contribution in Section 1.1.3.2. More details are in Chapter II.

1.1.3.1 On Exponential Convergence of Accuracy

Conceptually, the most desirable approximation for a function may be the one that leads to exponentially convergent accuracy. This is possible when the function is smooth, and we use polynomial basis functions for approximations, thanks to the Taylor expansion. However, smoothness may not be necessary for such fast convergence. Babuška and Lipton showed in [15] that nearly exponential convergence can be achieved for A-harmonic functions, even when $A \in L^\infty(\Omega)$.

Exponential convergence for approximating A-harmonic functions Consider two concentric cubes $\omega \subset \omega^*$ with side lengths $H < H^*$, and the space of A-harmonic functions in ω^* (and similarly for ω) defined by

$$U(\omega^*) := \{v \in H^1(\omega^*) : -\nabla \cdot (A\nabla v) = 0, \text{ in } \omega^*\}/\mathbb{R}\,. \tag{1.1.4}$$

We introduce the notation $\|\cdot\|_{H^1_A(\omega^*)} = \|A^{1/2}\nabla \cdot\|_{L^2(\omega^*)}$. Then, via an iterative argument of Caccioppoli's inequality [15, 179], one can show that the singular values $\sigma_m(R)$ of the restriction operator

$$R : (U(\omega^*), \|\cdot\|_{H^1_A(\omega^*)}) \to (U(\omega), \|\cdot\|_{H^1_A(\omega)})$$

decays exponentially fast:

$$\sigma_m(R) \leq C \exp\left(-bm^{\frac{1}{d+1}}\right) \tag{1.1.5}$$

for some C, b independent of H and m. Equivalently, it implies that we can find m functions $v_k \in U(\omega), 1 \leq k \leq m$ (which are left singular vectors of R), such that

$$\|u - \sum_{i=1}^{m} c_i v_i\|_{H^1_A(\omega)} \leq C \exp\left(-bm^{\frac{1}{d+1}}\right) \|u\|_{H^1_A(\omega^*)}, \tag{1.1.6}$$

for any $u \in H^1_A(\omega^*)$. We can interpret the above result as

The restriction of A-harmonic functions is of low approximation complexity.

Remark 1.1.1. *The above interpretation is fairly general. The choice of ω, ω^* to be cubes is for simplicity of analysis only. The decay will still hold when the two cubes are not concentric but share some boundary face if one further assumes functions in $U(\omega^*)$ satisfy some corresponding homogeneous boundary condition in that face.*

In the initial work [15], the bound is $\sigma_m(R) \leq C_\epsilon \exp\left(-m^{\frac{1}{d+1}-\epsilon}\right)$ for any positive ϵ, where C_ϵ depends on ϵ. Later, the work [179] removes such ϵ to get (1.1.5)*.*

Multiscale methods based on the partition of unity functions Building on the above fact, the works [15, 16, 179] develop exponentially convergent multiscale methods for solving elliptic equations with rough coefficients. The method uses the low complexity of A-harmonic functions locally and applies partition of unity functions [184] to connect local and global scales. More precisely, for the equation (1.1.1) with $V = 0$, one can write the solution in the form

$$u = \sum_i \eta_i u = \sum_i \eta_i u^{\mathsf{h}}_{\omega_i} + \sum_i \eta_i u^{\mathsf{b}}_{\omega_i}, \tag{1.1.7}$$

where $\{\eta_i\}_i$ are partition of unity functions subordinate to an *overlapped domain decomposition* $\{\omega_i\}_i$ and $u^{\mathsf{h}}_{\omega_i}, u^{\mathsf{b}}_{\omega_i}$ are obtained by the *harmonic-bubble splitting* in the local domain ω_i:

$$\begin{cases} -\nabla \cdot (A \nabla u^{\mathsf{h}}_{\omega_i}) = 0, & \text{in } \omega_i \\ u^{\mathsf{h}}_{\omega_i} = u, & \text{on } \partial\omega_i, \end{cases}$$
$$\begin{cases} -\nabla \cdot (A \nabla u^{\mathsf{b}}_{\omega_i}) = f, & \text{in } \omega_i \\ u^{\mathsf{b}}_{\omega_i} = 0, & \text{on } \partial\omega_i. \end{cases} \tag{1.1.8}$$

The part $u^{\mathsf{b}}_{\omega_i}$ is locally solvable since we know both the right-hand side and the boundary condition. The part $u^{\mathsf{h}}_{\omega_i}$ is A-harmonic so $\eta_i u^{\mathsf{h}}_{\omega_i}$ can be seen as a restriction

of A-harmonic functions. Thus, similar arguments using Caccioppoli's inequality imply that $\eta_i u^{\mathsf{h}}_{\omega_i}$ can be approximated by local basis functions with a nearly exponential convergence rate. Pre-computing these basis functions by solving related local spectral problems once, one can combine them with Galerkin's method to obtain an exponentially convergent solver for $\sum_i \eta_i u^{\mathsf{h}}_{\omega_i}$. Adding back the term $\sum_i \eta_i u^{\mathsf{b}}_{\omega_i}$ leads to exponential convergence of accuracy for solving u.

1.1.3.2 Our Contributions: Helmholtz's Equations and Non-overlapped Domain Decomposition

We extend results in Section 1.1.3.1 to solve Helmholtz's equations and develop our alternative multiscale framework based on non-overlapped domain decomposition, which can lead to more localized subdomains and basis functions with better "orthogonality" properties. We describe our idea briefly here, and details are presented in Chapter II, based on our works [46, 48, 47].

Extension to Helmholtz's equations The crucial step of the exponential convergence result in Section 1.1.3.1 is the low complexity of the restriction of A-harmonic functions, proved by an iterative argument of Caccioppoli's inequality. However, Caccioppoli's inequality holds for more general second-order elliptic equations, which allows us to generalize the result beyond A-harmonic functions.

Consider the prototypical equation (1.1.1) with $Vu = -k^2 u$, which represents the Helmholtz equation. Locally, on a mesh of size $H = O(1/k)$, the operator

$$u \to -\nabla \cdot (A\nabla u) - k^2 u$$

becomes positive definite. To see this, using the scaling transformation $x \to x/H$, the operator transforms to $u \to \frac{1}{H^2}(-\nabla \cdot (A\nabla u) - H^2 k^2 u)$ in a domain of size $O(1)$. Once $kH = O(1)$, the strong ellipticity of $-\nabla \cdot (A\nabla \cdot)$ dominates due to Poincaré's inequality, and thus the overall operator is elliptic and positive definite. This fact has been explored by Peterseim to obtain a provable multiscale method for Helmholtz's equations under the framework of Local Orthogonal Decomposition (LOD) [214].

We will use the above observation to design exponentially convergent multiscale methods for Helmholtz's equations. We define

$$U(\omega^*) := \{v \in H^1(\omega^*) : -\nabla \cdot (A\nabla v) - k^2 v = 0, \text{ in } \omega^*\}/\mathbb{R}\,.$$

Consider two concentric cubes $\omega \subset \omega^*$ with side lengths $H < H^* = O(1/H)$, which is the same set-up as in Section 1.1.3.1. Using an iterative argument of Caccioppoli's

inequality and the fact that the local equation is elliptic and positive definite, we can show that, for the operator

$$R : (U(\omega^*), \|\cdot\|_{\mathcal{H}(\omega^*)}) \to (U(\omega), \|\cdot\|_{\mathcal{H}(\omega)}),$$

where the definitions of $\|\cdot\|_{\mathcal{H}(\omega)}, \|\cdot\|_{\mathcal{H}(\omega^*)}$ follow from (1.1.3), its singular values decay exponentially fast similar to that in Section 1.1.3.1; see our work [48] and a contemporary work [177] with more refined analysis. This fact allows us to extend the exponentially convergent methodology to general Helmholtz's equations [48, 177], under the mesh constraint $H = O(1/k)$ for the local subdomains.

Remark 1.1.2. *It is worth noting that, for the Helmholtz equation, even in the case of a smooth A, the mesh size in a standard FEM needs to satisfy conditions like $H = O(1/k^2)$ to obtain accurate solutions due to the indefiniteness of the Helmholtz operator. This phenomenon is known as pollution effects [19]. Thus the condition $H = O(1/k)$ in the multiscale method is sufficient to overcome the pollution effect.*

Multiscale methods based on non-overlapped domain decomposition In Section 1.1.3, the multiscale method relies on formula (1.1.7) to localize the approximation problem, using partition of unity functions. While this method is general and leads to conformal basis functions, it adds a tuning parameter for choosing the partition of unity functions. It can result in large subdomain sizes due to the overlapping. Since we need to compute local spectral problems and solve local equations frequently, it is desirable to have smaller subdomains.

We propose an alternative framework that utilizes *non-overlapped domain decomposition* to couple local and global scales. Instead of partition of unity functions, we use indicator functions in non-overlapped subdomains $\{T_i\}_i$; we represent the solution u as

$$u = \sum_i \mathbb{1}_{T_i} u = \sum_i \mathbb{1}_{T_i} u_{T_i}^{\mathsf{h}} + \sum_i \mathbb{1}_{T_i} u_{T_i}^{\mathsf{b}}, \tag{1.1.9}$$

where $u_{T_i}^{\mathsf{h}}, u_{T_i}^{\mathsf{b}}$ satisfy (1.1.8) with ω_i replaced by T_i and the PDE part replaced by $-\nabla\cdot(A\nabla v) - k^2 v$. The component $\sum_i \mathbb{1}_{T_i} u_{T_i}^{\mathsf{h}}$ is fully determined by its value on *interface edges* since it satisfies the homogeneous PDE in each T_i. We rely on carefully constructed basis functions on the interfaces to communicate the local and global approximation. The geometric structure on the interface adds design complexity and analysis efforts, but the resulting basis functions are more localized. Moreover, the lack of overlapping provides better orthogonality for sub-components

in (1.1.9); this orthogonality is useful when assembling the stiffness and mass matrices in Galerkin's method.

In summary, our work studies fundamentally multiscale methods based on non-overlapped domain decomposition. We demonstrate for the first time its potential to achieve exponential convergence without any partition of unity functions. We present the details of the method in Chapter II.

Remark 1.1.3. *The decomposition in* (1.1.9) *dates back to the MsFEM [129, 128] and approximate component mode synthesis [123, 122], without provable exponential convergence. We achieve exponential convergence for solving elliptic and Helmholtz's equations with rough coefficients and term our approach the Exponentially Convergent Multiscale Finite Element Method (ExpMsFEM) [46, 48, 47].*

1.1.4 Dual Perspective: Selecting Coarse Variables

Multiscale methods classified into the dual perspective feature a primary focus on coarse scale variables, in contrast to the primal perspective whose focus is on coarse scale spaces. Examples of methods that may be interpreted via the dual perspective include the Variational Multiscale Method (VMS) [133], the Heterogeneous Multiscale Method (HMM) [1], and in the case of rough coefficients, Local Orthogonal Decomposition (LOD) [181, 121, 148, 117, 180] and Gamblets related approaches [208, 210, 201, 203, 131, 204, 45].

We will illustrate a very successful multiscale approach that underlies LOD and Gamblets. Consider the elliptic equation (1.1.1), where $V = 0$, $u \in H_0^1(\Omega)$ and $f \in L^2(\Omega)$ in $\Omega = [0,1]^d$. We denote $\mathcal{L} = -\nabla \cdot (A\nabla \cdot)$. Suppose we want to obtain the following coarse variables: $[u, \phi_i]$, where $i \in I$, and ϕ_i is some measurement function in $H^{-1}(\Omega)$. Here I is an index set, and $[\cdot, \cdot]$ denotes the standard L^2 inner product.

Remark 1.1.4. *As an example, we can take $[u, \phi_i], i \in I$ to be local cell averages of u. Intuitively, these quantities can represent the coarse-scale behavior of the solution and are thus suitable as coarse variables.*

Given the PDE information $\mathcal{L}u = f$, an ideal approach to obtain these coarse variables is to multiply the equation with a set of basis functions $\{\psi_i\}_{i\in I}$ that satisfy

$$\text{span } \{\psi_i\}_{i\in I} = \text{span } \{\mathcal{L}^{-1}\phi_i\}_{i\in I}\,. \tag{1.1.10}$$

In this case, after integration by parts, we obtain the information of $[u, \phi_i], i \in I$.

Remark 1.1.5. *A particularly useful representation of ψ_i that satisfies* (1.1.10) *can be obtained by the following optimization problem:*

$$\begin{aligned} \psi_i = argmin_{\psi \in H_0^1(\Omega)} \quad & \int_\Omega A|\nabla \psi|^2 \\ \text{subject to} \quad & [\psi, \phi_j] = \delta_{i,j} \ \text{ for } \ j \in I\,. \end{aligned} \tag{1.1.11}$$

A Bayesian interpretation of the formula is presented in [203, 204], which shows that

$$\psi_i = \mathbb{E}[\xi | [\xi, \phi_j] = \delta_{i,j} \ \text{ for } \ j \in I]\,,$$

where ξ is a Gaussian process with mean 0 *and covariance operator $\mathcal{L}^{-1}$.*

We can think of $\text{span}\{\psi_i\}_{i\in I}$ as the test space in Galerkin's method since we multiply the equation by them. One must still select a trial space S to solve the problem. At this stage, the problem is similar to *scattered data approximation*, where one also needs to find an approximation space S to interpolate the scattered data $[u, \phi_i], i \in I$.

In solving PDEs, it is common to choose the same trial and test space $S = \text{span}\{\psi_i\}_{i\in I}$. Under this setting, if we specifically select ϕ_i to be piecewise linear tent functions, we can recover the LOD method proposed in [181]. On the other hand, setting ϕ_i to be piecewise constant functions leads to the Gamblets method in [203]. They can handle rough coefficients provably.

The basis functions ψ_i described in (1.1.11) may have global support, which requires further localization for practical computations. The localization is first achieved in [181] then generalized in [203] by showing that ψ_i exhibits exponential decay in physical space. This allows one to replace the space $H_0^1(\Omega)$ in (1.1.11) by $H_0^1(\omega_i)$ with a local domain ω_i, resulting in a localized approximation for ψ_i.

1.1.4.1 Our Contribution: Subsampled Lengthscale in the Coarse Variables

We contribute to a new line of analysis [44, 45], summarized in Chapter III, for the multiscale approach illustrated in Section 1.1.4. Our analysis reveals an overlooked accuracy-efficiency tradeoff in this methodology, providing insights for the selection of coarse variables in multiscale PDEs, as well as data collection in scattered data approximation and machine learning.

Our idea is to choose ϕ_i as *subsampled measurement functions*. To illustrate we decompose the domain $\Omega = [0,1]^d$ uniformly into cubes with side length H. Let I be the index set of these cubes with cardinality $|I| = 1/H^d$. For each $i \in I$, the

measurement function $\phi_i^{h,H}$ is set to be the indicator function of a cube $\omega_i^{h,H}$ with side length $0 < h \leq H$, centered in the corresponding cube ω_i^H with side length H; see Figure 1.1 for a 2D example[2]. We refer to H as the coarse lengthscale and h as the *subsampled lengthscale*. When $h = H$, $\phi_i^{h,H}$ is the same as the measurement function in Gamblets [203], while $h = 0$ leads to Diracs-type measurement. Subsampled measurement functions interpolate between the two.

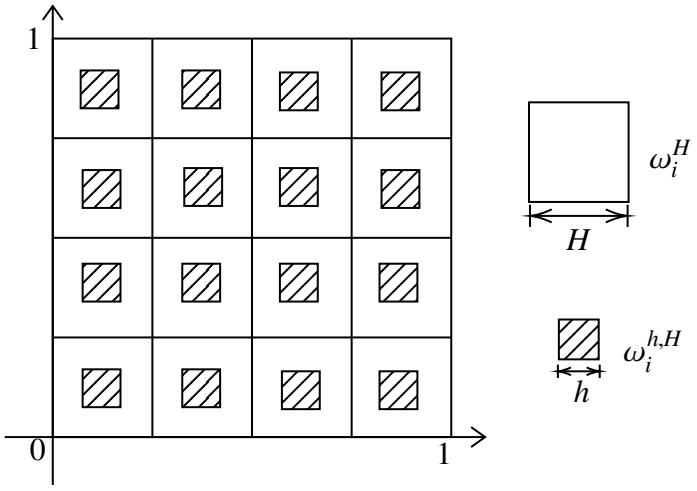


Figure 1.1: Illustration of Subsampled Measurements: $H = 1/4, h = 1/10$

The subsampled measurement function is a natural concept in scattered data approximation and machine learning, leading to cell-average-type data. Understanding the effect of h is helpful in guiding the data collection procedure. In the context of solving PDEs, we have the freedom to choose the coarse variables. It is interesting to understand how the value of h influences the accuracy and efficiency of the multiscale method, which can provide a guiding principle for numerical coarse-graining in general multiscale problems.

Intuitively, decreasing h may reduce the information content but can potentially make (1.1.11) decay faster as the information is more localized, resulting in better localization and faster computation of ψ_i. Our analysis characterizes the dependence of accuracy and exponential decay rate on the parameter h. We demonstrate that a *non-monotonic* behavior exists in the dependence of the exponential decay rate on h. Our theory and numerical experiments show that, given any localization radius, there exists a "sweet spot" of h that achieves the best accuracy, both in scattered data approximation and in solving multiscale PDEs. Thus, the results provide hints on

[2]Note that the choice of ω_i^H and $\omega_i^{h,H}$ being cubes here is for convenience of analysis only.

choosing the measurement functions to maximize performance. We also study the limit case of $h \to 0$ and connect it to the degeneracy issue in graph Laplacian-based methods. The detailed study is presented in Chapter III.

1.1.5 Summary

Section 1.1 provides an overview of the topics in multiscale numerical methods covered in this thesis. Multiscale methods have been a successful and active field in applied and computational mathematics. This thesis contributes to understanding exponentially convergent methods for Helmholtz's equations based on non-overlapped domain decomposition without partition of unity functions. Additionally, this thesis contributes to understanding the influence of subsampled lengthscale on accuracy and efficiency in multiscale methods and scattered data approximation.

As discussed, multiscale methods are specialized approaches designed to tackle very challenging problems, and they are powerful in their targeted problem class. However, they often require a case-by-case study for different equations, and most theoretical work is limited to solving linear PDEs. In practice, there is also a desire for general solvers with an automatic flavor that can be used to test a wide range of problems in PDE models and data science. This leads to our discussion about scientific machine learning and statistical numerical methods in the next section.

1.2 Statistical Numerical Methods

In recent years, there has been a growing trend toward automating the solution of computational problems through machine learning methods, particularly those based on neural networks (NNs). The success of NNs in applications such as computer vision, natural language processing, and game-playing has inspired researchers to explore the potential of machine learning in scientific computing.

One popular research direction is the automation of solving PDEs and inverse problems (IPs). Researchers have developed various physics-informed machine learning methods by viewing PDEs as sampled data at collocation points and integrating them into NNs-based statistical machine learning pipelines, for instance, the Physics Informed Neural Networks (PINNs) [222]. These methods are *automatic* and *easy to implement*, thanks to the well-developed auto-differentiation platforms such as PyTorch and JAX. We also note there is another line of research focusing on using NNs to approximate the input-to-solution map, leading to operator learning [163, 29, 196, 172].

While the NN-based approach has demonstrated empirical success in solving PDEs and inverse problems, it may need more rigor, efficiency, and accuracy. Indeed, the theoretical study of NNs is limited to specific cases [249, 174, 67], and their training often requires significant tuning and takes longer than traditional solvers [106]. Consequently, significant research efforts have been devoted to stabilizing and accelerating the training process to enhance efficiency and accuracy [150, 282, 284, 64, 299].

The second part of this thesis aims to investigate a more interpretable (machine learning) approach to automate the solution of general PDEs and IPs based on Gaussian processes. The design of the methodology is in line with the idea of statistical numerical methods: using statistical inference for numerical computation.

1.2.1 Statistical Inference for Numerical Computation

While using NNs to solve PDEs as a statistical machine learning problem has gained popularity mainly in the last few years, it is important to note that applying statistical inference to computational problems is not a new concept. As surveyed in the review paper [205]:

> Although numerical approximation and statistical inference are traditionally seen as entirely separate subjects, they are intimately connected through the common purpose of making estimations with partial information. This shared purpose is currently stimulating a growing interest in statistical inference/machine-learning approaches to solving PDEs [201, 223], in the use of randomized algorithms in linear algebra [113], and in the merging of numerical errors with modeling errors in uncertainty quantification [120]. While this interest might be perceived as a recent phenomenon, interplays between numerical approximation and statistical inference are not new. Indeed, they can be traced back to Poincaré's course in probability theory (1896) and to the pioneering investigations of Sul'din [263], Palasti and Renyi [212], Sard [237], Kimeldorf and Wahba [142] (on the correspondence between Bayesian estimation and spline smoothing/interpolation), and Larkin [157] (on the correspondence between Gaussian process regression and numerical approximation). Although their study initially "attracted little attention among numerical analysts" [157], it was revived in information-based complexity (IBC) [268], Bayesian numerical analysis [69], and more

recently in probabilistic numerics [120].

In accordance with the principles of statistical inference for numerical approximations, in this thesis we adopt Gaussian process regression and Bayes inference to solve *nonlinear* PDEs and IPs. We address the associated mathematical questions regarding consistency, efficiency, adaptivity, and more in a general and rigorous way.

Remark 1.2.1. *The reason to use GPs is that they are generally easier to interpret and optimize than NNs. In the meanwhile, GPs emerge in the infinite-width limit [194, 158] (and neural tangent kernel limit [134]) of NNs, so they also closely connect to NN-based approaches. Theoretically, GP-based methods share deep connections with kernel methods, which are linked to radial basis function-based approaches in numerical analysis, making them amenable to rigorous analysis. See also Remark 1.2.5.*

We outline our contributions in Sections 1.2.2-1.2.5, covering methodology, efficiency, adaptivity, and others (including high dimensional data science applications and posterior sampling). Notably, all our contributions use probability and statistics in specific ways. One may think they result in a probabilistic pipeline for computations in scientific machine learning, from function representation to numerical solvers and uncertainty quantification. We summarize the discussions in Section 1.2.6.

1.2.2 Methodology: Solving Nonlinear PDEs and IPs with GPs

In Chapter IV, we introduce GPs to solve nonlinear PDEs and IPs, based on our work [43]. As noted in Section 1.2.1, this concept is not new, and pre-existing works have mainly focused on linear PDEs. Our goal is to establish a more comprehensive framework covering nonlinear problems and address the new mathematical questions that have arisen.

Our approach involves assigning a GP prior distribution to the solution and computing the Maximum A Posterior (MAP) estimator conditioned on both the PDE at collocation points and data observations (if there are any). To illustrate, consider

the following nonlinear elliptic PDE[3]:

$$\begin{cases} -\Delta u + \tau(u) = f & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{1.2.1}$$

where τ is a nonlinear scalar function and Ω is a bounded open domain in $\mathbb{R}^d$ with a Lipschitz boundary. We assume the equation has a strong solution in the classical sense.

We sample M_Ω collocation points in the interior and $M_{\partial\Omega}$ on the boundary such that

$$\mathbf{x}_\Omega = \{\mathbf{x}_1, ..., \mathbf{x}_{M_\Omega}\} \subset \Omega \quad \text{and} \quad \mathbf{x}_{\partial\Omega} = \{\mathbf{x}_{M_\Omega+1}, ..., \mathbf{x}_M\} \subset \partial\Omega,$$

where $M = M_\Omega + M_{\partial\Omega}$. Then, by assigning a GP prior to the unknown function u with mean 0 and covariance kernel function $K : \overline{\Omega} \times \overline{\Omega} \to \mathbb{R}$, the method aims to compute the MAP estimator of the GP given the sampled PDE data, which leads to the following optimization problem:

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad -\Delta u(\mathbf{x}_m) + \tau(u(\mathbf{x}_m)) = f(\mathbf{x}_m), & \text{for } m = 1, \ldots, M_\Omega, \\ \qquad u(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \ldots, M. \end{cases} \tag{1.2.2}$$

Here, $\|\cdot\|$ is the Reproducing Kernel Hilbert Space (RKHS) norm corresponding to the kernel/covariance function K, and $\mathcal{U}$ is the corresponding RKHS.

Remark 1.2.2. *For ease of understanding, one can conceptually think:*

- *The RKHS norm* $\|u\|^2 = [u, \mathcal{K}^{-1}u]$ *where* $\mathcal{K}$ *is the integral operator associated with* k *such that* $(\mathcal{K}v)(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{y})v(\mathbf{y})\mathrm{d}\mathbf{y}$*, and* $[\cdot, \cdot]$ *is the* L^2 *inner product;*
- *The "density" of the GP is proportional to* $\exp\left(-\frac{1}{2}\|u\|^2\right)$.

Maximizing the density is equivalent to minimizing the norm, which justifies the "MAP" interpretation of the optimization problem.

[3]While the example for illustration does not involve data observations (for simplicity of presentation), the framework we outline can be applied directly to cases where data observations are present, such as IPs.

We can separate the linear differential operators and the nonlinear relationship in the PDE by rewriting the optimization problem into a nested one:

$$\begin{cases} \underset{\mathbf{z}^{(1)}\in\mathbb{R}^M,\mathbf{z}^{(2)}\in\mathbb{R}^{M_\Omega}}{\text{minimize}} & \begin{cases} \underset{u\in\mathcal{U}}{\text{minimize}} & \|u\| \\ \text{s.t.} & u(\mathbf{x}_m) = z_m^{(1)} \text{ and } \Delta u(\mathbf{x}_m) = z_m^{(2)}, \text{ for } m = 1,\dots,M, \end{cases} \\ \text{s.t.} & -z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), \quad \text{for } m = 1,\dots,M_\Omega, \\ & z_m^{(1)} = g(\mathbf{x}_m), \quad \text{for } m = M_\Omega+1,\dots,M. \end{cases}$$

Here, $\mathbf{z}^{(1)} \in \mathbb{R}^M$, $\mathbf{z}^{(2)} \in \mathbb{R}^{M_\Omega}$ are slack variables. We write $\mathbf{z} \in \mathbb{R}^N$ with $N = M + M_\Omega$ as a concatenation of $\mathbf{z}^{(1)}$ and $\mathbf{z}^{(2)}$.

The constraint in the inner minimization problem is linear. By the linear theory of Gaussian process regression, we know the optimal solution attains an explicit formula:

$$u(\mathbf{x}) = K(\mathbf{x},\boldsymbol{\phi})K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z}. \tag{1.2.3}$$

Remark 1.2.3. *We will elaborate more on the notations $K(\mathbf{x},\boldsymbol{\phi})$ and $K(\boldsymbol{\phi},\boldsymbol{\phi})$ in Chapter IV; for this specific example, we can write down $K(\mathbf{x},\boldsymbol{\phi})$ and $K(\boldsymbol{\phi},\boldsymbol{\phi})$ explicitly:*

$$\begin{aligned} K(\mathbf{x},\boldsymbol{\phi}) &= \big(K(\mathbf{x},\mathbf{x}_\Omega), K(\mathbf{x},\mathbf{x}_{\partial\Omega}), \Delta_{\mathbf{y}}K(\mathbf{x},\mathbf{x}_\Omega)\big) \in \mathbb{R}^{1\times N}, \\ K(\boldsymbol{\phi},\boldsymbol{\phi}) &= \begin{pmatrix} K(\mathbf{x}_\Omega,\mathbf{x}_\Omega) & K(\mathbf{x}_\Omega,\mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{y}}K(\mathbf{x}_\Omega,\mathbf{x}_\Omega) \\ K(\mathbf{x}_{\partial\Omega},\mathbf{x}_\Omega) & K(\mathbf{x}_{\partial\Omega},\mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{y}}K(\mathbf{x}_{\partial\Omega},\mathbf{x}_\Omega) \\ \Delta_{\mathbf{x}}K(\mathbf{x}_\Omega,\mathbf{x}_\Omega) & \Delta_{\mathbf{x}}K(\mathbf{x}_\Omega,\mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{x}}\Delta_{\mathbf{y}}K(\mathbf{x}_\Omega,\mathbf{x}_\Omega) \end{pmatrix} \in \mathbb{R}^{N\times N}. \end{aligned} \tag{1.2.4}$$

Here, $\Delta_{\mathbf{x}}, \Delta_{\mathbf{y}}$ are the Laplacian operator for the first and second arguments of k, respectively. We adopt the convention that if the variable inside a function is a set, it means that this function is applied to every element in this set; the output will be a vector or a matrix, e.g., $K(\mathbf{x}_\Omega,\mathbf{x}_\Omega) \in \mathbb{R}^{M_\Omega\times M_\Omega}$.

It remains to solve the outer optimization problem, which is finite dimensional:

$$\begin{cases} \underset{\mathbf{z}\in\mathbb{R}^N}{\text{minimize}} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & -z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), \quad \text{for } m = 1,\dots,M_\Omega, \\ & z_m^{(1)} = g(\mathbf{x}_m), \quad \text{for } m = M_\Omega+1,\dots,M. \end{cases} \tag{1.2.5}$$

Remark 1.2.4. *From* (1.2.3)*,* (1.2.4)*, and* (1.2.5)*, we can understand the methodology as a generalization of radial basis functions based meshless methods [239]. Specifically, it is the same as the symmetric collocation method for solving linear PDEs, e.g., when $\tau = 0$.*

More generally, for any PDEs, our methodology leads to the optimization problem

$$\begin{cases} \min\limits_{u\in\mathcal{U}} & \|u\| \\ \text{s.t.} & \text{PDE constraints at } \{\mathbf{x}_1,\dots,\mathbf{x}_M\} \in \overline{\Omega}, \end{cases} \tag{1.2.6}$$

and the equivalent finite dimensional problem

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbb{R}^N} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}, \end{cases} \tag{1.2.7}$$

where $\boldsymbol{\phi}$ is the concatenation of measurements of u induced by the PDE at the sampled points; we will explain them in detail in Chapter IV. The function F encodes the PDE, and the vector $\mathbf{y}$ encodes the right hand side and boundary data.

Theoretically, we prove that the methodology converges in the large data limit if the exact solution lives in the RKHS of the covariance kernel of the GP. Convergence rates can be further obtained if the stability of the PDE is assumed [23]. Numerically, we solve the above quadratic programming problem with nonlinear constraints via sequential quadratic programming. This is equivalent to a Gauss-Newton-type algorithm in the case of (1.2.5). In Chapter IV, we present numerical results, showing that $2-10$ iterations suffice to converge for a wide array of problems such as nonlinear elliptic PDEs, Burgers' equation, a regularized Eikonal equation, and the Darcy flow inverse problem.

Remark 1.2.5. *From formulas* (1.2.6) *and* (1.2.7)*, we see the methodology is quite general and has a similar flavor to NN-based machine learning solvers: the selection of kernels resemble that of NNs, and the solution is obtained by solving an optimization problem for any PDEs and IPs. However, the GP/kernel method is simpler, more interpretable, and amenable to analysis. In Section 1.2.3, we show another advantage of GP-based methods for low-dimensional PDEs: we can use fast solvers for dense kernel matrices to accelerate the optimization process. Additionally, we point out that the GP methodology provides a natural pipeline for uncertainty quantification due to its probabilistic interpretation.*

On the other hand, the function representation provided by the GP methodology is essentially linear, as the function space is the linear span of kernel functions. This is a disadvantage, as NNs are known to be successful partly because of their high expressivity. In Section 1.2.4, we discuss ways of using hierarchical learning to make GPs more adaptive and expressive.

1.2.3 Efficiency: Sparse Cholesky Factorization for GP-PDEs

The complexity bottleneck of GP-based methods lies in computing with dense kernel matrices. In the case of PDE problems, these matrices may also involve partial derivatives of the kernels (e.g., (1.2.4)), and fast algorithms for such matrices are less developed [81, 211, 66] compared to the derivative-free counterparts.

In Chapter V, we present a sparse Cholesky factorization algorithm for such kernel matrices, based on our work [50]. The algorithm relies on the near-sparsity of the Cholesky factor under a *multiscale* ordering of the pointwise and derivative-type entries of the matrices. In fact, it is known as a common practice that reordering the columns of a positive definite matrix may lead to better sparsity in its Cholesky factor. Recently in [241, 240], the authors provide a rigorous analysis of such phenomenon for a kernel matrix with derivative-free entries when the kernel is the Green function of some differential operators. Based on the analysis, they propose a sparse Cholesky factorization algorithm to factorize the inverse of the kernel matrix in near-linear time. The contribution of our work is to generalize the analysis and algorithms to kernel matrices with derivative entries, providing a rigorous algorithmic framework for scaling up computations in GPs for PDEs.

Central to the methodology is the interplay between linear algebra, Gaussian process conditioning, screening effects, and numerical homogenization. More precisely, consider a kernel matrix $\Theta \in \mathbb{R}^{N\times N}$, and a Gaussian random variable $Y \sim \mathcal{N}(0,\Theta)$. Suppose $\Theta^{-1} = U^\star U^{\star T}$ where $U^\star$ is the upper Cholesky factor, then we have the following relation:

$$\frac{U^\star_{ij}}{U^\star_{jj}} = (-1)^{i\neq j}\frac{\mathrm{Cov}[Y_i, Y_j|Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i|Y_{1:j-1\setminus\{i\}}]}, \quad i \leq j\,. \tag{1.2.8}$$

Here we used the MATLAB notation such that $Y_{1:j-1\setminus\{i\}}$ corresponds to $\{Y_q : 1 \leq q \leq j-1, q \neq i\}$.

The formula (1.2.8) links the values of $U^\star$ to the conditional covariance of a Gaussian variable. Note that in the GP picture, this Gaussian variable is a subsampling of a continuous GP. In spatial statistics, it is empirically well-known that conditioning a GP on coarse-scale measurements results in very small covariance values between finer-scale measurements. This phenomenon, known as *screening effects*, has been discussed in works such as [259, 256]. The implication is that conditioning on coarse scales screens out fine-scale interactions. Thus, based on this observation, one expects $U^\star$ to become approximately sparse if we reorder the columns of the

kernel matrix so that coarse scale measurements come before finer ones. This is the rationale behind the works of sparse Cholesky factorization [241, 240] when the kernel matrix does not contain derivatives of the kernel.

Our work's key contribution is figuring out how to order the derivative entries in the matrix to achieve similar sparsity. To that end, we point out a remarkable relation that connects the conditional covariance to conditional expectations:

$$\left|\frac{U^\star_{ij}}{U^\star_{jj}}\right| = \left|\frac{\mathrm{Cov}[Y_i, Y_j | Y_{1:j-1\setminus i}]}{\mathrm{Var}[Y_i | Y_{1:j-1\setminus i}]}\right| = \left|\mathbb{E}[Y_j | Y_i = 1, Y_{1:j-1\setminus i} = 0]\right| .$$

The last term in the above identity connects to the Bayes interpretation of basis functions (Gamblets) in multiscale methods; see Remark 1.1.5. This allows us to use analytic results in multiscale methods[4] to show the exponential decay of $\left|U^\star_{ij}/U^\star_{jj}\right|$. The analysis suggests that we need to treat derivative entries as *finer* scales compared to pointwise entries in the kernel matrix. Under such ordering, we rigorously identify the sparsity pattern and quantify the *exponentially convergent* accuracy of the corresponding Vecchia approximation [278, 140] of the GP, which provides a sparse approximation of Cholesky factors of Θ^{-1}. This enables us to compute ϵ-approximate inverse Cholesky factors of the kernel matrices with a state-of-the-art complexity $O(N \log^d(N/\epsilon))$ in space and $O(N \log^{2d}(N/\epsilon))$ in time, for d-dimensional PDE problems.

With the sparse factors, gradient-based optimization methods become scalable. Furthermore, we can use the often more efficient Gauss-Newton method for solving the optimization problem. In such case, we can apply the conjugate gradient algorithm with the sparse factor of a reduced kernel matrix as a preconditioner to solve the linear system. Our numerical experiments in Chapter V illustrate the algorithm's near-linear space/time complexity for a broad class of nonlinear PDEs such as the nonlinear elliptic, Burgers, and Monge-Ampère equations.

1.2.4 Adaptivity: Hierarchical Learning and Consistency Analysis

Compared to NNs, the function representation provided by the GP methodology is essentially *linear* since the function space is the linear span of kernel functions. To improve the expressivity and adaptivity of GP-based methods, hierarchical learning can be applied. In fact, a crucial aspect of the success of Gaussian processes, in

[4]Note that the works [241, 240] also employ the connection to multiscale methods and numerical homogenization to provide rigorous analysis of the sparse Cholesky factorization algorithms. However, their analysis relies additionally on operator-valued wavelets and the matrix algebra of exponential decay matrices. Our proof is much simpler and applies to the case of derivative entries.

a wide range of applications to complex and real-world problems, is hierarchical modeling and learning of hyperparameters. We contribute to some analysis of hierarchical learning in Chapter VI, based on our work [51].

To illustrate, consider a model problem of learning a function $u^\dagger : D \subset \mathbb{R}^d \to \mathbb{R}$ from pointwise observed data: $u(\mathbf{x}_i) = y_i, 1 \le i \le N$. Given a family of positive definite covariance/kernel functions $K_\theta : D \times D \to \mathbb{R}$ where $\theta \in \Theta$ is a hyperparameter, Gaussian process regression approximates $u^\dagger$ with the conditional expectation

$$u(\cdot, \theta, \mathcal{X}) := \mathbb{E}\left[\xi(\cdot, \theta) \mid \xi(\mathcal{X}, \theta) = u^\dagger(\mathcal{X})\right] = K_\theta(\cdot, \mathcal{X})[K_\theta(\mathcal{X}, \mathcal{X})]^{-1} u^\dagger(\mathcal{X}), \quad (1.2.9)$$

where $\xi(\cdot, \theta) \sim \mathcal{GP}(0, K_\theta)$ is a centered Gaussian process with covariance function K_θ. We have used the following compressed notation:

$$\mathcal{X} := (\mathbf{x}_1, \ldots, \mathbf{x}_N)^\mathsf{T} \quad \text{and} \quad u^\dagger(\mathcal{X}) := (u^\dagger(\mathbf{x}_1), \ldots, u^\dagger(\mathbf{x}_N))^\mathsf{T}.$$

Moreover, $K_\theta(\mathcal{X}, \mathcal{X})$ denotes the $N \times N$ dimensional Gram matrix with $(i, j)^{\text{th}}$ entry $K_\theta(\mathbf{x}_i, \mathbf{x}_j)$, and $K_\theta(\cdot, \mathcal{X})$ is a function mapping D to $\mathbb{R}^N$ with i^{th} component $K_\theta(\cdot, \mathbf{x}_i) : D \mapsto \mathbb{R}$. Note that the notation used here is slightly different from that in previous sections, as it pertains to a different setting.

Hierarchical learning the GP/kernel aims to select a good θ. In general, we have two paradigms for learning the kernels: the Bayes approach intrinsic to GPs and the approximation-theoretic approach centered around Monte Carlo (e.g., cross-validation) and numerical approximations (e.g., Kernel Flow [206]).

Bayes approach The empirical Bayes (EB) approach addresses the question by assuming that θ is sampled from a prior distribution and ξ is then sampled from the conditional distribution of $\xi|\theta$; then, it finds the posterior distribution of the pair (ξ, θ) conditioned on $\xi(\mathcal{X}) = u^\dagger(\mathcal{X})$, and selects the parameter θ that maximizes the marginal probability of θ under this posterior. If we work with uninformative priors, we get the following objective function:

$$\mathsf{L}^{\text{EB}}(\theta, \mathcal{X}, u^\dagger) = u^\dagger(\mathcal{X})^\mathsf{T}[K_\theta(\mathcal{X}, \mathcal{X})]^{-1} u^\dagger(\mathcal{X}) + \log\det K_\theta(\mathcal{X}, \mathcal{X}). \quad (1.2.10)$$

This is also twice the negative marginal log likelihood of θ given the data $u^\dagger(\mathcal{X})$. Then, EB will choose θ by minimizing this objective function, namely

$$\theta^{\text{EB}}(\mathcal{X}, u^\dagger) := \operatorname*{arg\,min}_{\theta \in \Theta} \mathsf{L}^{\text{EB}}(\theta, \mathcal{X}, u^\dagger). \quad (1.2.11)$$

Approximation theoretic approach On the other hand, considerations from approximation theory offer a different approach without proposing statistical models. This methodology involves finding an ideal θ that minimizes $\mathsf{d}(u^\dagger, u(\cdot, \theta, \mathcal{X}))$, where d is a cost function. Although $u^\dagger$ is not available in practice, ideas from cross-validation can split $\mathcal{X}$ into training and validation data and use the approximation error in the validation data to estimate the exact error. Inspired by this idea, one could optimize the following objective function:

$$\mathsf{d}(u(\cdot, \theta, \mathcal{X}), u(\cdot, \theta, \pi\mathcal{X})) \,. \tag{1.2.12}$$

Here, $\pi\mathcal{X}$ represents a subset of $\mathcal{X}$ obtained by subsampling a proportion, such as one-half, of $\mathcal{X}$. We will focus on a particular choice of d originating from the Kernel Flow (KF) approach [206]. To describe it, we denote by $(\mathcal{H}_\theta, \|\cdot\|_{K_\theta})$ the associated RKHS for the kernel K_θ; note that $\|K_\theta(\cdot, x)\|_{K_\theta}^2 = K_\theta(x, x)$. The objective function in KF is chosen as

$$\mathsf{L}^{\mathrm{KF}}(\theta, \mathcal{X}, \pi\mathcal{X}, u^\dagger) := \frac{\|u(\cdot, \theta, \mathcal{X}) - u(\cdot, \theta, \pi\mathcal{X})\|_{K_\theta}^2}{\|u(\cdot, \theta, \mathcal{X})\|_{K_\theta}^2} \,. \tag{1.2.13}$$

This measures the discrepancy in the RKHS norm between the Gaussian Process Regression solution using the whole data $\mathcal{X}$ and using a subset of the data $\pi\mathcal{X}$, normalized by the RKHS norm of the former. It is natural to expect a good θ will make the discrepancy small so the KF estimator is defined as

$$\theta^{\mathrm{KF}}(\mathcal{X}, \pi\mathcal{X}, u^\dagger) := \operatorname*{arg\,min}_{\theta \in \Theta} \mathsf{L}^{\mathrm{KF}}(\theta, \mathcal{X}, \pi\mathcal{X}, u^\dagger) \,. \tag{1.2.14}$$

Consistency analysis and robustness In Chapter VI, we provide the first rigorous analysis for Kernel Flow (and new analysis results for Empirical Bayes), characterizing the large data consistency (i.e., what is $\lim_{N\to\infty} \theta^{\mathrm{EB}}, \theta^{\mathrm{KF}}$) and implicit bias (i.e., how these limits reflect the structure of the problem) in learning a Matérn-like Gaussian process. These results are of interest to statistical learning theory and spatial statistics. In addition, our study illustrates that the Bayes approach is more accurate for well-specified models and approximation-theoretic approaches are more favorable regarding robustness to model misspecification.

1.2.5 Additional Topics: Randomized Numerics and Posterior Sampling

This thesis also covers some other related and prospective topics pertaining to numerical methods based on probability and statistics. Specifically, we study randomized numerical linear algebra to scale up kernel computations in high dimensional

applications and use probability flows to sample posterior distributions in Bayes inference.

1.2.5.1 High Dimensional Problems through Randomized Numerics

The sparse Cholesky factorization algorithm discussed in Section 1.2.3 achieves near-linear complexity and is highly efficient and scalable. However, it is primarily suited for low-dimensional problems where dense neighboring data can screen out far-field interactions, leading to near-sparsity in the Cholesky factors. This sparsity structure enables a full-scale approximation of the dense matrix.

For high-dimensional scientific applications, such as in chemistry, where the data could not sufficiently fill the space, such a sparse Cholesky factorization algorithm may not be performative. Indeed, in high-dimensional problems, it is more reasonable to aim for a low-rank approximation of the kernel matrix. To derive an accurate low-rank approximation, we usually need to find low-complexity structures in the data. For this purpose, *randomized numerical linear algebra* [183] provides a promising and widely recognized tool.

In the first part of Chapter VII, we briefly describe our work on a randomized algorithm, *Randomly Pivoted Cholesky (RPCholesky)*, which provides a provable and efficient low-rank approximation for dense kernel matrices. RPCholesky can be understood as an adaptive Cholesky factorization or Nyström approximation of the kernel matrix with specific pivoting rules. More precisely, the column Nyström approximation approximates $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ via

$$\hat{\boldsymbol{K}}_{\mathsf{S}} = \boldsymbol{K}(:,\mathsf{S})\boldsymbol{K}(\mathsf{S},\mathsf{S})^{\dagger}\boldsymbol{K}(\mathsf{S},:),$$

where $\mathsf{S} = \{s_1,...,s_k\} \subset \{1,...,N\}$ is a carefully chosen set of columns. In this expression, $\boldsymbol{K}(:,\mathsf{S})$ is the submatrix with the selected columns, $\boldsymbol{K}(\mathsf{S},:)$ is the submatrix with the selected rows, and $\boldsymbol{A}(\mathsf{S},\mathsf{S})^{\dagger}$ is the Moore–Penrose pseudoinverse of the submatrix with the selected rows and columns.

RPCholesky selects the columns progressively in a *probabilistic* way: having selected an index set S_m, the next index s_{m+1} is sampled according to the probability

$$\mathbb{P}\{s_{m+1} = i\} = \boldsymbol{R}^{(m)}_{ii}/\mathrm{tr}\boldsymbol{R}^{(m)},$$

where $\boldsymbol{R}^{(m)} = \boldsymbol{K} - \hat{\boldsymbol{K}}_{\mathsf{S}_m}$ is the Schur complement at the m-step.

Compared to the greedy selection $s_{m+1} = \mathrm{argmax}_i \boldsymbol{R}^{(m)}_{ii}$ that has been widely used in experimental design based on posterior variances of Gaussian processes [94]

and numerical linear algebra (known as complete pivoting [124]), RPCholesky strikes a *better balance* between *exploring* small diagonal entries and *exploiting* large ones. This balance of exploration-exploitation is crucial for a robust and accurate approximation of the kernel matrix. We support our intuition through in-depth theoretical and numerical studies [42], which shows RPCholesky matches or improves alternative algorithms in the literature, such as uniform sampling [291] and ridge leverage score sampling [3, 36, 192, 232].

This study highlights the efficacy of randomness in scaling up kernel methods, showcasing their powerful potential in high-dimensional scientific computing.

1.2.5.2 Posterior Sampling through Gradient Flows

Although we adopted the principle of statistical numerical methods to solve PDEs and inverse problems in Section 1.2.2, we only focused on the MAP estimator and did not explore the posterior distributions. As a result, we did not fully harness the potential of a Bayesian perspective.

In Bayesian inference, generating samples from the posterior distributions is often of interest, which can be helpful for uncertainty quantification. The sampling of posterior distributions has primarily been addressed by MCMC. However, while MCMC is guaranteed to converge to the true posterior in the limit, it is typically very slow for large-scale problems.

Recently, gradient flows in the density space based on optimal-transport-type metrics have been influential in generating interacting particle dynamics for sampling distributions [271]. These methods converge to the true posterior in the asymptotic limit and may be faster than pre-existing MCMC. Meanwhile, many MCMC methods in the continuous-time limit can be formulated as gradient flows in the density space; an example is the Langevin dynamics. The optimization perspective of gradient flows also leads to more amenable analysis. These motivate us to study gradient flows as a systematic methodology for sampling.

In the second part of Chapter VII, we briefly describe our work [49], which makes progress in understanding several questions regarding the canonical choice of gradient flows for sampling from the perspective of *invariance*.

We need to choose an energy functional and metric to describe any gradient flow. Formally, let $\mathcal{P}$ be the probability space in $\mathbb{R}^d$, $\mathcal{E}$ be an energy functional on $\mathcal{P}$ that maps to $\mathbb{R}$, and g_ρ be a Riemannian metric on $\mathcal{P}$ at ρ, where $T_\rho\mathcal{P}$ is the tangent

space consisting of measures with zero means. The metric can also be expressed as $g_\rho(\sigma_1, \sigma_2) = \langle \sigma_1, M(\rho)\sigma_2 \rangle_{L^2}$, where $M(\rho)$ is an operator that acts on $T_\rho \mathcal{P}$. The gradient flow equation is given by:

$$\frac{\partial \rho_t}{\partial t} = -\nabla_g \mathcal{E}(\rho_t) = -M(\rho_t)^{-1} \frac{\delta \mathcal{E}}{\delta \rho}|_{\rho=\rho_t},$$

where ∇_g is the Riemannian gradient operator and $\frac{\delta \mathcal{E}}{\delta \rho}$ is the first variation of $\mathcal{E}$. We can think $M(\rho)^{-1}$ as a preconditioner for the "Euclidean gradient" $\frac{\delta \mathcal{E}}{\delta \rho}$. Numerical simulation of the flow can lead to sampling algorithms. Some examples of gradient flows and their corresponding numerical scheme include

- Wasserstein gradient flow [135, 200] and Langevin's dynamics;
- Stein variational gradient flow [168] and Stein variational gradient descent [169];
- Wasserstein-Fisher-Rao gradient flow [175] and birth-death dynamics;
- Kalman-Wasserstein gradient flow [95] and interacting Langevin's dynamics.

These gradient flows are based on choosing $\mathcal{E}$ as the Kullback–Leibler (KL) divergence

$$\mathcal{E}(\rho) = \mathrm{KL}[\rho \| \rho_{\mathrm{post}}] = \int \rho \log\Big(\frac{\rho}{\rho_{\mathrm{post}}}\Big)$$

with different metric g_ρ. See also [271] for a recent review on gradient flow for sampling.

Our work identifies canonical choices of $\mathcal{E}$ and g_ρ that result in favorable flow properties, contributing to the fundamental understanding of gradient flow for sampling. Specifically, we have achieved the following:

- We prove that, out of all f-divergences, the KL divergence is the only one that, up to scaling, has a first variation *invariant* to the normalization constant of the posterior distribution ρ_{post}. This property is highly desirable because it means the flow can be implemented without knowledge of the normalization constant. Our result demonstrates that this property is uniquely possessed by the KL divergence, validating its widespread use.
- We focus on the Fisher-Rao metric, which has been shown to be the only metric, up to scaling, that is *invariant* under any diffeomorphism of the state

space [40, 11, 24]. Along with other concurrent research [176], our work establishes the unconditional and uniformly exponential convergence of the Fisher-Rao gradient flow. This property is remarkable, as the convergence behavior of other gradient flows, such as the Wasserstein gradient flow, depends crucially on ρ_{post}, particularly its log-Sobolev constant. As a result, the Fisher-Rao gradient flow is ideal for sampling arbitrary posterior distributions, as it offers excellent convergence properties irrespective of the specific posterior distribution.

However, simulating the Fisher-Rao gradient flows requires extra effort. Particle methods based on birth-death dynamics have been employed in previous studies [175, 176], but their effectiveness depends on the quality of the density estimator for particle distributions. As a result, these methods may deteriorate when applied to high-dimensional problems, necessitating the use of many particles.

We investigate parametric approximations of the Fisher-Rao gradient flows. Specifically, we demonstrate the equivalence between the Gaussian projection of the flow and natural gradient methods in variational inference. Additionally, we explore the use of Kalman methodology to obtain a derivative-free approximation of the Fisher-Rao gradient flow, recovering a recently proposed Kalman-type sampler [132] that has demonstrated success in large-scale Bayes inverse problems. We will continue to work towards enhancing these approximations to create efficient and robust samplers for posterior distributions in Bayes inference.

1.2.6 Summary

In Section 1.2, we overview statistical numerical methods for PDEs and inverse problems covered in this thesis. By modeling solutions as Gaussian processes (GPs) and performing Bayes inference based on the PDE and observational data, one gets a rigorous and automatic solver with scientific machine learning flavors. Our works lay out the theoretical underpinning of the methodology, design efficient algorithms to scale up the computation with dense kernel matrices, analyze the use of hierarchical learning for enhanced adaptivity, and contribute to gradient flows based sampling algorithms for uncertainty quantification. By combining these components, we have developed a probabilistic pipeline for solving PDEs and inverse problems with Gaussian processes, which is particularly effective for problems in low-dimensional physical space and can also apply to high-dimensional problems very flexibly.

The GP-based method is a versatile approach that can be applied to general PDEs

and inverse problems at an algorithmic level. This complements the multiscale methods presented in Section 1.1, which are specialized to tackle difficult PDEs with rough coefficients. We note that, without a more targeted design of kernel functions, the GP-based method may not be able to handle such challenging PDEs with rough coefficients effectively.

For high-dimensional scientific computing problems, randomness can help identify potential low-dimensional structures. In kernel computations, this involves constructing an efficient and accurate low-rank approximation of dense kernel matrices. We show that using the randomly pivoted Cholesky algorithm achieves a favorable balance between exploration and exploitation in high-dimensional space.

Overall, the second part of this thesis demonstrates the efficacy of using probabilistic and statistical approaches to address scientific computing problems in an automated, efficient, and robust manner.

Chapter 2

EXPONENTIALLY CONVERGENT MULTISCALE FINITE ELEMENT METHOD

In this chapter, we present the exponentially convergent multiscale methods (ExpMsFEM) that we have proposed in a series of works [46, 48, 47]. Specifically, our discussion centers on solving general high-frequency Helmholtz equations, and is based on the work [48] (to appear in *SIAM Multiscale Modeling & Simulation*).

2.1 Introduction

We focus on solving the Helmholtz equation in heterogeneous media and high frequency regimes. We consider the model problem in a bounded polygonal domain $\Omega \subset \mathbb{R}^d$ with a Lipschitz boundary Γ. For generality, the boundary can contain three disjoint parts $\Gamma = \Gamma_D \cup \Gamma_N \cup \Gamma_R$ where Γ_D, Γ_N and Γ_R correspond to the Dirichlet, Neumann and Robin type conditions, respectively. Given positive constants $A_{\min}$, $A_{\max}$, $\beta_{\min}$, $\beta_{\max}$, $V_{\min}$, $V_{\max}$ and functions $A, \beta, V : \Omega \to \mathbb{R}$ that satisfy $A_{\min} \leq A(x) \leq A_{\max}, \beta_{\min} \leq \beta(x) \leq \beta_{\max}$ and $V_{\min} \leq V(x) \leq V_{\max}$, the Helmholtz equation with homogeneous boundary conditions[1] is formulated as follows:

$$\begin{cases} -\nabla \cdot (A\nabla u) - k^2 V^2 u = f, \text{ in } \Omega \\ u = 0, \text{ on } \Gamma_D \\ A\nabla u \cdot \nu = T_k u, \text{ on } \Gamma_N \cup \Gamma_R \,. \end{cases} \tag{2.1.1}$$

Here, ν is the outer normal to the boundary. The boundary operator satisfies $T_k u = 0$ for $x \in \Gamma_N$ and $T_k u = ik\beta u$ for $x \in \Gamma_R$, where i denotes the imaginary number. The wavenumber k is real and positive, and functions u and f are complex-valued. Our aim is to design a multiscale method for solving (2.1.1) that achieves a nearly exponential rate of convergence with respect to the computational degrees of freedom. This is a challenging problem due to combined difficulties of heterogeneity and high frequency. We review the related literature of this research field in Section 2.1.1 and discuss our methodology as well as its motivations and related work in Section 2.1.2.

[1] For simplicity of presentation, homogeneous boundary conditions are considered here. Generalization to non-homogeneous data is straightforward; see Section 2.5.3 or [46] (Section 5.3).

2.1.1 Literature for Solving Helmholtz Equations

The Helmholtz equation has been widely used in studying wave propagation in complex media. Numerical simulation of this equation still remains a challenging task, especially in the regime where the wavenumber k is large. The main numerical difficulty lies in the highly oscillatory pattern of the solution. Furthermore, the operator in the equation is indefinite, which leads to severe instability issues for standard numerical solvers such as the finite element method (FEM). Indeed, a well-known pre-asymptotic effect called the pollution effect [19] can occur—that is, in order to get a reasonably accurate solution, the mesh size H in the FEM needs to be much smaller than $1/k$. For example, for a standard P1-FEM approach, the mesh size needs to satisfy $H = O(1/k^2)$ for quasi-optimality of the solution [13, 19]. This constraint on H is much stronger than the typical condition in the approximation theory for representing an oscillatory function with frequency k, where $H = O(1/k)$, i.e., a fixed number of grid points per wavelength, would suffice for an accurate approximate solution.

In the literature, there have been many attempts to overcome or alleviate the difficulty associated with the pollution effect, so that a mesh size of $H = O(1/k)$ can be used. We highlight two classes of methods, namely the hp-FEM and multiscale methods, which can theoretically deal with the pollution effect under their respective model assumptions. The hp-FEM is proposed in [185, 186], which is a FEM using local high order polynomials. It is shown that by choosing the degrees of local polynomials $p = O(\log k)$, the pollution effect can be suppressed in principle for the Helmholtz equation with constant A, V and β. Nevertheless, to the best of our knowledge, there have been no theoretical results for this methodology when these coefficients become rough. There have been some recent developments for hp-FEM methods when piecewise regularity of the coefficients is assumed [26, 154]. In general, it is well-known that polynomials might behave arbitrarily badly even for elliptic equations with rough coefficients [17].

Multiscale methods, on the other hand, have long been developed to address the difficulty associated with rough coefficients in elliptic equations. In particular, we mention the LOD and Gamblets related approaches [181, 121, 210, 203, 204, 44, 45], variants of the multiscale finite element method (MsFEM) [129, 74, 128, 52, 89, 46] and generalized finite element methods based on partition of unity methods (PUM) [15, 253, 35, 41, 16, 243, 178, 179], which are most related to our work. Recently, the LOD method has been generalized to the case of Helmholtz equations with high

wavenumber and heterogeneous media [214, 93, 34, 215]. They show that with a coarse mesh of size $O(H)$ and localized multiscale basis functions of support size $O(H\log(1/H)\log k)$, the pollution effect can be overcome once the stability constant of the solution operator of the Helmholtz equation is of at most polynomial growth. An error of at most $O(H)$ is established. Very recently, there is also a super-localized version of LOD-type method for the Helmholtz equations, proposed in [86], where the support of basis functions is further reduced to $O(H\log^{(d-1)/d}(k/H))$.

From the perspective of MsFEM methodology, the authors in [91] introduce WMsFEM to address the pollution effect successfully. Their basis functions are all of local support size $O(H)$. On the theoretical side, they require $O(k)$ number of basis functions in each element in order to achieve $O(H)$ accuracy. In contrast, our method, which can be viewed as a generalization of MsFEM, only requires $O(\log^{d+1} k)$ number of basis function of support size $O(H)$ in each element to handle the pollution effect and to achieve $O(H)$ accuracy. More importantly, our method yields an overall exponential rate of convergence regarding the number of basis functions, thanks to a systematic decomposition and treatment of coarse and fine scale parts of the solution.

In the literature, multiscale methods with exponential convergence for elliptic equations with rough media first appeared in [15], which is based on local optimal basis approximation combined with the partition of unity method (PUM). There has been a number of recent papers that are actively working on improving the methodology [253, 35, 41, 16, 243, 178, 179], aiming for more refined continuous and discrete analysis, randomized computation, efficient implementation, and generalization beyond elliptic equations. Our initial work [46] on exponentially convergent multiscale methods for elliptic equations draws many motivations from these results, especially the Caccioppoli-type inequality that is essential for proving the exponential convergence. Different from the PUM based approach, our method is based on non-overlapped domain decomposition. More comparisons will be discussed in Subsection 2.1.2. In a concurrent work [177], the authors also proposed an exponentially convergent method for the Helmholtz equations using the PUM-based optimal local approximation methodology.

In addition to those methods mentioned above, there have also been several algorithms based on the MsFEM methodology [198, 90] or the HMM methodology [199] with particular empirical success for solving the Helmholtz equation. It is also worth noting that, in conjunction with designing a good discretization scheme

as above, one could also consider fast solvers for the discrete linear system. See, for example, the method of sweeping preconditioners [77, 78, 218], where a preconditioning matrix is constructed to compute approximations of the Schur complements successively. Very recently, the LOD approach has also been combined with the hierarchical approach of Gamblets [116] to get a multiresolution solver for the discrete system.

2.1.2 Main Contributions and Motivations

We propose a multiscale framework for solving the Helmholtz equation in rough media and high frequency regimes, specifically in dimension $d = 2$ where the mesh geometry of the non-overlapped domain decomposition is simplest to describe. Our idea is based on a multiscale method in our previous work [46] for solving elliptic equations with rough coefficients in an exponentially convergent manner. We aim to extend this framework to the more challenging Helmholtz equation where the operator is non-Hermitian and indefinite. It is perhaps surprising that the techniques in multiscale methods for elliptic equations can be systematically adapted to the Helmholtz equation. Indeed, it has been proved in [79] that the Green function of the Helmholtz equations requires a polynomial in k number of degrees of freedom to approximate, where they consider basis functions independent of the right hand side. Here, our results demonstrate that one can actually compress the solution operator exponentially efficiently by adding a number of local basis functions that depend on the local information of the right hand side. This shows that one can still achieve significant compression of the high frequency Helmholtz solution operator with rough coefficients by developing a data-driven compression operator adapted to the right hand side.

We outline our main contributions:

1. In studying the solution behavior of the Helmholtz equation (2.1.1), we introduce a coarse-fine scale decomposition of its solution space. This decomposition is adapted to the coarse mesh structure; a mesh size of $O(1/k)$ suffices to make this coarse-fine scale decomposition well defined. Moreover, the decomposition is adapted to the coefficients A, V, β and the wavenumber k.

2. Analytically, we show the fine scale part is of $O(H)$ in the energy norm, and it can be computed efficiently by solving the Helmholtz equations locally. Meanwhile, we prove that the space of the coarse scale part is of low complexity, such that there exist local multiscale basis functions that can approximate

this part in a nearly exponentially convergent manner. These serve as the cornerstone of our multiscale numerical method.

3. Numerically, we propose a multiscale framework that solves the two parts separately. The nearly exponential rate of convergence in the energy norm and L^2 norm is theoretically proved.

4. Experimentally, we conduct a number of numerical tests and observe consistently that our multiscale methods give a nearly exponential rate of convergence, even for problems with high-contrast media. Based on these numerical studies, several recommendations for efficient implementations of our methods are provided, especially on how to design the offline and online computation to handle multiple right hand sides efficiently.

To the best of our knowledge, this multiscale framework is the first one that can be proved rigorously to achieve a nearly exponential rate of convergence in solving (2.1.1) with rough A, β, V and large k, especially for $d = 2$. It generalizes our previous work on exponential convergence for solving rough elliptic equations [46], which is motivated by the PUM approach using optimal local approximation spaces for elliptic equations [15].

Different from the PUM that uses an overlapped domain decomposition, our method relies on non-overlapped domain decomposition and an edge coupling approach to combine local basis functions as in MsFEM. Our coarse-fine scale decomposition of the solution space is built on this non-overlapped edge coupling. For elliptic equations, this decomposition is the same as the orthogonal decomposition in previous work of MsFEM [128, 46] and approximate component mode synthesis [123, 122]. Under this line of methodology, we contribute a principled framework for obtaining nearly exponentially convergent basis functions for multiscale Helmholtz equations.

There are many differences between the multiscale methods based on PUM and edge coupling. Basically, the support of basis functions in PUMs is usually larger than that of MsFEMs since non-overlapped domain decomposition leads to smaller decomposed domains than its overlapped counterpart. There is no need to introduce additional freedom of partition of unity functions as well. On the other hand, in 2D, the number of local edges could be twice as many as the number of local domains, leading to more work in constructing the basis functions. Moreover, there will be an increasing design complexity for the non-overlapped edge coupling approach

for higher-dimensional problems since the boundaries of high dimensional local domains become more complicated. This is why we dedicate specifically to 2D Helmholtz equations for detailed analysis and numerical experiments.

We are not going to dive very deeply about the fundamental comparison between overlapped and non-overlapped decomposition in multiscale methods. Our aim is to demonstrate that one could achieve a nearly exponential convergence rate theoretically using the non-overlapped edge coupling framework in a principled way and show that this method is very competitive numerically. A number of technical difficulties, such as the appropriate approximation space for the edge functions and the spectral analysis of the local restriction operator, are carefully addressed to lay out this framework. We believe this work could help future researchers understand and analyze multiscale methods that are built on different local decomposition and global coupling approaches.

Lastly, we remark that in principle, our multiscale algorithm can be applied to general Helmholtz equations numerically, while most of our theoretical results rely on analytical properties of the solution to equation (2.1.1), related to the well-posedness, stability and C^α estimates. Therefore, typical conditions (usually very mild) of these analytical properties will be assumed here, in order to get a rigorous theory. We will mention several references to these results. Some numerical examples in which these assumptions are violated will be also presented to illustrate the effectiveness of our algorithm in a general context.

2.1.3 Organization

The rest of this chapter is organized as follows. In Section 2.2, we review preliminary results for the Helmholtz equation, including the well-posedness, stability, adjoint problems, and Hölder C^α estimates. Section 2.3 is devoted to analyzing the solution space based on a coarse-fine scale decomposition. Moreover, the computational properties of the coarse and fine parts are rigorously studied in detail. Building upon these properties, in Section 2.4 we develop the multiscale computational framework and prove the nearly exponential rate of convergence for our multiscale methods. The detailed numerical algorithms are discussed and implemented in Section 2.5 for several Helmholtz equations. To improve the readability, some technical proofs of theorems and propositions will be deferred to Section 2.6. Some concluding remarks are made in Section 2.7.

2.2 Preliminaries on Helmholtz's Equation

Our multiscale algorithm relies on an in-depth understanding of the solution space of (2.1.1). To achieve this, we first present several analytic results for (2.1.1), which will serve as preliminaries for our subsequent discussions. We cover the weak formulation, the well-posedness of the equation, the stability estimates of the solution, and Hölder estimates.

2.2.1 Notations

We use $H^1(\Omega)$ to denote the standard complex Sobolev space in Ω, containing L^2 functions with L^2 first order derivatives. We write $(u, v)_D := \int_D u\bar{v}$ for any domain D. We use C as a generic constant, and its value can change from place to place; we will state explicitly the parameters that this constant may or may not depend on.

2.2.2 Analytic Results

For the model problem (2.1.1), we consider the complex Sobolev space $\mathcal{H}(\Omega) := \{u \in H^1(\Omega) : u|_{\Gamma_D} = 0\}$ in which functions have zero trace on the Dirichlet boundary. This space is equipped with the norm $\|\cdot\|_{\mathcal{H}(\Omega)}$ such that

$$\|u\|_{\mathcal{H}(\Omega)} := \int_\Omega A|\nabla u|^2 + k^2V^2|u|^2 \,.$$

The dual space of $\mathcal{H}(\Omega)$ is denoted by $\mathcal{H}^{-1}(\Omega)$ equipped with the norm $\|\cdot\|_{\mathcal{H}^{-1}(\Omega)}$; by definition one has

$$\|f\|_{\mathcal{H}^{-1}(\Omega)} := \sup_{v\in\mathcal{H}(\Omega)} \frac{|(f, v)_\Omega|}{\|v\|_{\mathcal{H}(\Omega)}} \,.$$

Now, we present several analytic results pertaining to the Helmholtz equation (2.1.1).

Weak formulation. The weak formulation of (2.1.1) is given by

$$a(u, v) := (A\nabla u, \nabla v)_\Omega - k^2(V^2u, v)_\Omega - (T_k u, v)_{\Gamma_N\cup\Gamma_R} = (f, v)_\Omega, \quad \forall v \in \mathcal{H}(\Omega) \,. \tag{2.2.1}$$

Continuity estimate. By the Cauchy-Schwarz and trace inequalities (see Lemma 3.1 of [185]), the sesquilinear form $a(\cdot,\cdot)$ is bounded on $\mathcal{H}(\Omega)$ with a constant C_c independent of k, i.e., for any $u, v \in \mathcal{H}(\Omega)$, one has the continuity estimate:

$$|a(u, v)| \le C_c\|u\|_{\mathcal{H}(\Omega)}\|v\|_{\mathcal{H}(\Omega)} \,. \tag{2.2.2}$$

Well-posedness and stability. If Γ_R has positive $d-1$ dimensional measure, then under some mild conditions (see Assumption 2.3 and Theorem 2.4 in [103]), problem

(2.2.1) admits a unique solution given the right hand side $f \in L^2(\Omega)$. We will assume these conditions. Let the solution operator be N_k, so that $u = N_k f$. Under the same conditions, this operator is stable (Theorem 2.4 in [103]) in the sense that

$$C_{\text{stab}}(k) := \sup_{f \in L^2(\Omega)\setminus\{0\}} \frac{\|N_k f\|_{\mathcal{H}(\Omega)}}{\|f\|_{L^2(\Omega)}} < \infty\,. \tag{2.2.3}$$

To avoid getting into detailed discussions of these assumptions and for simplicity of presentation, we will base most of our arguments on assuming (2.2.3) holds.

The stability constant $C_{\text{stab}}(k)$ will depend on k in general, and obtaining an explicit characterization of this dependence has been a hard task; see [27, 34, 104, 190, 238]. A prevalent and reasonable assumption on the constant is that of polynomial growth, namely $C_{\text{stab}}(k) \leq C(1 + k^\gamma)$ for some constants γ and C; see for example [153]. We are not going into detailed discussions on this assumption here, while we mention that the final error estimate of our numerical solution will depend on $C_{\text{stab}}(k)$ explicitly; thus, those estimates on $C_{\text{stab}}(k)$ in the literature can be readily applied to our context.

In addition, stability for $f \in L^2(\Omega)$ can yield well-posedness and stability for $f \in \mathcal{H}^{-1}(\Omega)$. According to Lemma 2.1 in [214] and also [82], one has

$$\sup_{f \in \mathcal{H}^{-1}(\Omega)\setminus\{0\}} \frac{\|N_k f\|_{\mathcal{H}(\Omega)}}{\|f\|_{\mathcal{H}^{-1}(\Omega)}} \leq k C_{\text{stab}}(k)\,. \tag{2.2.4}$$

Adjoint problems. Due to the presence of the Robin boundary condition, $a(\cdot,\cdot)$ is not Hermitian. Its adjoint sesquilinear form is defined as $a^*(u,v) = \overline{a(v,u)}$. The adjoint problem for (2.2.1) is given by $a^*(u,v) = (f,v)_\Omega$ for any $v \in \mathcal{H}(\Omega)$. It also corresponds to the following PDE:

$$\begin{cases} -\nabla\cdot(A\nabla u) - k^2 V^2 u = f, \text{ on } \Omega \\ u = 0, \text{ in } \Gamma_D \\ A\nabla u\cdot\nu = T_k^* u, \text{ on } \Gamma_N \cup \Gamma_R\,, \end{cases}$$

where $T_k^* u := \overline{T_k \overline{u}} = -T_k u$. The adjoint solution operator is denoted by N_k^*. One can readily check that $N_k^* \overline{f} = \overline{N_k f}$. Therefore, the adjoint problem admits the same stability constant as the original problem; namely it holds

$$C_{\text{stab}}(k) = \sup_{f \in L^2(\Omega)\setminus\{0\}} \frac{\left\|N_k^\star f\right\|_{\mathcal{H}(\Omega)}}{\|f\|_{L^2(\Omega)}} < \infty\,.$$

The adjoint problem will play a valuable role when we analyze the convergence property of our multiscale methods for the Helmholtz equation.

C^α Hölder regularity. We will need the C^α estimates of the solution in order to demonstrate the theoretical properties of our multiscale methods.

Proposition 2.2.1. *Suppose $d \leq 3$ and (2.2.3) holds. If $f \in L^2(\Omega)$, then the solution $u \in C^\alpha(\Omega)$ for some $\alpha \in (0, 1)$.*

We defer the proof of this proposition to Subsection 2.6.1.

Remark 2.2.2. *The global regularity estimate may depend on the wavenumber k. Nevertheless, we only use it to show qualitatively that our solution is continuous, so that the nodal interpolation in Subsection 2.3.4.2 is mathematically rigorous. Later, when we derive error estimates of our methods, we will only use the local version of the regularity estimate, where the constant is independent of the wavenumber; see Lemma 2.6.2.*

We have presented several critical analytic results for the Helmholtz equation. Based on these results, we are now ready to study the solution space of (2.1.1) in the next section. The key is a coarse-fine scale decomposition of the solution space, which will play an essential role in designing our multiscale algorithms.

2.3 Coarse-Fine Scale Decomposition

In this section, we develop a coarse-fine scale operator-adapted decomposition of the solution space. This decomposition is adaptive to the mesh structure, and a mesh of size $H = O(1/k)$ suffices to make this coarse-fine scale decomposition well defined. We discuss the setting of the mesh structure in Subsection 2.3.1, followed by introducing the coarse-fine scale decomposition in Subsection 2.3.2. In Subsection 2.3.3 we show the fine scale part is local and small up to $O(H)$ in the $\mathcal{H}(\Omega)$ norm. In Subsection 2.3.4 we show the coarse-scale component can be approximated via local edge basis functions in a nearly exponentially convergent manner.

2.3.1 Mesh Structure

We begin by discussing related concepts of the mesh structure. The focus here is on $d = 2$. In the mesh structure, we discuss two dimensional elements in Subsection 2.3.1.1, one dimensional edges and zero dimensional nodes, and their neighborship in Subsection 2.3.1.2. See also Figure 2.1 for illustrations.

2.3.1.1 Elements

We consider a shape regular and uniform partition of the domain Ω into finite elements, such as triangles and quadrilaterals. The collection of elements is denoted by $\mathcal{T}_H = \{T_1, T_2, ..., T_r\}$. For simplicity, we assume that each connected component of the domain is at least partitioned into two elements.

The mesh size is H, i.e., $\max_{T\in\mathcal{T}_H} \text{diam}(T) = H$. The uniformity of the mesh implies $\min_{T\in\mathcal{T}_H} \text{diam}(T) \geq c_0 H$ for some $0 < c_0 \leq 1$ that is independent of H and T. The shape regularity property implies there is a constant $c_1 > 0$ independent of H and T, such that $\max_{T\in\mathcal{T}_H} \text{diam}(T)^d/|T| \leq c_1$, where $|T|$ is the volume of T.

In this mesh, by using a scaling argument, the following Poincaré inequality will hold uniformly for $T \in \mathcal{T}_H$. This inequality will be used frequently later.

Proposition 2.3.1 (The Poincaré inequality)**.** *For any $T \in \mathcal{T}_H$ and a function $v \in H^1(T)$ that vanishes on one of the edges of T, it holds that*

$$\|v\|_{L^2(T)} \leq C_P H \|\nabla v\|_{L^2(T)}, \tag{2.3.1}$$

where C_P depends on c_0, c_1 and d.

2.3.1.2 Nodes, Edges, and Their Neighbors

Let $\mathcal{N}_H = \{x_1, x_2, ..., x_p\}$ be the collection of interior nodes, and $\mathcal{E}_H = \{e_1, e_2, ..., e_q\}$ be the collection of edges except those fully on the boundary of Ω. An edge $e \in \mathcal{E}_H$ is defined such that there exists two different elements T_i, T_j with $e = \overline{T}_i \bigcap \overline{T}_j$ that has co-dimension 1 in $\mathbb{R}^d$. We will use $E_H = \bigcup_{e\in\mathcal{E}_H} e \subset \Omega$ to denote the edges as a whole set.

We use the symbol $\sim$ to describe the neighbourship between nodes, edges, and elements. More precisely, if we consider a node $x \in \mathcal{N}_H$, an edge $e \in \mathcal{E}_H$, and an element $T \in \mathcal{T}_H$, then, (1) $x \sim e$ denotes $x \in e$; (2) $e \sim T$ denotes $e \subset \overline{T}$; (3) $x \sim T$ denotes $x \in \overline{T}$. The relationship $\sim$ is symmetric.

We use $\text{N}(\cdot,\cdot)$ to describe the union of neighbors as a set. For example, $\text{N}(x, \mathcal{E}_H) = \bigcup\{e \in \mathcal{E}_H : e \sim x\} \subset E_H$, $\text{N}(x, \mathcal{T}_H) = \bigcup\{T \in \mathcal{T}_H : T \sim x\} \subset \Omega$, and $\text{N}(e, \mathcal{T}_H) = \bigcup\{T \in \mathcal{T}_H : T \sim e\} \subset \Omega$.

2.3.2 Decomposition of Solution Space

With the mesh structure defined, we now discuss the coarse-fine scale decomposition of the solution space. We first discuss decomposition in the local element T in

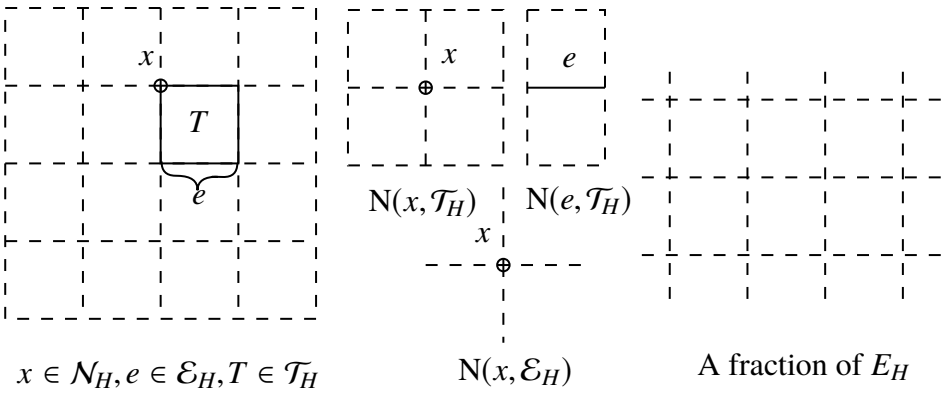


Figure 2.1: Geometry of the mesh

Subsection 2.3.2.1 and then the global decomposition in Subsection 2.3.2.2.

2.3.2.1 Local decomposition

A crucial requirement for the decomposition to be well defined is that the mesh size is order $O(1/k)$; see Assumption 2.3.2. As we will see later, this bound on H ensures that local Helmholtz problems in each element have properties that are similar to those of elliptic problems; thus, techniques in elliptic equations can then be applied.

Assumption 2.3.2. *The mesh size satisfies $H \leq A_{\min}^{1/2}/(\sqrt{2}C_P V_{\max} k)$, where C_P is the constant in Proposition 2.3.1.*

Given Assumption 2.3.2, we decompose[2] u into two parts $u = u_T^{\mathsf{h}} + u_T^{\mathsf{b}}$ in each element $T \in \mathcal{T}_H$. The two components satisfy:

$$\begin{cases} -\nabla\cdot(A\nabla u_T^{\mathsf{h}}) - k^2 V^2 u_T^{\mathsf{h}} = 0, \text{ in } T \\ \qquad\qquad u_T^{\mathsf{h}} = u, \text{ on } \partial T \setminus (\Gamma_N \cup \Gamma_R) \\ \qquad A\nabla u_T^{\mathsf{h}}\cdot\nu = T_k u_T^{\mathsf{h}}, \text{ on } \partial T \cap (\Gamma_N \cup \Gamma_R), \end{cases} \\ \begin{cases} -\nabla\cdot(A\nabla u_T^{\mathsf{b}}) - k^2 V^2 u_T^{\mathsf{b}} = f, \text{ in } T \\ \qquad\qquad u_T^{\mathsf{b}} = 0, \text{ on } \partial T \setminus (\Gamma_N \cup \Gamma_R) \\ \qquad A\nabla u_T^{\mathsf{b}}\cdot\nu = T_k u_T^{\mathsf{b}}, \text{ on } \partial T \cap (\Gamma_N \cup \Gamma_R). \end{cases} \tag{2.3.2}$$

In short, the part u_T^{h} incorporates the boundary value of u, while u_T^{b} contains information of the right hand side. Both equations in (2.3.2) should be understood

[2]This decomposition is inspired by that in the elliptic case [46].

in the standard weak sense using the following local sesquilinear form $a_T(\cdot,\cdot)$ in T:

$$a_T(v,w) := (A\nabla v, \nabla w)_T - k^2(V^2 v, w)_T - (T_k v, w)_{\partial T \cap (\Gamma_N \cup \Gamma_R)} \text{ for } v, w \in \mathcal{H}(T), \tag{2.3.3}$$

where $\mathcal{H}(T) := \mathcal{H}(\Omega)|_T$, the restriction of $\mathcal{H}(\Omega)$ in the domain T. The well-posedness of the two problems is due to the following proposition:

Proposition 2.3.3. *Under Assumption 2.3.2, for $v \in \mathcal{H}(T)$ that vanishes on one of the edges of T, the corresponding sesquilinear form is coercive such that*

$$\operatorname{Re} a_T(v,v) \geq \frac{1}{2}\|A^{1/2}\nabla v\|^2_{L^2(T)}.$$

Proof. Using the Poincaré inequality (2.3.1) and Assumption 2.3.2, we get

$$\begin{aligned} \operatorname{Re} a_T(v,v) &= \|A^{1/2}\nabla v\|^2_{L^2(T)} - \|kVv\|^2_{L^2(T)} \\ &\geq (1 - C_P^2 H^2 k^2 V_{\max}^2 A_{\min}^{-1})\|A^{1/2}\nabla v\|^2_{L^2(T)} \geq \frac{1}{2}\|A^{1/2}\nabla v\|^2_{L^2(T)}. \end{aligned} \tag{2.3.4}$$

□

Since both equations in (2.3.2) contain Dirichlet's boundary condition on at least one of the edges of T, the coercivity implied by Proposition 2.3.3 suffices for the well-posedness. Consequently, the solutions u_T^{h} and u_T^{b} are well-defined.

Remark 2.3.4. *An important property is that u_T^{h} is "left-orthogonal" to u_T^{b} in T with respect to the local sesquilinear form $a_T(\cdot,\cdot)$ in T, in the sense of $a_T(u_T^{\mathsf{h}}, u_T^{\mathsf{b}}) = 0$, according to the weak form of the equation. Note that we might not have $a_T(u_T^{\mathsf{b}}, u_T^{\mathsf{h}}) = 0$ for T near the boundary (i.e., $\partial T \cap (\Gamma_N \cup \Gamma_R) \neq \emptyset$) due to the fact that $a_T(\cdot,\cdot)$ is not Hermitian here.*

2.3.2.2 Global decomposition

In this subsection, we define a global decomposition $u = u^{\mathsf{b}} + u^{\mathsf{h}}$, such that for each T, it holds that $u^{\mathsf{h}}(x) = u_T^{\mathsf{h}}(x)$ and $u^{\mathsf{b}}(x) = u_T^{\mathsf{b}}(x)$ when $x \in T$. Both u^{h} and u^{b} are well-defined and belong to $\mathcal{H}(\Omega)$ due to the continuity across edges. Here, the component u_T^{h} (resp. u^{h}) is called the local (resp. global) *Helmholtz-harmonic part* and u_T^{b} (resp. u^{b}) is the local (resp. global) *bubble part*, of the solution u.

We further introduce the function space for the Helmholtz-harmonic part

$$\begin{aligned} V^{\mathsf{h}} := \{v \in \mathcal{H}(\Omega) : &-\nabla\cdot(A\nabla v) - k^2 V^2 v = 0 \text{ in each } T \in \mathcal{T}_H, \\ &A\nabla v \cdot \nu = T_k v, \text{ on } \Gamma_N \cup \Gamma_R\}, \end{aligned} \tag{2.3.5}$$

so that $u^{\mathsf{h}} \in V^{\mathsf{h}}$, and the space for the bubble part

$$V^{\mathsf{b}} := \{v \in \mathcal{H}(\Omega) : v = 0 \text{ on } E_H\}, \tag{2.3.6}$$

such that $u^{\mathsf{b}} \in V^{\mathsf{b}}$. In this way, the solution space of (2.1.1) can be decomposed to $V^{\mathsf{h}} + V^{\mathsf{b}}$. Furthermore, for any $v \in V^{\mathsf{h}}$ and $w \in V^{\mathsf{b}}$, it holds that $a(v, w) = 0$ by summing up local sesquilinear forms $a_T(\cdot, \cdot)$ and using Remark 2.3.4.

We will treat V^{b} as the fine scale or microscopic space, and refer to V^{h} as the coarse scale or macroscopic space. The idea of our multiscale framework is to compute the two parts separately by exploring their own structures.

In the next two subsections, we will study the computational properties of $u^{\mathsf{h}} \in V^{\mathsf{h}}$ and $u^{\mathsf{b}} \in V^{\mathsf{b}}$, respectively. These properties serve as the cornerstone of designing our multiscale algorithm.

2.3.3 Local and Small Bubble Part

In this subsection, we analyze the bubble part u^{b}. This part depends locally on f in each T. Thus, it can be computed efficiently in a parallel manner. Moreover, it is small and can be ignored if the target accuracy is $O(H)$; see Proposition 2.3.5.

Proposition 2.3.5. *Under Assumption 2.3.2, it holds that*

$$\left\|u^{\mathsf{b}}\right\|_{\mathcal{H}(\Omega)} \leq \frac{3C_P}{A_{\min}^{1/2}} H \|f\|_{L^2(\Omega)}. \tag{2.3.7}$$

Proof. By definition, inside each patch T, it holds that $a_T(u^{\mathsf{b}}, u^{\mathsf{b}}) = (f, u^{\mathsf{b}})_T$. The coercivity estimate in (2.3.4) implies the inequality $\|kVu^{\mathsf{b}}\|^2_{L^2(T)} \leq \frac{1}{2}\|A^{1/2}\nabla u^{\mathsf{b}}\|^2_{L^2(T)}$. Using the estimate, we get

$$\begin{aligned}\operatorname{Re} a_T(u^{\mathsf{b}}, u^{\mathsf{b}}) &= \|A^{1/2}\nabla u^{\mathsf{b}}\|^2_{L^2(T)} - \|kVu^{\mathsf{b}}\|^2_{L^2(T)} \\ &\geq \frac{1}{3}(\|A^{1/2}\nabla u^{\mathsf{b}}\|^2_{L^2(T)} + \|kVu^{\mathsf{b}}\|^2_{L^2(T)}) = \frac{1}{3}\|u^{\mathsf{b}}\|^2_{\mathcal{H}(T)}.\end{aligned}$$

Combining the above estimate with the Cauchy-Schwarz inequality, we arrive at

$$\|u^{\mathsf{b}}\|^2_{\mathcal{H}(T)} \leq 3 \operatorname{Re} a_T(u^{\mathsf{b}}, u^{\mathsf{b}}) = 3(f, u^{\mathsf{b}})_T \leq 3\|f\|_{L^2(T)}\|u^{\mathsf{b}}\|_{L^2(T)}.$$

Meanwhile, by the Poincaré inequality (2.3.1), we get

$$\|u^{\mathsf{b}}\|_{L^2(T)} \leq C_P H \|\nabla u^{\mathsf{b}}\|_{L^2(T)} \leq \frac{C_P H}{A_{\min}^{1/2}}\|u^{\mathsf{b}}\|_{\mathcal{H}(T)}.$$

Combining all the above inequalities gives $\|u^{\mathsf{b}}\|_{\mathcal{H}(T)} \leq 3(C_P H / A_{\min}^{1/2})\|f\|_{L^2(T)}$ for each element T. Summing them up for all elements T yields the desired conclusion.

□

2.3.4 Low Complexity of the Helmholtz-Harmonic Part

Now, we turn to the study of the Helmholtz-harmonic part u^{h}. The goal is to show that u^{h} can be approximated via local basis functions in an exponentially efficient manner. To achieve this, our approximation framework[3] contains three steps: (1) reducing the approximation of u^{h} to that of edge functions in Subsection 2.3.4.1, (2) localizing the approximation to every single edge in Subsection 2.3.4.2, and (3) realizing local approximation via oversampling and SVD in Subsection 2.3.4.3. Combining all these three steps, we establish the low complexity in approximation of u^{h} in Subsection 2.3.4.4.

2.3.4.1 Approximation via Edge Functions

We start with the first step of approximating u^{h}. By definition, u^{h} belongs to V^{h}. A key observation is that any function in V^{h} is determined entirely by its value on the edge set E_H. Thus, define

$$\tilde{V}^{\mathsf{h}} := \{\tilde{\psi} : E_H \to \mathbb{R}, \text{ there exists a function } \psi \in V^{\mathsf{h}}, \text{ such that } \tilde{\psi} = \psi|_{E_H}\};$$

then under Assumption 2.3.2, there is a one to one correspondence $\tilde{\psi} \in \tilde{V}^{\mathsf{h}} \leftrightarrow \psi \in V^{\mathsf{h}}$. More precisely, in each T, it holds that

$$\begin{cases} -\nabla \cdot (A\nabla\psi) - k^2 V^2 \psi = 0, & \text{in } T \\ \psi = \tilde{\psi}, & \text{on } \partial T \setminus (\Gamma_N \cup \Gamma_R) \\ A\nabla\psi \cdot \nu = T_k \psi, & \text{on } \partial T \cap (\Gamma_N \cup \Gamma_R). \end{cases} \tag{2.3.8}$$

Indeed, we have $\tilde{V}^{\mathsf{h}} = H^{1/2}(E_H)$ by the trace theory since the local equation is elliptic. Using the above identification, approximating u^{h} corresponds to approximating $\tilde{u}^{\mathsf{h}}$, which is a function defined on edges and of lower complexity. We need to pay attention to the norm we use when approximating $\tilde{u}^{\mathsf{h}}$ so that we can use the error bound of the approximation to control the error of u^{h} in the energy norm. This will be the focus of the next section.

Remark 2.3.6. *In the remaining part of this chapter, we will frequently use the correspondence between V^{h} and $\tilde{V}^{\mathsf{h}}$. Conventionally, when we write a tilde on the top of a function in V^{h}, it refers to its corresponding part in $\tilde{V}^{\mathsf{h}}$.*

[3] It is similar to that in our previous work for elliptic equations [46].

2.3.4.2 Localization of Approximation

We discuss how to approximate the edge function $\tilde{u}^{\mathsf{h}}$, whose domain is E_H, which is nonlocal. Since it is often preferable to have localized basis functions for approximation and numerical algorithms, our second step is to localize the task of approximating $\tilde{u}^{\mathsf{h}}$ to every single edge.

To achieve localization, we study the geometry of the edge set E_H first. Observing that different edges only communicate with each other along their shared nodes, we can use nodal interpolation to localize the approximation. More precisely, we proceed with the following steps:

1. Interpolation: for each node $x_i \in \mathcal{N}_H$, choose $\tilde{\psi}_i$ to be the piecewise linear tent function on E_H, satisfying $\tilde{\psi}_i(x_j) = \delta_{ij}$ for each $x_j \in \mathcal{N}_H$. This defines an interpolation operator for $v \in V^{\mathsf{h}} \cap C(\overline{\Omega})$:

$$I_H v := \sum_{x_i \in \mathcal{N}_H} v(x_i)\psi_i(x)\,.$$

   Note that $\psi_i(x)$ is the same as the basis function constructed via the multiscale finite element method (MsFEM [129]). The interpolation residual $v - I_H v$ vanishes on each $x_i \in \mathcal{N}_H$. Set[4] $v = \tilde{u}^{\mathsf{h}}$ and let $I_H\tilde{u}^{\mathsf{h}}$ be one part of the approximation for $\tilde{u}^{\mathsf{h}}$. Then, it remains to approximate the residue $\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}$.

2. Localization: we wish to explore the fact that $\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}$ vanishes on nodes to localize the subsequent approximation task. To achieve so, define $R_e\tilde{u}^{\mathsf{h}} = P_e(\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}) := (\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}})|_e$. The goal is to find some basis functions on each e to approximate $R_e\tilde{u}^{\mathsf{h}}$. To make this problem precise, we need to specify the function space of $R_e\tilde{u}^{\mathsf{h}}$, and the norm for approximation.

   It turns out that the natural function space $R_e\tilde{u}^{\mathsf{h}}$ is the Lions-Magenes space; see the following Proposition 2.3.7.

Proposition 2.3.7. *Let $d = 2$. Suppose $f \in L^2(\Omega)$ and (2.2.3) holds. For each $e \in \mathcal{E}_H$, it holds that $R_e\tilde{u}^{\mathsf{h}} \in H_{00}^{1/2}(e)$, the Lions-Magenes space which contains functions $v \in H^{1/2}(e)$ such that*

$$\frac{v(x)}{\operatorname{dist}(x, \partial e)} \in L^2(e)\,.$$

Here $\operatorname{dist}(x, \partial e)$ *is the Euclidean distance from x to the boundary of e.*

[4] Note that we can apply I_H to $\tilde{u}^{\mathsf{h}}$ due to the C^α estimate of u in Proposition 2.2.1.

It might seem unclear at this stage why we should consider such a complicated function space. In fact, this is related to the zero extension of functions. According to Chapter 33 of [266], $H_{00}^{1/2}(e)$ can also be characterized as the space of functions in $H^{1/2}(e)$, such that their zero extensions to E_H is still in $H^{1/2}(E_H)$. This is the key and in fact the only property that we will use for $H_{00}^{1/2}(e)$. The zero extension allows us to connect local approximation and global approximation. In the following we will not distinguish $\tilde{\psi} \in H_{00}^{1/2}(e)$ and its zero extension to E_H that belongs to $H^{1/2}(E_H)$.

For any function in $H_{00}^{1/2}(e)$, we define a norm to measure approximation accuracy.

Definition 2.3.8. *Let $d = 2$. The $\mathcal{H}^{1/2}(e)$ norm of a function $\tilde{\psi} \in H_{00}^{1/2}(e)$ is defined as:*

$$\|\tilde{\psi}\|^2_{\mathcal{H}^{1/2}(e)} := \int_{\Omega} A|\nabla\psi|^2 + k^2|V\psi|^2\,, \tag{2.3.9}$$

where we have used the one to one correspondence $\tilde{\psi} \in \tilde{V}^{\mathsf{h}} \leftrightarrow \psi \in V^{\mathsf{h}}$. Here we identify $\tilde{\psi}$ as the zero extension of its value on the edge e to E_H.

The $\mathcal{H}^{1/2}(e)$ norm in Definition 2.3.8 is the natural one to consider here since eventually, we aim for approximation accuracy in the energy norm.

The following theorem is the cornerstone for the above localization strategy. It states that a local accuracy guarantee can be seamlessly coupled to form a global accuracy guarantee.

Theorem 2.3.9 (Global error estimate)**.** *Let $d = 2$. Suppose for each edge e, there exists an edge function $\tilde{v}_e \in H_{00}^{1/2}(e)$ that satisfies*

$$\|R_e\tilde{u}^{\mathsf{h}} - \tilde{v}_e\|_{\mathcal{H}^{1/2}(e)} \leq \epsilon_e\,. \tag{2.3.10}$$

Let $v_e \in V^{\mathsf{h}}$ be the corresponding part of $\tilde{v}_e \in \tilde{V}^{\mathsf{h}}$. Then, it holds that

$$\|u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\in\mathcal{E}_H} v_e\|^2_{\mathcal{H}(\Omega)} \leq C_{\mathrm{mesh}} \sum_{e\in\mathcal{E}_H} \epsilon_e^2\,, \tag{2.3.11}$$

where C_{mesh} is a constant depending on the number of edges for the elements only, e.g., for quadrilateral mesh $C_{\mathrm{mesh}} = 4$.

Given this theorem, to approximate u^{h} it suffices to find local edge basis functions that satisfy (2.3.10) for some desired ϵ_e. This is a localized task for each e.

The proofs for Propositions 2.3.7 and Theorem 2.3.9 are similar to that in the setting of elliptic equations [46]. However, for completeness, we will also present them here in Subsections 2.6.2 and 2.6.3.

2.3.4.3 Local Approximation via Oversampling

The last step of approximation is to find local edge basis functions for each e so that (2.3.10) is satisfied. In this subsection, we discuss how to achieve this via oversampling and SVD, which can yield exponentially decaying ϵ_e. The general idea is to explore the fact that for a coarse scale function, its behavior on e can be controlled very well by that in an oversampling domain due to the compactness property of the restriction operator.

More precisely, for a given edge e, consider an oversampling domain ω_e associated with the edge. In general, any domain containing e in the interior can serve as a candidate. Here, for simplicity of presentation and as an illustrative example, we set

$$\omega_e = \overline{\bigcup \{T \in \mathcal{T}_H : \overline{T} \cap e \neq \emptyset\}}\,. \tag{2.3.12}$$

For interior edges and edges connected to the boundary, an illustration of this choice (2.3.12) for a quadrilateral mesh is given in Figure 2.2.

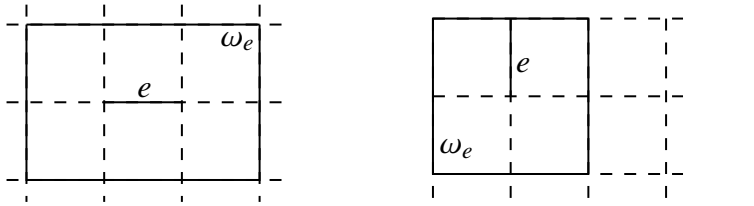


Figure 2.2: Illustration of oversampling domains. On the right, we use an edge connected to the upper boundary as an illustrating example.

The key idea is to treat the residue $R_e\tilde{u}^{\mathsf{h}}$ as a restriction of a coarse scale function in ω_e and explore the compactness property of such restriction operators. By an abuse of notation via the correspondence of V^{h} and $\tilde{V}^{\mathsf{h}}$, for any $\mathcal{H}(\Omega)$ in V, we identify $R_e v$ as $R_e\tilde{v}^{\mathsf{h}}$. As a first step, we write

$$R_e\tilde{u}^{\mathsf{h}} = R_e u = R_e u^{\mathsf{h}}_{\omega_e} + R_e u^{\mathsf{b}}_{\omega_e}\,, \tag{2.3.13}$$

where we decompose u in ω_e into its coarse and fine scale components, via (2.3.2) with T replaced by ω_e, and we shall use $u^{\mathsf{h}}_{\omega_e}$ and $u^{\mathsf{b}}_{\omega_e}$ to denote the corresponding local Helmholtz-harmonic and bubble part respectively. Then, to approximate $R_e\tilde{u}^{\mathsf{h}}$, we could approximate the two terms in (2.3.13) separately. We will show that the first term can be approximated in an exponentially efficient manner due to a compactness property, and the second term can be computed locally and is very small.

Remark 2.3.10. *One may ask whether the decomposition* (2.3.13) *in the oversampling domain is still well-defined. Indeed, similar to* (2.3.1)*, we have a uniform Poincaré inequality for every ω_e: for any edge e and $H^1(\omega_e)$ function v vanishing on any one of the edge boundaries of ω_e, it holds that*

$$\|v\|_{L^2(\omega_e)} \leq C'_P H \|\nabla v\|_{L^2(\omega_e)}, \tag{2.3.14}$$

where C'_P is a constant that only depends on c_0, c_1, d and our choice of oversampling domain. For the particular choice (2.3.12)*, C'_P is a constant multiple of C_P; without loss of generality we assume $C'_P \geq C_P$. Based on this observation, we will choose a small H so that Assumption 2.3.11 holds, which guarantees that local Helmholtz operators in the oversampling domain behave in a manner similar to that of elliptic case; this is similar to Proposition 2.3.3.*

Assumption 2.3.11. *The mesh size satisfies $H \leq A_{\min}^{1/2}/(\sqrt{2}C'_P V_{\max} k)$, where C'_P is the constant in* (2.3.14)*.*

Note that Assumption 2.3.11 implies Assumption 2.3.2. Now, we discuss in detail how to deal with the two terms in (2.3.13).

1. For the first term, we consider the following function space in ω_e:

$$\begin{aligned} U(\omega_e) := \{ v \in \mathcal{H}(\omega_e) : &- \nabla \cdot (A\nabla v) - k^2 V^2 v = 0, \text{ in } \omega_e \\ &A\nabla v \cdot \nu = T_k v, \text{ on } (\Gamma_N \cup \Gamma_R) \cap \partial\omega_e \}. \end{aligned} \tag{2.3.15}$$

   Functions in this space are fully determined by their trace on $\partial\omega_e \backslash (\Gamma_N \cup \Gamma_R)$. By definition, $u^{\mathsf{h}}_{\omega_e}$ belongs to $U(\omega_e)$. Under Assumption 2.3.11, $(U(\omega_e), \|\cdot\|_{\mathcal{H}(\omega_e)})$ is a Hilbert space, since the Helmholtz operator in ω_e is elliptic. Then, by abuse of notation, consider the operator

$$R_e : (U(\omega_e), \|\cdot\|_{\mathcal{H}(\omega_e)}) \to (H_{00}^{1/2}(e), \|\cdot\|_{\mathcal{H}^{1/2}(e)}),$$

   such that $R_e v = P_e(v - I_H v)$ for $v \in U(\omega_e)$. A critical property is that the singular values of R_e decay nearly exponentially fast; see Theorem 2.3.12. Its proof is deferred to Subsection 2.6.4.

   Theorem 2.3.12. *Let d=2. Under Assumption 2.3.11, the operator R_e is compact for each $e \in \mathcal{E}_H$. Denote the pairs of its left singular vectors and singular values by $\{\tilde{v}_{m,e}, \lambda_{m,e}\}_{m\in\mathbb{N}}$, where $\tilde{v}_{m,e} \in H_{00}^{1/2}(e)$ and the sequence $\{\lambda_{m,e}\}_{m\in\mathbb{N}}$ is in a descending order. Then, for any $\epsilon > 0$, it holds that*

$$\lambda_{m,e} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right), \tag{2.3.16}$$

where C_ϵ is a constant that is independent of k, H and may depend on ϵ, d and the mesh parameters c_0, c_1.

Remark 2.3.13. *As we can see from the proof, we actually show that* (2.3.16) *still holds by setting C_ϵ to be* 1 *and requiring for $m > N_\epsilon$ with N_ϵ depending on k and H. But we can also make the above inequality hold for all m by introducing the constant C_ϵ.*

We discuss the implication of this theorem. By definition of singular values, if we set $W_{m,e} = \text{span}\,\{\tilde{v}_{j,e}\}_{j=1}^{m-1}$, then Theorem 2.3.12 implies that

$$\min_{\tilde{v}_e \in W_{m,e}} \|R_e v - \tilde{v}_e\|_{\mathcal{H}^{1/2}(e)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|v\|_{\mathcal{H}(\omega_e)}\,. \tag{2.3.17}$$

Applying this result to $v = u^{\mathsf{h}}_{\omega_e} \in U(\omega_e)$ leads to

$$\min_{\tilde{v}_e \in W_{m,e}} \|R_e u^{\mathsf{h}}_{\omega_e} - \tilde{v}_e\|_{\mathcal{H}^{1/2}(e)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|u^{\mathsf{h}}_{\omega_e}\|_{\mathcal{H}(\omega_e)}\,. \tag{2.3.18}$$

Thus, there is a nearly exponential efficiency in approximating the first term $R_e u^{\mathsf{h}}_{\omega_e}$.

2. For the second term in (2.3.13), the oversampling bubble part $u^{\mathsf{b}}_{\omega_e}$ can be efficiently computed by solving local Helmholtz problems. Moreover, under Assumption 2.3.11 this term is small in the $\mathcal{H}(\Omega)$ norm as shown in the following proposition.

Proposition 2.3.14. *Under Assumption 2.3.11, for each $e \in \mathcal{E}_H$ the following estimate holds for the oversampling bubble part:*

$$\|R_e u^{\mathsf{b}}_{\omega_e}\|_{\mathcal{H}^{1/2}(e)} \leq CH\|f\|_{L^2(\omega_e)}\,,$$

where C is a constant independent of k and H.

The proof is deferred to Subsection 2.6.5.

We further define a special Helmholtz-harmonic function $u^{\mathsf{s}} \in V^{\mathsf{h}}$, such that that its restriction on each edge $e \in E_H$ equals $R_e u^{\mathsf{b}}_{\omega_e}$. Namely this special Helmholtz-harmonic function accounts for the second term in (2.3.13) for each edge. By the previous proposition, we immediately have the estimate

$$\|u^{\mathsf{s}}\|_{\mathcal{H}(\Omega)} \leq CH\|f\|_{L^2(\Omega)}\,,$$

where C is a constant independent of k and H. Along with Proposition 2.3.5, we conclude that there is a constant C_s independent of k and H such that

$$\|u^{\mathsf{s}}\|_{\mathcal{H}(\Omega)} + \|u^{\mathsf{b}}\|_{\mathcal{H}(\Omega)} \leq C_s H \|f\|_{L^2(\Omega)} . \tag{2.3.19}$$

Now consider the following space of basis functions:

$$\tilde{V}^{(1)}_{H,m,e} := W_{m,e} .$$

In practice, this space can be computed locally by an SVD of R_e. Due to (2.3.13) and (2.3.18), we have the following error estimate on each e:

$$\min_{\tilde{v}_e \in \tilde{V}^{(1)}_{H,m,e}} \|R_e u^{\mathsf{h}} - u^{\mathsf{s}} - \tilde{v}_e\|_{\mathcal{H}^{1/2}(e)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|u^{\mathsf{h}}_{\omega_e}\|_{\mathcal{H}(\omega_e)} . \tag{2.3.20}$$

Remark 2.3.15. *The operator R_e involves nodal interpolation, which is in general not stable for H^1 functions if the dimension is greater than* 1*. However, in Theorem 2.3.12, we take the domain of the operator to be $U(\omega_e)$, which contains Helmholtz-harmonic functions that are Hölder continuous, due to the standard C^α estimates for elliptic equations. More specifically, Lemma 2.6.2 implies the stability of R_e in this space.*

Remark 2.3.16. *If we follow the proof of Lemma 3.13 in [179], it is be possible to remove the small parameter ϵ in Theorem 2.3.12 to get a better asymptotic bound $O(\exp\left(-m^{\frac{1}{d+1}}\right))$.*

2.3.4.4 Low Complexity in Approximation

Finally, define the collection of edge basis functions

$$\tilde{V}^{(1)}_{H,m} = \text{span}\,\{\bigcup_e \tilde{V}^{(1)}_{H,m,e}\},$$

and denote by $\tilde{V}^{(0)}_H$ the span of the nodal interpolation basis used earlier, i.e. $\tilde{V}^{(0)}_H := \text{span}\,\{\tilde{\psi}_i\}$. Define the overall edge approximation $\tilde{V}_{H,m} = \text{span}\,\{\tilde{V}^{(0)}_H \bigcup \tilde{V}^{(1)}_{H,m}\}$. Let $V_{H,m} \subset V^{\mathsf{h}}$ be the corresponding part of $\tilde{V}_{H,m} \subset \tilde{V}^{\mathsf{h}}$, via (2.3.8). Then, using (2.3.20) and Theorem 2.3.9, we get a nearly exponentially decaying error estimate for approximating u^{h}; see Theorem 2.3.17.

Theorem 2.3.17. *Let $d = 2$. Under Assumption 2.3.11 and* (2.2.3)*, it holds that*

$$\min_{v \in V_{H,m}} \|u^{\mathsf{h}} - u^{\mathsf{s}} - v\|_{\mathcal{H}(\Omega)} \leq C_d(C_{\text{stab}}(k) + H) \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|f\|_{L^2(\Omega)},$$

where C_d is a generic constant independent of k, m, H.

Proof. By Theorem 2.3.12 and the global error estimate in Theorem 2.3.9, we get

$$\min_{v\in V_{H,m}} \|u^{\mathsf{h}} - u^{\mathsf{s}} - v\|^2_{\mathcal{H}(\Omega)} \leq C_{\mathrm{mesh}} C_\epsilon^2 \exp\left(-2m^{(\frac{1}{d+1}-\epsilon)}\right) \sum_{e\in\mathcal{E}_H} \|u^{\mathsf{h}}_{\omega_e}\|^2_{\mathcal{H}(\omega_e)}\,. \quad (2.3.21)$$

Due to Assumption 2.3.11, we have the elliptic estimate for the oversampling bubble part:

$$\|u^{\mathsf{b}}_{\omega_e}\|_{\mathcal{H}(\omega_e)} \leq \frac{3C'_P}{A_{\min}^{1/2}} H \|f\|_{L^2(\omega_e)}\,. \quad (2.3.22)$$

This is similar to Proposition 2.3.5, which is a consequence of Assumption 2.3.2. Then, using $u^{\mathsf{h}}_{\omega_e} = u - u^{\mathsf{b}}_{\omega_e}$, it follows that

$$\|u^{\mathsf{h}}_{\omega_e}\|^2_{\mathcal{H}(\omega_e)} \leq 2(\|u\|^2_{\mathcal{H}(\omega_e)} + \|u^{\mathsf{b}}_{\omega_e}\|^2_{\mathcal{H}(\omega_e)}) \leq \frac{18C'^2_P}{A_{\min}} H^2 \|f\|^2_{L^2(\omega_e)} + 2\|u\|^2_{\mathcal{H}(\omega_e)}\,. \quad (2.3.23)$$

Note that by our choice of oversampling domains, every element T can only be covered by $\{\omega_e\}_{e\in\mathcal{E}_H}$ at most C_1 times for a fixed C_1. Therefore it holds that

$$\sum_{e\in\mathcal{E}_H} \|f\|^2_{L^2(\omega_e)} \leq C_1 \|f\|^2_{L^2(\Omega)}, \quad (2.3.24)$$

as well as

$$\sum_{e\in\mathcal{E}_H} \|u\|^2_{\mathcal{H}(\omega_e)} \leq C_1 \|u\|^2_{\mathcal{H}(\Omega)} \leq C_1 C^2_{\mathrm{stab}}(k) \|f\|^2_{L^2(\Omega)}, \quad (2.3.25)$$

where the last inequality is due to the *a priori* estimate (2.2.3). Combining (2.3.21), (2.3.23), (2.3.24), and (2.3.25) completes the proof. □

Clearly, Theorem 2.3.17 implies the low complexity property of the part $u^{\mathsf{h}} - u^{\mathsf{s}}$. Each edge contains at most m basis functions, so the space $V_{H,m}$ is of dimension $O(m/H^d)$, while the approximation accuracy is of order $\exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right)$. We will use the space $V_{H,m}$ in our multiscale framework for approximating $u^{\mathsf{h}} - u^{\mathsf{s}}$.

Remark 2.3.18. *$V_{H,m}$ does not depend on the right hand side f or the solution u. Therefore, we can use the same $V_{H,m}$ for different right-hand sides.*

2.4 The Multiscale Methods

In this section, we discuss the multiscale methods for solving (2.1.1), based on the coarse-fine scale decomposition established in the last section.

By the nature of a multiscale algorithm, we will handle the "coarse part" $u^{\mathsf{h}} - u^{\mathsf{s}}$ and the "fine part" $u^{\mathsf{b}} + u^{\mathsf{s}}$ separately. Conceptually, the locality and small magnitude

of $u^{\mathsf{b}} + u^{\mathsf{s}}$ imply that it can be computed efficiently or ignored without affecting the accuracy much, and the low complexity of $u^{\mathsf{h}} - u^{\mathsf{s}}$ indicates that we can use a Galerkin method with a small number of basis functions to approximate it accurately.

In Subsection 2.4.1, we outline our general multiscale computational framework. Depending on how the trial and test spaces in the Galerkin method are selected, we get two categories of algorithms, namely the Ritz-Galerkin approach and Petrov-Galerkin approach that we will make precise in Subsections 2.4.2 and 2.4.3, respectively.

2.4.1 The Multiscale Framework

The bubble part u^{b} and the special function u^{s} are first computed locally. Given these parts, we form an effective equation for $u^{\mathsf{h}} - u^{\mathsf{s}}$ as

$$a(u^{\mathsf{h}} - u^{\mathsf{s}}, v) = (f, v)_{\Omega} - a(u^{\mathsf{b}} + u^{\mathsf{s}}, v)\,, \tag{2.4.1}$$

for any $v \in \mathcal{H}(\Omega)$.

Remark 2.4.1. *The right hand side in* (2.4.1) *can be seen as a bounded linear functional on $v \in \mathcal{H}(\Omega)$. By the estimate in* (2.2.4)*, this equation for $u^{\mathsf{h}} - u^{\mathsf{s}}$ (given fixed $u^{\mathsf{b}} + u^{\mathsf{s}}$) is well-posed.*

Numerically, we solve the equation (2.4.1) for $u^{\mathsf{h}} - u^{\mathsf{s}}$ using a Galerkin method. That is, we choose a trial space S and a test space S_{test} to find a numerical solution $u_S \in S$ that satisfies

$$a(u_S, v) = (f, v)_{\Omega} - a(u^{\mathsf{b}} + u^{\mathsf{s}}, v)\,, \tag{2.4.2}$$

for any $v \in S_{\text{test}}$. If $S_{\text{test}} = S$, then it is called a Ritz-Galerkin method, otherwise it is a Petrov-Galerkin method. Here since the equation is formulated in the complex domain, we specifically refer to the choice $S_{\text{test}} = \overline{S}$ as the Petrov-Galerkin method.

In Subsection 2.4.2, we formulate our Ritz-Galerkin method and present theories for the well-posedness of the discrete problem, as well as the error estimate in both the energy norm and the L^2 norm. In Subsection 2.4.3, we discuss the Petrov-Galerkin method, which is more straightforward and appears more convenient in practical computation.

2.4.2 The Ritz-Galerkin Method

First, we establish a general strategy for analyzing the Ritz-Galerkin method in solving (2.4.1). We start with a definition of the approximation accuracy of S.

Definition 2.4.2. *For $S \subset V^{\mathsf{h}}$, the approximation accuracy of S is defined as*

$$\eta(S) := \sup_{f \in L^2(\Omega)\setminus\{0\}} \inf_{v \in S} \frac{\|u - v\|_{\mathcal{H}(\Omega)}}{\|f\|_{L^2(\Omega)}}, \tag{2.4.3}$$

where u and f are related via the Helmholtz equation in (2.1.1)*.*

For the Ritz-Galerkin method, it turns out that $\eta(S)$ is critical in analyzing the solution errors of u_S.

Theorem 2.4.3. *Suppose* (2.2.3) *holds and $k\eta(S) \leq 1/(4C_c V_{\max})$ as well as $\overline{S} = S$. Then, the following statements hold for the Ritz-Galerkin method:*

1. *The Galerkin solution u_S is a quasi-optimal approximation in the sense that*

$$\|u^{\mathsf{h}} - u^{\mathsf{s}} - u_S\|_{\mathcal{H}(\Omega)} \leq 2C_c \inf_{v \in S} \|u^{\mathsf{h}} - u^{\mathsf{s}} - v\|_{\mathcal{H}(\Omega)},$$
$$\|u^{\mathsf{h}} - u^{\mathsf{s}} - u_S\|_{L^2(\Omega)} \leq C_c \eta(S) \|u^{\mathsf{h}} - u^{\mathsf{s}} - u_S\|_{\mathcal{H}(\Omega)}.$$

2. *If we further assume $Hk \leq 1/(8C_s C_c V_{\max})$, for constant C_s defined in* (2.3.19)*, the discrete problem satisfies the discrete inf-sup stability condition:*

$$\inf_{v \in S} \sup_{v' \in S\setminus\{0\}} \frac{|a(v, v')|}{\|v\|_{\mathcal{H}(\Omega)} \|v'\|_{\mathcal{H}(\Omega)}} \geq \frac{1}{4 + 3C_c^{-1} + 8kV_{\max} C_{\mathrm{stab}}(k)}.$$

The proof of this theorem is deferred to Subsection 2.6.6. It is inspired by the standard Gårding-type inequality for a posteriori estimate; see for example [185]. However, our proofs are slightly different since only the part $u^{\mathsf{h}} - u^{\mathsf{s}}$ is approximated via the basis functions.

The above theorem implies that once $\eta(S)$ is small, the discrete problem is well-posed, and the Galerkin solution approximates the exact solution accurately.

Given Theorem 2.4.3, we can choose $S = V_{H,m} + \overline{V_{H,m}}$ where $V_{H,m}$ is defined in Theorem 2.3.17 independent of the right hand side. For the quantity $\eta(S)$, we have the following estimate using its subspace $V_{H,m}$:

$$\eta(S) \leq \eta(V_{H,m}) \leq \max(C_d, C_s) \left((C_{\mathrm{stab}}(k) + H) \exp\left(-m^{\left(\frac{1}{d+1} - \epsilon\right)} \right) + H \right). \tag{2.4.4}$$

Here we have used (2.3.19) for the small parts u^{b} and u^{s} of size $O(H)$, and Theorem 2.3.17 for the approximation error for $u^{\mathsf{h}} - u^{\mathsf{s}}$. Invoking Theorems 2.4.3 and 2.3.17, we get the following error analysis for the Galerkin solution:

Theorem 2.4.4. *Let $d = 2$. Suppose Assumption 2.3.11 and* (2.2.3) *hold, and*

$$\max(C_d, C_s)k\left((C_{\text{stab}}(k) + H)\exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) + H\right) \leq 1/(4C_cV_{\max}),$$

where C_s, C_d are generic constants defined in (2.3.19) *and Theorem 2.3.17 respectively. Then using $S = V_{H,m} + \overline{V_{H,m}}$ in the Ritz-Galerkin method leads to a solution u_S that satisfies:*

$$\|u^{\mathsf{h}} - u^{\mathsf{s}} - u_S\|_{\mathcal{H}(\Omega)} \leq 2C_cC_d(C_{\text{stab}}(k) + H)\exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right)\|f\|_{L^2(\Omega)}. \quad (2.4.5)$$

For the ϵ that satisfies $\frac{1}{d+1} - \epsilon = \frac{1}{d+2}$, we can take $m \sim O(\log^{d+2}(kC_{\text{stab}}(k))$. Then the condition in Theorem 2.4.4 holds, provided that the mesh size H satisfies the following Assumption 2.4.5:

Assumption 2.4.5. *The mesh size satisfies $H \leq 1/(8\max(C_d, C_s)C_cV_{\max}k)$.*

Furthermore, if $C_{\text{stab}}(k) \leq C(1 + k^\gamma)$ for some constants γ and C, then the condition $m \sim O(\log^{d+2}(kC_{\text{stab}}(k))$ reduces to $m \sim \log^{d+2}(k)$. This implies that once m is moderately large, i.e., logarithmic in k, the nearly exponential convergence of the Galerkin solution shown in Theorem 2.4.4 will become effective. As in Remark 2.3.16, we can improve the index $d + 2$ to $d + 1$.

We provide several additional remarks of the Ritz-Galerkin method below.

Remark 2.4.6. *In the Ritz-Galerkin method, the trial and test spaces are $S = V_{H,m} + \overline{V_{H,m}}$. One can intuitively understand that $V_{H,m}$ is needed to represent the desired solution, and $\overline{V_{H,m}}$ is used for the approximation of the adjoint problem, which is required in the numerical analysis of the Helmholtz equation. There can be a lot of overlap between $V_{H,m}$ and $\overline{V_{H,m}}$: on each interior edge, since the singular vectors of R_e are real, these edge basis functions are real-valued. Thus, $V_{H,m}$ and $\overline{V_{H,m}}$ can only differ on the edges connected to the boundary, where the presence of the Robin boundary condition makes the operator non-Hermitian.*

Remark 2.4.7. *Combining* (2.4.5) *with the local computation of the fine parts will yield the overall error estimate for u, which is nearly exponentially convergent.*

2.4.3 The Petrov-Galerkin Method

In this subsection, we introduce the Petrov-Galerkin method. We choose $S = V_{H,m}$ and $S_{\text{test}} = \overline{V_{H,m}}$. We give the following remarks on this method.

Remark 2.4.8. *The trial and test spaces in the Petrov-Galerkin method often have smaller dimensions than their Ritz-Galerkin counterpart, since we do not put the complex conjugate $\overline{V_{H,m}}$ in S. This can save computational efforts.*

Remark 2.4.9. *Our current theory does not address the stability of the discrete system and the $\mathcal{H}(\Omega)$ error estimate for the Petrov-Galerkin method. This is left for our future work. We note that our numerical experiments in the next section imply that these properties also hold for the Petrov-Galerkin method.*

2.5 Numerical Experiments

In this section, we will outline and discuss our numerical algorithms in detail based on the established theoretical analysis. Several Helmholtz equations are solved using our algorithm, which confirm our theoretical results. We also consider some examples in which our theoretical assumptions are not satisfied. Even for these examples, our methods still give a nearly exponential rate of convergence. This provides further evidence for the robustness of our methods.

2.5.1 Set-up

We consider the domain $\Omega = [0,1] \times [0,1]$ and discretize it by a uniform two-level quadrilateral mesh; see a fraction of this mesh in Figure 2.3, where we also show an edge e and its oversampling domain ω_e in solid lines. The coarse and fine mesh

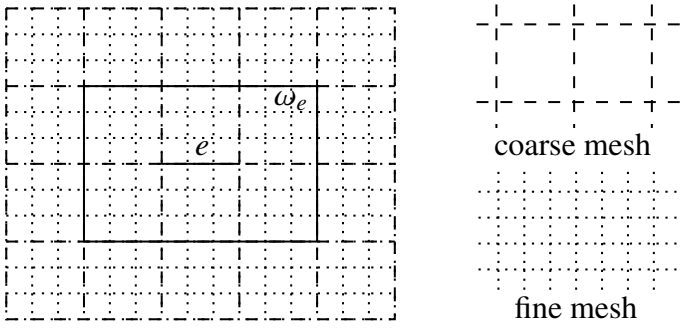


Figure 2.3: Two level mesh: a fraction

sizes are denoted by H and h, respectively.

For a given Helmholtz equation, we compute the reference solution u_{ref} using the classical FEM on the fine mesh; with a sufficiently small h, it is reasonable to treat u_{ref} as the ground truth u. We remark that via a posteriori estimates, we can check that the fine mesh indeed resolve the corresponding problems; thus the associated fine mesh solutions could serve as good reference solutions, for all of our numerical

examples. To be precise, we check that the relative error between the solutions using fine mesh of size h and $h/2$ are small, such that it is of order 10^{-2} in energy norm and 10^{-4} in L^2 norm.

The accuracy of a numerical solution u_{sol} is computed by comparing it with the reference solution u_{ref} on the fine mesh. The accuracy will be measured both in the L^2 norm and energy norm:

$$\begin{aligned} e_{L^2} &= \frac{\|u_{\text{ref}} - u_{\text{sol}}\|_{L^2(\Omega)}}{\|u_{\text{ref}}\|_{L^2(\Omega)}}, \\ e_{\mathcal{H}} &= \frac{\|u_{\text{ref}} - u_{\text{sol}}\|_{\mathcal{H}(\Omega)}}{\|u_{\text{ref}}\|_{\mathcal{H}(\Omega)}}. \end{aligned} \tag{2.5.1}$$

2.5.2 Multiscale Algorithms

We outline our numerical algorithms for obtaining u_{sol}. There are offline and online stages, depending on whether the steps involve the information of the right hand side.

2.5.2.1 Offline Stage

For each edge $e \in \mathcal{E}_H$ and its associated oversampling domain ω_e, the key step in the offline stage is to construct the discretized version of the operator

$$R_e : (U(\omega_e), \|\cdot\|_{\mathcal{H}(\omega_e)}) \to (H_{00}^{1/2}(e), \|\cdot\|_{\mathcal{H}^{1/2}(e)}),$$

which is defined by $R_e v = (v - I_H v)|_e$. Here $U(\omega_e)$ is defined in (2.3.15), $\|\cdot\|_{\mathcal{H}(\omega_e)}$ is the energy norm in ω_e, while $H_{00}^{1/2}(e)$ is the Lions-Magenes space, and $\|\cdot\|_{\mathcal{H}^{1/2}(e)}$ is defined in (2.3.9).

We note that functions in $U(\omega_e)$ are fully determined by their traces on $\partial\omega_e \backslash (\Gamma_N \cup \Gamma_R)$. Thus, we can take the discretized matrix version of R_e as a linear mapping from Dirichlet's data on $\partial\omega_e \backslash (\Gamma_N \cup \Gamma_R)$ to the image of R_e, which contains functions on the edge e. The discretization of the $\|\cdot\|_{\mathcal{H}(\omega_e)}$ and $\|\cdot\|_{\mathcal{H}^{1/2}(e)}$ norms leads to positive definite matrices on the discretized domains $\partial\omega_e \backslash (\Gamma_N \cup \Gamma_R)$ and e. To obtain these positive definite matrices, we construct the Helmholtz-harmonic extension operators both on e and $\partial\omega_e \backslash (\Gamma_N \cup \Gamma_R)$, which maps boundary data to the Hemholtz-harmonic function in the domain. Based on this operator, we can calculate the energy norms of the extended Hemholtz-harmonic function. This leads to the required norms as well as the positive definite matrices defining these norms[5].

[5] See also the implementation in Subsection 4.2 of [46] on how these matrices are constructed for elliptic problems.

With the discretized matrices constructed, the next step is to compute the top m left singular vectors of R_e for some selected $m \in \mathbb{N}$. This SVD problem turns out to be a generalized eigenvalue problem for these discrete matrices. For each e, denote the singular vectors by $\tilde{v}_{1,e}, ..., \tilde{v}_{m,e} \in H_{00}^{1/2}(e)$. Their Helmholtz-harmonic extensions to the domain are denoted by $v_{1,e}, ..., v_{m,e} \in \mathcal{H}(\Omega)$, obtained via the correspondence (2.3.8). The basis function space formed by the collection of all $v_{j,e}, 1 \leq j \leq m$ and $e \in \mathcal{E}_H$, together with the interpolation part $\{\psi_i\}_{x_i \in \mathcal{N}_H}$, are denoted by $V_{H,m}$ and will constitute the Galerkin basis as defined in Subsection 2.3.4.4. Note that here $\{\psi_i\}_{x_i \in \mathcal{N}_H}$ are the same as the basis functions in the MsFEM.

We are now in a position to construct our Galerkin basis and the associated stiffness matrix. The construction depends on how to choose the trial and test spaces in the Galerkin method. We will outline two possible choices below:

- Ritz-Galerkin: $S = V_{H,m} + \overline{V_{H,m}}$ and $S_{\text{test}} = S$.
- Petrov-Galerkin: $S = V_{H,m}$ and $S_{\text{test}} = \overline{V_{H,m}}$.

2.5.2.2 Online Stage

In the online stage, we solve the coarse and fine scales separately. Firstly we solve for u^{b} and u^{s}, and then we use the effective equation (2.4.1) to solve for $u^{\mathsf{h}} - u^{\mathsf{s}}$.

For the bubble part u^{b}, we solve the local Hemholtz problem in each element $T \in \mathcal{T}_H$, which leads to u_T^{b} defined in (2.3.2). Gluing them together leads to u^{b}.

For u^{s}, on each $e \in \mathcal{E}_H$ and ω_e, we construct the oversampling bubble part $u_{\omega_e}^{\mathsf{b}}$ via solving a local Helmholtz equation. Then, we get an edge function $R_e u_{\omega_e}^{\mathsf{b}}$ for each edge. We solve locally the Helmholtz-harmonic extension of these edge functions and add them together to obtain u^{s}.

Now we can form the right-hand side vector in our effective equation (2.4.1), and use the offline-assembled stiffness matrix to obtain the Galerkin solution for the part $u^{\mathsf{h}} - u^{\mathsf{s}}$.

This construction yields a practical numerical algorithm that efficiently handles multiple right-hand sides.

We note that all the above algorithms consider a uniform number of basis functions, namely m, for each edge $e \in \mathcal{E}_H$. It is also possible to make this number vary with edges, so it is thus fully adaptive to the problem's local properties such as the

approach in [128]. Consequently, this will lead to an adaptive algorithm where the truncated singular values serve as local error indicators. We do not pursue this in detail here and will leave this to our future work.

In the following, we will test our algorithms for different model problems. Our general set-up is to fix a reasonable coarse scale H and then study how the errors behave as m changes, for the two choices outlined above.

Remark 2.5.1. *Our numerical experience implies that in the Ritz-Galerkin method, one does not need to add the conjugate space $\overline{V_{H,m}}$ into S while still obtaining an exponential rate of convergence.*

2.5.3 A High Wavenumber Example: Planar Wave

We start with an example of planar wave where the coefficients are constant and the wavenumber is high. More precisely, we set $A = V = \beta = 1$ and $f = 0$. The wavenumber $k = 2^7$. We take the exact solution to be

$$u(x_1, x_2) = \exp(-ik(0.6x_1 + 0.8x_2)) .$$

Using this solution, we are able to specify the Robin boundary condition on $\partial\Omega$. Note that this is an inhomogeneous boundary condition, so it is beyond our previous discussion. In this case, the inhomogeneous data are incorporated to the equation of the bubble part u^{b}, while the treatment for the Helmholtz-harmonic part remains the same as that in the homogeneous case. To be specific, now our decomposition on each element T is $u = u_T^{\mathsf{h}} + u_T^{\mathsf{b}} + u_T^{\mathsf{p}}$ where u_T^{p} stands for a particular solution. The part $u_T^{\mathsf{b}} + u_T^{\mathsf{p}}$ satisfies

$$\begin{aligned} -\nabla \cdot (A\nabla(u_T^{\mathsf{b}} + u_T^{\mathsf{p}})) - k^2 V^2 (u_T^{\mathsf{b}} + u_T^{\mathsf{p}}) &= f, \text{ in } T \\ u_T^{\mathsf{b}} + u_T^{\mathsf{p}} &= 0, \text{ on } \partial T \setminus (\Gamma_N \cup \Gamma_R) \\ A\nabla(u_T^{\mathsf{b}} + u_T^{\mathsf{p}}) \cdot \nu &= T_k(u_T^{\mathsf{b}} + u_T^{\mathsf{p}}) + g, \text{ on } \partial T \cap (\Gamma_N \cup \Gamma_R) . \end{aligned}$$

We will use $u^{\mathsf{b}} + u^{\mathsf{p}}$ to replace u^{b} on the right-hand side of the effective equation for Galerkin solution (2.4.1). Similarly, when we compute the special Helmholtz-harmonic function u^{s} to account for the oversampling bubble part, its restriction on each edge equals $R_e(u^{\mathsf{b}}_{\omega_e} + u^{\mathsf{p}}_{\omega_e})$ instead of $R_e u^{\mathsf{b}}_{\omega_e}$. In this way we can take care of the boundary data via local particular problems and still obtain the desired accuracy. The error analysis in such case remains the same once we replace u_T^{b} in the homogeneous data case by $u_T^{\mathsf{b}} + u_T^{\mathsf{p}}$; in the bound we will also have the norm of g.

We set the fine mesh $h = 2^{-10}$, coarse mesh $H = 2^{-5}$. We vary the number of edge basis functions in each $e \in \mathcal{E}_H$, choosing $m = 1, 2, ..., 7$ and implementing the two algorithms outlined in Subsection 2.5.2.2. The results are shown in Figure 2.4. We

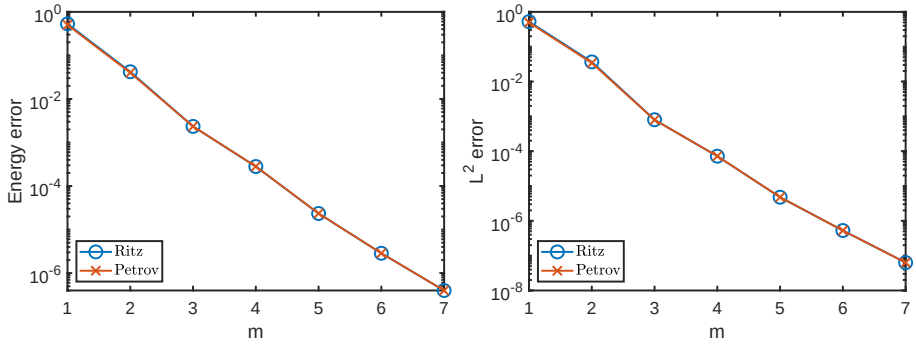


Figure 2.4: Numerical results for the high wavenumber example. Left: $e_{\mathcal{H}}$ versus m; right: e_{L^2} versus m.

observe that the online basis approaches achieve nearly exponential decaying errors with respect to m. The difference between the Ritz-Galerkin and Petrov-Galerkin approaches is almost negligible. We can see that a few basis per edge suffice for very high accuracy.

Furthermore, we make some comparison between our edge coupling approach (the Ritz-Galerkin version) and the PUM approach reported in [177]. We adopt the same setting there with $k = 100$, $H = 1/20$, $h = 1/1000$ and vary the number of edge basis functions in each $e \in \mathcal{E}_H$, choosing $m = 2, 3, ..., 7$. We present the results in Figure 2.5. We see that both errors decay very fast, and in particular, the error in

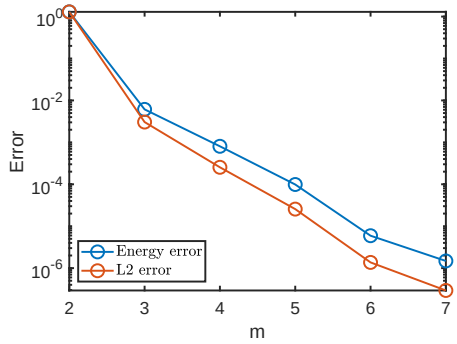


Figure 2.5: Numerical results for the high wavenumber example with $k = 100$, $H = 1/20$, $h = 1/1000$.

our method for $m = 7$ is smaller than the error in [177] with oversampling ratio $H_*/H = 2$ and 35 local basis per patch. With the same wavenumber and number

of coarse patches, our method uses a slightly larger oversampling domain, while reducing the number of multiscale basis by a factor of around $35/(2 \times 7) = 2.5$. Here, we have used the fact that the number of edges is twice as many as domains in 2D. Nevertheless, the support of basis functions in our approach and PUM approach could be different by a factor of 2, and the size of the overlapped domain decomposition in the PUM approach could also influence the result, leading to additional complexities for comparison. More detailed numerical study of the two approaches could be of future interest.

2.5.4 A High Contrast Example: Mie resonances

In this example, we consider an $A(x)$ with high contrast channels. More precisely, define the domain

$$\Omega_\varepsilon = (0.25, 0.75)^2 \cap \bigcup_{j \in \mathbb{Z}^2} \varepsilon \left(j + (0.25, 0.75)^2 \right), \tag{2.5.2}$$

and the coefficient is defined as

$$A(x) = \begin{cases} 1, & x \notin \Omega_\varepsilon \\ \varepsilon^2, & x \in \Omega_\varepsilon \end{cases}.$$

Here, ε is a parameter controlling the contrast. We choose $\varepsilon = 2^{-4}$ and visualize $\log_{10} A(x)$ in the left plot of Figure 2.6.

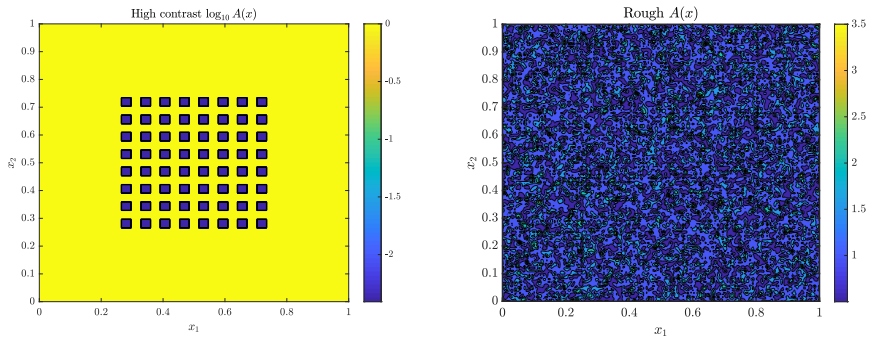


Figure 2.6: Left: the contour of $\log_{10} A$ for the high contrast example; right: the contour of A for the rough media example.

We take $\beta = 1, V = 1, k = 9$. For such a choice of k, the model exhibits an unusual behavior induced by Mie resonances in the small inclusions; see [199, 215]. An accurate numerical solution for this model would be hard to compute and it serves

as a proper benchmark for our method. The right hand side is

$$f(x_1,x_2)=\begin{cases}10000\exp\left(-\dfrac{1}{1-400\times \mathrm{dist}(x,z)^2}\right), & \mathrm{dist}(x,z)^2<\dfrac{1}{400}\\ 0, \text{ otherwise}, & \end{cases}$$

where $z=(0.125,0.5)$ and $\mathrm{dist}(x,z)^2=(x_1-0.125)^2+(x_2-0.5)^2$. We impose the homogeneous Robin boundary condition on $\partial\Omega$. We take the fine mesh $h=2^{-9}$ and the coarse mesh $H=2^{-5}$. As before we take $m=1,2,...,7$ and the numerical results are shown in Figure 2.7. A nearly exponential rate of convergence is observed consistently, and in this particular example, the Ritz method slightly outperforms the Petrov method.

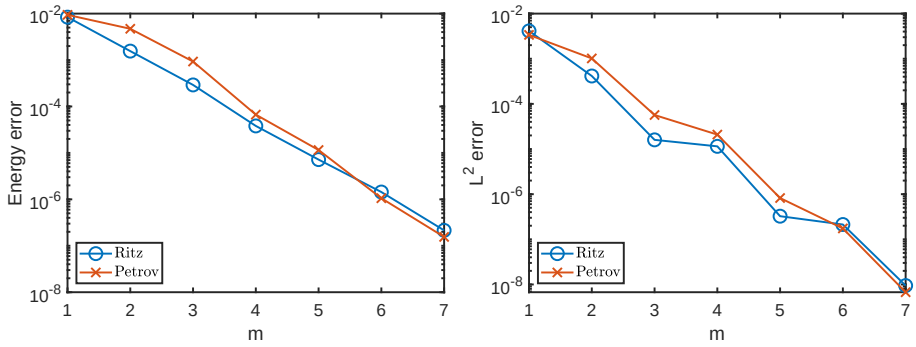


Figure 2.7: Numerical results for the high contrast example. Left: $e_{\mathcal{H}}$ versus m; right: e_{L^2} versus m.

2.5.5 An Numerical Example with Mixed Boundary and Rough Field

In the last example, we consider a mixed boundary problem. We impose the homogeneous Dirichlet boundary condition on $(x_1,0), x_1\in[0,1]$, the homogeneous Neumann boundary condition on $(x_1,1), x_1\in[0,1]$, and the homogeneous Robin boundary condition on the other two parts of $\partial\Omega$. We choose $A(x)$ to be a realization of some random field; more precisely,

$$A(x)=|\xi(x)|+0.5\,, \tag{2.5.3}$$

where the field $\xi(x)$ satisfies

$$\xi(x)=a_{11}\xi_{i,j}+a_{21}\xi_{i+1,j}+a_{12}\xi_{i,j+1}+a_{22}\xi_{i+1,j+1},\ \text{if } x\in[\frac{i}{2^7},\frac{i+1}{2^7})\times[\frac{j}{2^7},\frac{j+1}{2^7})\,.$$

Here, $\{\xi_{i,j}, 0\le i,j\le 2^7\}$ are i.i.d. unit Gaussian random variables. In addition, $a_{11}=(i+1-2^7x_1)(j+1-2^7x_2)$, $a_{21}=(2^7x_1-i)(j+1-2^7x_2)$, $a_{12}=(i+1-$

$2^7x_1)(2^7x_2-j)$, $a_{22}=(2^7x_1-i)(2^7x_2-j)$ are interpolating coefficients to make $\xi(x)$ piecewise linear. A sample from this field is displayed in the right plot of Figure 2.6.

Moreover, we also take $V(x)$ and $\beta(x)$ as independent samples drawn from this random field. We choose the wavenumber $k=2^5$, the right hand side $f(x_1,x_2)=x_1^4-x_2^3+1$, the fine mesh $h=2^{-10}$ and the coarse mesh $H=2^{-5}$. Again we take $m=1,2,...,7$ and present the numerical results in Figure 2.8. A nearly exponential

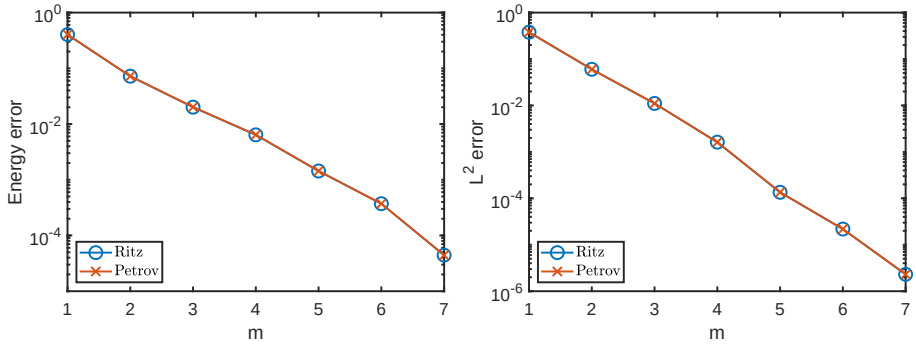


Figure 2.8: Numerical results for the mixed boundary and rough field example. Left: $e_{\mathcal{H}}$ versus m; right: e_{L^2} versus m.

rate of convergence is still observed for this challenging example. The differences between the Ritz-Galerkin method and Petrov-Galerkin method is very mild.

It is worth noting that this example is constructed artificially, mixing different kinds of boundary conditions and rough coefficients, without taking into account the analytical properties of this combination. Thus, the numerical results for this example demonstrate the effectiveness of our multiscale methods in a more general setting. Moreover, our right hand side f is global, so most oversampling bubble parts would be non-zero.

2.5.6 Summary

We summarize what we have observed in these numerical examples. Both algorithms lead to a nearly exponential rate of convergence with respect to m, and we are able to use the offline-computed Galerkin basis to solve for multiple right-hand sides.

Moreover, it is observed that the difference between the Ritz-Galerkin and the Petrov-Galerkin approaches is very mild in most cases, but sometimes Ritz-Galerkin method

can have better performances. Therefore, we recommend using the Ritz-Galerkin approach in practice.

2.6 Proofs

This section presents the theoretical proofs. Some proofs are similar to those in the elliptic case. We will refer these proofs to the corresponding proofs in the elliptic case [46], while we will make relevant remarks on possible changes and modifications.

2.6.1 Proof of Proposition 2.2.1

In this subsection, we provide the proof of the qualitative version of C^α estimate. It is a direct application of related results for elliptic equations.

Proof. We note that the Helmholtz PDE (2.1.1) is equivalent to

$$\begin{cases} -\nabla \cdot (A\nabla u) = f + k^2 V^2 u, \text{ in } \Omega \\ u = 0, \text{ on } \Gamma_D \\ A\nabla u \cdot \nu = T_k u, \text{ on } \Gamma_N \cup \Gamma_R \,. \end{cases} \tag{2.6.1}$$

Since $f \in L^2(\Omega)$, we know by the a priori estimate of the Helmholtz equation that $u \in H^1(\Omega)$. Therefore we can regard (2.6.1) as an elliptic PDE with k^2V^2u known as a part of the right hand side. This PDE has its right hand side in $L^2(\Omega)$ and has u as its solution. We can invoke the result in Remark 6.5 of [105], which concludes that u lies in some Hölder space $C^\alpha(\Omega)$ such that

$$\|u\|_{C^\alpha(\Omega)} \leq C(\|f\|_{L^2(\Omega)} + k^2\|u\|_{L^2(\Omega)}),$$

for some Hölder exponent $\alpha \in (0,1)$ and C. □

2.6.2 Proof of Proposition 2.3.7

The proof relies on the fact that any function v on e belonging to $H^{1/2}(e) \bigcap C^\alpha(e)$ and vanishing at ∂e will be in the space $H^{1/2}_{00}(e)$; see Proposition 2.1 in [46] for detailed arguments of this fact. Then, $R_e\tilde{u}^{\mathsf{h}} \in H^{1/2}(e) \bigcap C^\alpha(e)$ and vanishes at ∂e, so it belongs to $H^{1/2}_{00}(e)$.

2.6.3 Proof of Theorem 2.3.9

We decompose the energy norm into the contribution from each element $T \in \mathcal{T}_H$:

$$\|u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\in\mathcal{E}_H} v_e\|^2_{\mathcal{H}(\Omega)} = \sum_{T\in\mathcal{T}_H} \|u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\sim T} v_e\|^2_{\mathcal{H}(T)},$$

where we have used the fact that $v_e = 0$ in T if e and T are not neighbors.

Let us fix an element T. For each $e \sim T$, the trace of the function $u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\in T} v_e$ on e is $\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}} - \tilde{v}_e \in H_{00}^{1/2}(e)$. We can extend this trace to $\partial T\backslash e$ by 0 to get an $H^{1/2}(\partial T)$ boundary data. Then, this boundary data can be used to define a Helmholtz-harmonic function in T, via the correspondence (2.3.8). Using the triangle inequality and the Cauchy-Schwarz inequality, we get

$$\|u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\sim T} v_e\|^2_{\mathcal{H}(T)} \leq C_{\text{mesh}} \sum_{e\sim T} \|P_e(\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}) - \tilde{v}_e\|^2_{\mathcal{H}_T^{1/2}(e)},$$

where the $\mathcal{H}_T^{1/2}(e)$ norm of a function $\tilde{\psi} \in H_{00}^{1/2}(e)$ is defined as

$$\|\tilde{\psi}\|^2_{\mathcal{H}_T^{1/2}(e)} := \int_T A|\nabla\psi|^2 + k^2|V\psi|^2\,. \tag{2.6.2}$$

The constant C_{mesh} depends on the mesh type only; for example $C_{\text{mesh}} = 4$ for the quadrilateral mesh and $C_{\text{mesh}} = 3$ for the triangular mesh. Then, we sum the above inequality over all $T \in \mathcal{T}_H$, which yields

$$\begin{aligned}\|u^{\mathsf{h}} - I_H u^{\mathsf{h}} - \sum_{e\in\mathcal{E}_H} v_e\|^2_{\mathcal{H}(\Omega)} &\leq C_{\text{mesh}} \sum_{T\in\mathcal{T}_H}\sum_{e\sim T} \|P_e(\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}) - \tilde{v}_e\|^2_{\mathcal{H}_T^{1/2}(e)} \\ &= C_{\text{mesh}} \sum_{e\in\mathcal{E}_H} \|P_e(\tilde{u}^{\mathsf{h}} - I_H\tilde{u}^{\mathsf{h}}) - \tilde{v}_e\|^2_{\mathcal{H}^{1/2}(e)} \\ &\leq C_{\text{mesh}} \sum_{e\in\mathcal{E}_H} \epsilon_e^2\,.\end{aligned} \tag{2.6.3}$$

The proof is completed.

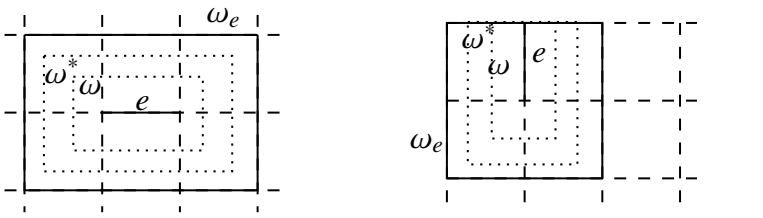
2.6.4 Proof of Theorem 2.3.12

This is the key theorem underlying the exponential convergence for approximating u^{h}. To prove it, we need to analyze the spectrum of the operator R_e for each edge e. The treatments for interior edges and edges connected to the boundary are slightly different, due to the different boundary conditions involved. We will explain the proof for interior edges in detail and comment on the changes needed to be made for edges connected to the boundary.

Since this theorem is stated for all edges, we start by discussing some geometric relations that hold uniformly for all interior edges.

2.6.4.1 Geometric Relation: Interior Edges

Suppose e is an interior edge, so that e lies strictly in the interior domain of ω_e; see Figure 2.2. We describe some geometric relation[6] between e and ω_e that will be needed in our analysis. Figure 2.9 illustrates our ideas for a uniform quadrilateral mesh. For each interior edge e, there exists two concentric rectangles $\omega \subset \omega^*$ with center being the midpoint m_e of e, such that $e \subset \omega \subset \omega^* \subset \omega_e$; the center m_e is the center of gravity of ω and ω^*. We require $\omega^* \cap \partial\Omega = \emptyset$. Moreover, one side of ω and ω^* should be parallel to e. We introduce three parameters l_1, l_2, l_3 to specify and describe the geometry:



interior edge edge connected to boundary

Figure 2.9: Geometric relation $e \subset \omega \subset \omega^* \subset \omega_e$

1. With respect to the center m_e, the two rectangles ω and ω^* are scaling equivalent, such that there exists $l_1 > 1$, $\omega^* - m_e = l_1 \cdot (\omega - m_e)$. Here we use the notation that $t \cdot X := \{tx : x \in X\}$ for a set X and a scalar t. For our choice of ω_e, the parameter l_1 can be selected to only depend on c_0 and c_1 in Subsection 2.3.1.1.

2. The ratio of ω's larger side length over the smaller side length is bounded by a uniform constant $l_2 > 1$ that depends on c_0 and c_1 only.

3. There is a constant $l_3 > 1$ depending on c_0 and c_1 only such that $l_3 \cdot e \subset \omega$.

We note that l_1, l_2, l_3 are universal constants for all interior edges. All three parameters depend on c_0, c_1 only. We introduce these parameters in order to get a uniform treatment for every interior edge. Indeed, several constants in our estimates depend on l_1, l_2, l_3, but not on k and H, uniformly for all interior edges.

[6] It is similar to that in Subsection 3.3.1 of [46].

2.6.4.2 Main Idea of the Proof

In the following, we explain the main ideas of our proof. Recall the target is to show the left singular values of R_e decays nearly exponentially fast. Similar to the rationale behind (2.3.17), it suffices to show there exists an $m-1$ dimensional space $W_{m,e} \subset H_{00}^{1/2}(e)$ such that

$$\min_{\tilde{v}_e \in W_{m,e}} \|R_e v - \tilde{v}_e\|_{\mathcal{H}^{1/2}(e)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|v\|_{\mathcal{H}(\omega_e)}, \tag{2.6.4}$$

for any $v \in U(\omega_e)$. We also use $U(\omega')$ to denote the function space in ω' defined via (2.3.15) with ω_e replaced by any ω'. Our proof contains two main steps, summarized in the following two lemmas.

Lemma 2.6.1. *For $d > 0$ and any $v \in U(\omega^*)$, there exists an $m-1$ dimensional space $\Phi_{m,e} \subset U(\omega)$ such that*

$$\min_{\chi \in \Phi_{m,e}} \|v - \chi\|_{\mathcal{H}(\omega)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|v\|_{\mathcal{H}(\omega^*)}, \tag{2.6.5}$$

for some C_ϵ independent of k and H.

Lemma 2.6.2. *For $d = 2$ and any $v \in H^1(\omega)$ and $\nabla \cdot (A\nabla v) \in L^2(\omega)$, it holds that*

$$\|R_e v\|_{\mathcal{H}^{1/2}(e)} \leq C \left(\|v\|_{\mathcal{H}(\omega)} + H \|\nabla \cdot (A\nabla v) + k^2 V^2 v\|_{L^2(\omega)} \right), \tag{2.6.6}$$

for some C independent of k and H.

Remark 2.6.3. *Here in Lemma 2.6.1 and 2.6.2, the constants are independent of k because in the local domain the operator behaves similarly to an elliptic operator. Moreover, for edges connected to the boundary that we will discuss in Subsection 2.6.4.7, the boundary condition is of order 1 after rescaling due to the assumption on our mesh size $Hk \lesssim 1$. Thus, eventually no k-dependence will be involved for our estimates in the local domain. This is different from the global C^α regularity estimate in the proof of Proposition 2.2.1. See Subsection 2.6.4.6 for details, where we only use C^α estimate of an elliptic equation.*

We will defer the proofs of the two lemmas to Subsections 2.6.4.3 and 2.6.4.6, and describe how to prove Theorem 2.3.12 using them here.

Proof of Theorem 2.3.12. From the above discussion, it remains to show (2.6.4). For $v \in \mathcal{H}(\omega_e)$, we have $v \in \mathcal{H}(\omega^*)$ and $\|v\|_{\mathcal{H}(\omega^*)} \leq \|v\|_{\mathcal{H}(\omega_e)}$. By Lemma 2.6.1, we get

$$\min_{\chi \in \Phi_{m,e}} \|v - \chi\|_{\mathcal{H}(\omega)} \leq C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|v\|_{\mathcal{H}(\omega_e)}.$$

Now since $v - \chi$ satisfies the condition in Lemma 2.6.2 and v and χ both vanish under the operator $v \to \nabla \cdot (A\nabla v) + k^2 V^2 v$, we obtain

$$\min_{\chi \in \Phi_{m,e}} \|R_e v - R_e \chi\|_{\mathcal{H}^{1/2}(e)} \leq C C_\epsilon \exp\left(-m^{(\frac{1}{d+1}-\epsilon)}\right) \|v\|_{\mathcal{H}(\omega_e)}.$$

Thus, taking $W_{m,e} = R_e \Phi_{m,e}$ completes the proof. □

2.6.4.3 Proof of Lemma 2.6.1

The proof of this lemma is inspired by Theorem 3.3 in [15], which states a similar result but for elliptic equations only. We generalize it here for the Helmholtz equation.

First, by our geometric construction, $\omega^* - m_e = l_1 \cdot (\omega - m_e)$. We denote a sequence of domains $\omega = \omega_0 \subset \omega_1 \subset \cdots \subset \omega_{N-1} \subset \omega_N = \omega^*$ such that they are concentric and that $\omega_j - m_e = (1+t)(\omega_{j-1} - m_e)$ for $j = 1, 2, \cdots, N$. Here $t = l_1^{1/N} - 1$. Then, there are two important lemmas, whose proofs are presented in Subsections 2.6.4.4 and 2.6.4.5.

Lemma 2.6.4. *For each $0 \leq j \leq N$ and any $n \in \mathbb{N}$, there is an n-dimensional space $W_n(\omega_j) \subset U(\omega_j)$, such that for all $v \in U(\omega_j)$, it holds that*

$$\inf_{w \in W_n(\omega_j)} \|v - w\|_{L^2(\omega_j)} \leq CHn^{-1/d} \|v\|_{\mathcal{H}(\omega_j)}, \tag{2.6.7}$$

where C is a generic constant independent of k, H, t, and n.

Lemma 2.6.5. *For each $1 \leq j \leq N$ and every $v \in U(\omega_j)$, it holds that*

$$\|v\|_{\mathcal{H}(\omega_{j-1})} \leq C/(tH) \|v\|_{L^2(\omega_j)}, \tag{2.6.8}$$

where C is a generic constant independent of k, H, and t.

With the two lemmas, we are ready to prove Lemma 2.6.1.

Proof of Lemma 2.6.1. Choose $n = \lfloor m/N \rfloor$. The proof relies on an iteration argument. We start from $j = N$. By (2.6.7) and (2.6.8), we get an n dimensional space $W_n(\omega_N) \subset U(\omega_N)$ and a function $w_N \in W_n(\omega_N)$ such that

$$\|v - w_N\|_{\mathcal{H}(\omega_{N-1})} \leq C/(tH) \|v - w_N\|_{L^2(\omega_N)} \leq Ct^{-1} n^{-1/d} \|v\|_{\mathcal{H}(\omega_N)},$$

where we have used the fact that the infimum in (2.6.7) is attained since it is a finite dimensional optimization problem. Here by abuse of notation the value of the

constant C varies in different places. It is a generic constant independent of k, H, t, and n.

Now, we iterate the above process. The function $v - w_N \in U(\omega_{N-1})$, so again by (2.6.7) and (2.6.8), we get an n dimensional space $W_n(\omega_{N-1}) \subset U(\omega_{N-1})$ and a function $w_{N-1} \in W_n(\omega_{N-1})$ such that

$$\|v - w_N - w_{N-1}\|_{\mathcal{H}(\omega_{N-2})} \leq Ct^{-1}n^{-1/d}\|v - w_N\|_{\mathcal{H}(\omega_{N-1})} \leq (Ct^{-1}n^{-1/d})^2\|v\|_{\mathcal{H}(\omega_N)}\,.$$

Repeating the above procedure, we get

$$\|v - \sum_{j=1}^{N} w_j\|_{\mathcal{H}(\omega)} \leq (Ct^{-1}n^{-1/d})^N\|v\|_{\mathcal{H}(\omega^*)}\,,$$

where each $w_j \in U(\omega_j) \subset U(\omega_0) = U(\omega)$. Therefore, there exists an $nN \leq m$ dimensional space $\Phi_{m,e} \subset U(\omega)$ such that

$$\inf_{w\in\Phi_{m,e}} \|v - w\|_{\mathcal{H}(\omega)} \leq (Ct^{-1}n^{-1/d})^N\|v\|_{\mathcal{H}(\omega^*)}\,.$$

For a paramter q to be determined later, choose $N = \left\lfloor m^{\frac{q}{q+1}} \right\rfloor$, then we obtain

$$(Ct^{-1}n^{-1/d})^N \leq \left(Ct^{-1}(\frac{m}{N})^{-1/d}\right)^N = \exp\left(N\left(\frac{1}{d}\log\left(\frac{N}{m}\right) + \log C - \log t\right)\right). \quad (2.6.9)$$

Using $N \leq m^{\frac{q}{q+1}}$ and $t = l_1^{1/N} - 1 = \exp\left(\frac{1}{N}\log l_1\right) - 1 \geq \frac{1}{N}\log l_1 \geq m^{-\frac{q}{q+1}}\log l_1$, we can bound the right hand side of (2.6.9) as

$$\begin{aligned}(Ct^{-1}n^{-1/d})^N &\leq \exp\left(-m^{\frac{q}{q+1}}\left((\frac{1}{d} - q)\frac{1}{q+1}\log m - \log C + \log\log l_1\right)\right) \\ &\leq C_q \exp\left(-m^{\frac{q}{q+1}}\right),\end{aligned} \quad (2.6.10)$$

for some constant C_q that depends on q, d, C, l_1, if $q < 1/d$. Here in the last inequality, we used the fact that when $q < 1/d$, there exists an M_q such that if $m \geq M_q$ then

$$(\frac{1}{d} - q)\frac{1}{q+1}\log m - \log C + \log\log l_1 \geq 1\,,$$

and thus $(Ct^{-1}n^{-1/d})^N \leq \exp\left(-m^{\frac{q}{q+1}}\right)$ for $m \geq M_q$. By choosing

$$C_q = \max_{1\leq m<M_q} \exp\left(-m^{\frac{q}{q+1}}\left((\frac{1}{d} - q)\frac{1}{q+1}\log m - \log C + \log\log l_1\right)\right)\exp\left(m^{\frac{q}{q+1}}\right) + 1$$

we can prove that (2.6.10) is valid.

Now, we choose $q < 1/d$ and denote $\frac{q}{q+1} = \frac{1}{d+1} - \epsilon$ for some $\epsilon > 0$. There is a one-to-one correspondence between q and small positive ϵ, so we can also write the error estimate in (2.6.10) in terms of ϵ as

$$(Ct^{-1}n^{-1/d})^N \leq C_\epsilon \exp\left(-m^{\frac{1}{d+1}-\epsilon}\right).$$

This completes the proof. □

2.6.4.4 Proof of Lemma 2.6.4

First, using the spectrum of the Laplacian operator with Neumann's boundary condition, there exists an n dimensional space $S_n \subset H^1(\omega_j)$ such that for any $v \in H^1(\omega_j)$,

$$\inf_{w\in S_n} \|v - w\|_{L^2(\omega_j)} \leq CHn^{-1/d}\|v\|_{H^1(\omega_j)} \leq CHn^{-1/d}\|v\|_{\mathcal{H}(\omega_j)}, \tag{2.6.11}$$

where C is a generic constant independent of k, H, t and n. Equivalently, this implies the identity embedding operator $Q : (\mathcal{H}(\omega_j), \|\cdot\|_{\mathcal{H}(\omega_j)}) \to (L^2(\omega_j), \|\cdot\|_{L^2(\omega_j)})$ such that $Qv = v$ is compact and the its n-th largest left singular value $\mu_n \leq CHn^{-1/d}$.

Now, since $U(\omega_j)$ is a closed subspace of $(\mathcal{H}(\omega_j), \|\cdot\|_{\mathcal{H}(\omega_j)})$, we can view Q as an operator from $(U(\omega_j), \|\cdot\|_{\mathcal{H}(\omega_j)})$ to $(L^2(\omega_j), \|\cdot\|_{L^2(\omega_j)})$. Denote its singular values in a non-increasing order by $\{\mu_n'\}$. Using the max-min theorem for singular values, we obtain

$$\begin{aligned}\mu_n' &= \max_{S_n\subset U(\omega_j),\dim(S_n)=n} \min_{v\in S_n, \|v\|_{\mathcal{H}(\omega_j)}=1} \|Qv\|_{L^2(\omega_j)} \\ &\leq \max_{S_n\subset \mathcal{H}(\omega_j),\dim(S_n)=n} \min_{v\in S_n, \|v\|_{\mathcal{H}(\omega_j)}=1} \|Qv\|_{L^2(\omega_j)} = \mu_n .\end{aligned}$$

Thus, $\mu_n' \leq CHn^{-1/d}$. Therefore, there is an n-dimensional space $W_n(\omega_j) \subset U(\omega_j)$, such that for all $v \in U(\omega_j)$, it holds that

$$\inf_{w\in S_n} \|v - w\|_{L^2(\omega_j)} \leq CHn^{-1/d}\|v\|_{H^1(\omega_j)} \leq CHn^{-1/d}\|v\|_{\mathcal{H}(\omega_j)}.$$

The proof is completed.

2.6.4.5 Proof of Lemma 2.6.5

We introduce a cutoff function $\eta \in C^1(\omega_j)$ such that $0 \leq \eta \leq 1$, and $\eta = 1$ in ω_{j-1}, as well as $|\nabla\eta(x)| \leq C/(tH)$ for some constant C independent of k, H, and t.

For any $v \in U(\omega_j)$, we use the test function $\eta^2 v$ and the weak form to get

$$(A\nabla v, \nabla(\eta^2 v))_{\omega_j} - k^2(Vv, V\eta^2 v)_{\omega_j} = 0, \tag{2.6.12}$$

where we have used the definition of $U(\omega_j)$ (see the beginning of Subsection 2.6.4.2), and the property of our construction that $\partial\omega_j \cap (\Gamma_N \cup \Gamma_R) = \emptyset$.

Using the relation $\|A^{1/2}\eta\nabla v\|^2_{L^2(\omega_j)} = (A\nabla v, \eta^2\nabla v)_{\omega_j}$ and the above formula, we obtain

$$\begin{aligned}\|A^{1/2}\eta\nabla v\|^2_{L^2(\omega_j)} &= -2(A^{1/2}\eta\nabla v, A^{1/2}v\nabla\eta)_{\omega_j} + k^2(Vv, V\eta^2 v)_{\omega_j}\,,\\ &\leq \frac{1}{2}\|A^{1/2}\eta\nabla v\|^2_{L^2(\omega_j)} + 2\|A^{1/2}v\nabla\eta\|^2_{L^2(\omega_j)} + k^2V^2_{\max}\|v\|^2_{L^2(\omega_j)}\,,\end{aligned} \tag{2.6.13}$$

which leads to $\|A^{1/2}\eta\nabla v\|^2_{L^2(\omega_j)} \leq 4\|A^{1/2}v\nabla\eta\|^2_{L^2(\omega_j)} + 2k^2V^2_{\max}\|v\|^2_{L^2(\omega_j)}$. Therefore, using the fact that $\eta = 1$ in ω_{j-1}, we have

$$\begin{aligned}\|v\|^2_{\mathcal{H}(\omega_{j-1})} &\leq \|A^{1/2}\eta\nabla v\|^2_{L^2(\omega_j)} + k^2V^2_{\max}\|v\|^2_{L^2(\omega_j)}\\ &\leq 4\|A^{1/2}v\nabla\eta\|^2_{L^2(\omega_j)} + 3k^2V^2_{\max}\|v\|^2_{L^2(\omega_j)}\\ &\leq \left(\frac{4C^2}{(tH)^2} + 3k^2V^2_{\max}\right)\|v\|^2_{L^2(\omega_j)}\\ &\leq \frac{C'^2}{(tH)^2}\|v\|^2_{L^2(\omega_j)}\,,\end{aligned} \tag{2.6.14}$$

for some C' independent of k, H and t, where we have used Assumption 2.3.2 such that $kV_{\max}H \leq C''$ for $C'' = A^{1/2}_{\min}/(\sqrt{2}C_P)$. This completes the proof.

2.6.4.6 Proof of Lemma 2.6.2

We use Lemma 3.10 of [46], which implies that

$$\|R_e v\|_{\mathcal{H}^{1/2}(e)} \leq C\left(\|A^{1/2}\nabla v\|_{L^2(\omega)} + H\|\nabla\cdot(A\nabla v)\|_{L^2(\omega)}\right)\,, \tag{2.6.15}$$

for some C independent of k, H. Indeed, Lemma 3.10 of [46] implies that C can depend on the C^α estimate constant of v in ω. The discussion in Remark 2.6.3 implies that this constant is independent of k.

By a triangular inequality, we have

$$\begin{aligned}H\|\nabla\cdot(A\nabla v)\|_{L^2(\omega)} &\leq H\|k^2V^2v\|_{L^2(\omega)} + H\|\nabla\cdot(A\nabla v) + k^2V^2v\|_{L^2(\omega)}\\ &\leq C'\|kVv\|_{L^2(\omega)} + H\|\nabla\cdot(A\nabla v) + k^2V^2v\|_{L^2(\omega)}\,,\end{aligned}$$

where we have used Assumption 2.3.2 such that $kV_{\max}H \leq C'$ for $C' = A^{1/2}_{\min}/(\sqrt{2}C_P)$. Now, using the definition of the $\mathcal{H}(\omega)$ norm, we have

$$\|A^{1/2}\nabla v\|_{L^2(\omega)} + C'\|kVv\|_{L^2(\omega)} \leq C''\|v\|_{\mathcal{H}(\omega)}\,,$$

for some generic constant C'' that does not depend on anything else. Combining the above inequalities concludes the proof.

2.6.4.7 For Edges Connected to the Boundary

The above proofs are for interior edges. For edges connected to the boundary, we need a different geometric relation, as depicted in the right of Figure 2.9. The quantitative characterization of this geometric relation is the same as that in Subsection 3.3.2 of [46], which introduces three other parameters l_4, l_5, l_6 to describe the geometry associated with edges, similar to l_1, l_2, l_3 for interior edges.

The main idea of the proof for this case is the same as that for the interior edges. We need to prove Lemmas 2.6.1 and 2.6.2 for edges connected to the boundary. The proof of Lemma 2.6.2 remains nearly the same. A technical part is that the constant in the inequality depends on the local C^α estimate. According to the discussion in Remark 2.6.3, the local C^α constant is independent of k for edges connected to the boundary. To prove Lemma 2.6.1, we again use the same strategy in Subsection 2.6.4.3, by establishing Lemmas 2.6.4 and 2.6.5 and then using an iteration argument. The iteration argument and the proof for Lemma 2.6.4 remain unchanged. For Lemma 2.6.5, the only slight change is (2.6.12), which becomes

$$(A\nabla v, \nabla(\eta^2 v))_{\omega_j} - k^2(Vv, V\eta^2 v)_{\omega_j} = (T_k v, \eta^2 v)_{\partial\omega_j \cap (\Gamma_N \cup \Gamma_R)}, \tag{2.6.16}$$

due to the boundary conditions involved. However, since $\mathrm{Re}(T_k v, \eta^2 v)_{\partial\omega_j \cap (\Gamma_N \cup \Gamma_R)} \leq 0$, the conclusion of Lemma 2.6.5 still holds.

Therefore, the result also holds for edges connected to the boundary.

Remark 2.6.6. *We have assumed that Ω is a polygonal domain, so the shape of the local domains around the boundary is well-behaved. In particular, a uniform Poincaré inequality will hold for these domains (in general, the constant in the Poincaré inequality depends on the shape of the domain). This guarantees that we can obtain a uniform constant in Theorem 2.3.12 for both interior edges and edges connected to the boundary.*

2.6.5 Proof of Proposition 2.3.14

First we have the bound on the oversampling bubble part in (2.3.22):

$$\|u^{\mathsf{b}}_{\omega_e}\|_{\mathcal{H}(\omega_e)} \leq \frac{3C'_P}{A_{\min}^{1/2}} H \|f\|_{L^2(\omega_e)}. \tag{2.6.17}$$

Applying Lemma 2.6.2 and the definition of $u^{\mathsf{b}}_{\omega_e}$ leads to

$$\begin{aligned} \|R_e u^{\mathsf{b}}_{\omega_e}\|_{\mathcal{H}^{1/2}(e)} &\leq C\left(\|u^{\mathsf{b}}_{\omega_e}\|_{\mathcal{H}(\omega)} + H\|\nabla\cdot(A\nabla u^{\mathsf{b}}_{\omega_e}) + k^2V^2u^{\mathsf{b}}_{\omega_e}\|_{L^2(\omega)}\right) \\ &\leq C' H \|f\|_{L^2(\omega_e)}, \end{aligned} \tag{2.6.18}$$

where C' is a constant independent of k and H.

2.6.6 Proof of Theorem 2.4.3

Proof. Define $e_S = u^{\mathsf{h}} - u^{\mathsf{s}} - u_S \in V^{\mathsf{h}}$. Take $\psi = N_k^\star(e_S)$. It holds that

$$\|e_S\|^2_{L^2(\Omega)} = a(e_S, \psi) = a(e_S, \psi - v)\,,$$

for any $v \in S$, due to the property of the Galerkin solution. Thus, using the boundedness of $a(\cdot,\cdot)$, we obtain that

$$\|e_S\|^2_{L^2(\Omega)} \le C_c \|e_S\|_{\mathcal{H}(\Omega)} \|\psi - v\|_{\mathcal{H}(\Omega)} = C_c \|e_S\|_{\mathcal{H}(\Omega)} \|\overline{\psi} - \overline{v}\|_{\mathcal{H}(\Omega)}\,. \tag{2.6.19}$$

As $\overline{\psi} = N_k \overline{e_S}$ according to the definition of the adjoint problem in Subsection 2.2.2, we can take infimum of v over S, using the fact that $S = \overline{S}$, the definition (2.4.3), the inequality (2.6.19), to get

$$\|e_S\|^2_{L^2(\Omega)} \le C_c \|e_S\|_{\mathcal{H}(\Omega)} \cdot \eta(S) \|\overline{e_S}\|_{L^2(\Omega)}\,,$$

which leads to the desired $L^2(\Omega)$ error estimate: $\|e_S\|_{L^2(\Omega)} \le C_c \eta(S) \|e_S\|_{\mathcal{H}(\Omega)}$.

For the $\mathcal{H}(\Omega)$ error, the property of Galerkin's solution implies that for any $v \in S$, we have

$$\begin{aligned}
\|e_S\|^2_{\mathcal{H}(\Omega)} &= \operatorname{Re} a(e_S, e_S) + \{\|e_S\|^2_{\mathcal{H}(\Omega)} - \operatorname{Re} a(e_S, e_S)\} \\
&= \operatorname{Re} a(e_S, u^{\mathsf{h}} - u^{\mathsf{s}} - v) + 2\|kV(x)e_S\|^2_{L^2(\Omega)} + \operatorname{Re}(T_k e_S, e_S)_{\Gamma_N \cup \Gamma_R} \\
&\le C_c \|e_S\|_{\mathcal{H}(\Omega)} \|u^{\mathsf{h}} - u^{\mathsf{s}} - v\|_{\mathcal{H}(\Omega)} + 2(kV_{\max} C_c \eta(S))^2 \|e_S\|^2_{\mathcal{H}(\Omega)}\,,
\end{aligned} \tag{2.6.20}$$

where we have used the fact that $\operatorname{Re}(T_k e_S, e_S)_{\Gamma_N \cup \Gamma_R} \le 0$ and the $L^2(\Omega)$ error estimate that we established earlier.

By the assumption $k\eta^{\mathsf{h}}(S) \le 1/(2C_c V_{\max})$, the last term in (2.6.20) is bounded by $\frac{1}{2}\|e_S\|^2_{\mathcal{H}(\Omega)}$. Thus due to the arbitrariness of v, we arrive at

$$\|e_S\|_{\mathcal{H}(\Omega)} \le 2C_c \inf_{v \in S} \|u^{\mathsf{h}} - v\|_{\mathcal{H}(\Omega)}\,.$$

This completes the proof for the first part. Next, we move to the proof for the discrete inf-sup stability. For any $v \in S$, set $z = 2N_k^\star(k^2 V^2 v) \in \mathcal{H}(\Omega)$ so that $a(v, z) = 2k^2 (V^2 v, v)_\Omega$. Plugging v and $v + z$ into the sesquilinear form yields

$$\begin{aligned}
a(v, v+z) &= a(v, v) + a(v, z) \\
&= (A\nabla v, \nabla v)_\Omega - k^2 (V^2 v, v)_\Omega - (T_k v, v)_{\Gamma_N \cup \Gamma_R} + 2k^2 (V^2 v, v)_\Omega \\
&= \|v\|^2_{\mathcal{H}(\Omega)} - (T_k v, v)_{\Gamma_N \cup \Gamma_R}\,.
\end{aligned}$$

By the definition of T_k, $\mathrm{Re}(T_k v, v)_{\Gamma_N \cup \Gamma_R} \leq 0$, so it holds that

$$\mathrm{Re}\, a(v, v+z) \geq \|v\|^2_{\mathcal{H}(\Omega)}\,.$$

Now, by the definition of the adjoint problem, we have $\overline{z} = 2N_k(k^2V^2\overline{v})$. Let $z_S \in S$ achieve the best approximation in (2.4.3) for $f = 2k^2V^2\overline{v}$, so that

$$\|\overline{z}^{\mathsf{h}} - \overline{z}^{\mathsf{s}} - z_S\|_{\mathcal{H}(\Omega)} \leq \eta(S)\|2k^2V^2\overline{v}\|_{L^2(\Omega)} \leq 2kV_{\max}\eta(S)\|v\|_{\mathcal{H}(\Omega)}\,. \tag{2.6.21}$$

We can choose $v' = v + \overline{z_S} \in S$ to compute

$$\mathrm{Re}\, a(v, v+\overline{z_S}) = \mathrm{Re}\, a(v, v+z) - \mathrm{Re}\, a(v, z-\overline{z_S}) \geq \|v\|^2_{\mathcal{H}(\Omega)} - C_c\|v\|_{\mathcal{H}(\Omega)}\|\overline{z} - z_S\|_{\mathcal{H}(\Omega)}\,.$$

We use the bound in (2.6.21) and the triangle inequality to get

$$|a(v, v+\overline{z_S})| \geq \|v\|^2_{\mathcal{H}(\Omega)}(1 - 2C_c kV_{\max}\eta(S)) - C_c\|v\|_{\mathcal{H}(\Omega)}(\|z^{\mathsf{s}}\|_{\mathcal{H}(\Omega)} + \|z^{\mathsf{b}}\|_{\mathcal{H}(\Omega)})\,.$$

Meanwhile, by a triangle inequality, we get

$$\|v + \overline{z_S}\|_{\mathcal{H}(\Omega)} \leq \|v\|_{\mathcal{H}(\Omega)} + \|z^{\mathsf{h}} - z^{\mathsf{s}} - \overline{z_S}\|_{\mathcal{H}(\Omega)} + \|z\|_{\mathcal{H}(\Omega)} + \|z^{\mathsf{s}}\|_{\mathcal{H}(\Omega)} + \|z^{\mathsf{b}}\|_{\mathcal{H}(\Omega)}\,.$$

Finally we are left to estimate the energy norm of z and its fine scale parts. By the stability estimate in (2.2.3), we have

$$\|z\|_{\mathcal{H}(\Omega)} \leq C_{\mathrm{stab}}(k)\|2k^2V^2v\|_{L^2(\Omega)} \leq 2C_{\mathrm{stab}}(k)kV_{\max}\|v\|_{\mathcal{H}(\Omega)}\,,$$

and by the bound on the fine part as given by (2.3.19), it holds that

$$\|z^{\mathsf{s}}\|_{\mathcal{H}(\Omega)} + \|z^{\mathsf{b}}\|_{\mathcal{H}(\Omega)} \leq C_s H\|2k^2V^2\overline{v}\|_{L^2(\Omega)} \leq 2C_s HkV_{\max}\|v\|_{\mathcal{H}(\Omega)}\,.$$

Therefore, we obtain

$$\begin{aligned}\sup_{v'\in S\setminus\{0\}} \frac{|a(v,v')|}{\|v\|_{\mathcal{H}(\Omega)}\|v'\|_{\mathcal{H}(\Omega)}} &\geq \frac{|a(v, v+\overline{z_S})|}{\|v\|_{\mathcal{H}(\Omega)}\|v+\overline{z_S}\|_{\mathcal{H}(\Omega)}} \\ &\geq \frac{(1 - 2\eta(S)C_c kV_{\max} - 2C_cC_s HkV_{\max})\|v\|^2_{\mathcal{H}(\Omega)}}{(1 + 2\eta(S)kV_{\max} + 2C_{\mathrm{stab}}(k)kV_{\max} + 2C_s HkV_{\max})\|v\|^2_{\mathcal{H}(\Omega)}}\,.\end{aligned}$$

Using the assumptions that $\eta(S)kV_{\max} \leq 1/(4C_c)$ and $C_s HkV_{\max} \leq 1/(8C_c)$, we obtain the desired estimate. □

2.7 Conclusions

We have developed a multiscale framework for solving the Helmholtz equation in heterogeneous media and high frequency regimes. The coarse-fine scale decomposition of the solution space is motivated by the MsFEM. In our algorithm, the coarse scale Helmholtz-harmonic part and the fine scale bubble part are computed separately. Their own structures are carefully explored, such as the low complexity of the coarse part and the locality of the fine part. A nearly exponential rate of convergence is proved rigorously and is confirmed numerically for a wide range of the Helmholtz equations with rough coefficients, high contrast, and mixed boundary conditions.

Perhaps surprisingly, our framework implies that designing an accurate multiscale method for the Helmholtz equation is not much more different from that for the elliptic equation. Many techniques in the elliptic case can be successfully adapted once the mesh size satisfies $H = O(1/k)$, a condition that does not suffer from the pollution effect. This work also demonstrates the broad applicability of our exponentially convergent multiscale framework proposed originally in [46].

Most discussions in this chapter are concerned with dimension $d = 2$. In our future work, we will generalize the methodology to dimension $d = 3$, where we can use nodal, edge, and face basis to approximate the local solution in the non-overlapped domain decomposition.

It is also of future interest to extend this methodology systematically to other equations such as the Schrodinger equation, where the problem is time-dependent and the potential function could introduce indefiniteness into the system. On the other hand, developing a better theoretical understanding of the behavior of the multiscale framework with respect to high contrast in the media is also an exciting direction for further exploration.

Chapter 3

ANALYSIS OF SUBSAMPLED LENGTHSCALES IN MULTISCALE METHODS

There is an intimate connection between numerical coarse-graining of multiscale PDEs and scattered data approximation of heterogeneous functions: the coarse variables selected for deriving an upscaled equation (in the former) correspond to the sampled information used for approximation (in the latter). As such, both problems can be thought of as recovering a target function based on some coarse data that are either artificially chosen by an upscaling algorithm or determined by some physical measurement process. In this chapter, we study, under such a setup and for a specific elliptic problem, how the lengthscale of the coarse data, which we refer to as the subsampled lengthscale, influences the accuracy of recovery, given limited computational budgets. Our analysis and experiments identify that reducing the subsampling lengthscale may improve the accuracy, implying a guiding criterion for coarse-graining or data acquisition in this computationally constrained scenario, especially leading to direct insights for the implementation of the Gamblets method in the numerical homogenization literature. Moreover, reducing the lengthscale to zero may lead to a blow-up of approximation error if the target function does not have enough regularity, suggesting the need for a stronger prior assumption on the target function to be approximated. We introduce a singular weight function to deal with it, both theoretically and numerically. This work sheds light on the interplay of the lengthscale of coarse data, the computational costs, the regularity of the target function, and the accuracy of approximations and numerical simulations.

The exposition of this chapter is based on our work [44], published in *SIAM Multiscale Modeling & Simulation*, 20(1):188–219, 2022.

3.1 Introduction

We are interested in studying a common approach for solving the following two categories of problems.

3.1.1 Problem 1: Numerical Upscaling

The aim of this problem is to identify the coarse scale solution of a multiscale PDE via solving an upscaled equation for coarse variables. As a prototypical

example, in $\Omega = [0,1]^d$, consider the elliptic equation for $u \in H_0^1(\Omega), f \in L^2(\Omega)$ and $\mathcal{L} = -\nabla \cdot (a\nabla\cdot)$:

$$\begin{cases} \mathcal{L}u = f, & \text{in } \Omega \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{3.1.1}$$

where the rough coefficient $a(x)$ satisfies $0 < a_{\min} \leq a(x) \leq a_{\max} < \infty$ for $x \in \Omega$. Suppose we select the upscaled data of the solution: $[u, \phi_i], i \in I$ where ϕ_i is some *measurement function* that is often localized in space, I is an index set and $[\cdot,\cdot]$ denotes the standard L^2 inner product. Then, the task is to derive an effective model for these upscaled variables and use them to approximate the solution of the PDE.

3.1.2 Problem 2: Scattered Data Approximation

This problem aims to recover a function u (assume it has an underlying PDE model as (3.1.1)) based on sampled data $[u, \phi_i], i \in I$. Here we intentionally use the same notation for the sampled data as that of the upscaled data in Problem 1 to make an explicit connection. We will also often call $[u, \phi_i], i \in I$ the coarse data in both problems.

3.1.3 A Common Approach

Problem 1 is a standard task in multiscale PDEs computations, while Problem 2 has more of its backgrounds from data scientific investigations. Despite their distinguished origins, there is an approach that solves and connects the two; studying of this method is the focus of the present chapter.

To motivate the method, we start from Problem 1: a natural and ideal approach for getting the coarse data is to multiply the equation with the set of *basis functions*:

$$\text{span } \{\psi_i\}_{i\in I} = \text{span } \{\mathcal{L}^{-1}\phi_i\}_{i\in I},$$

so that $[\psi_i, f], i \in I$, after an integration by part, matches the target $[u, \phi_i], i \in I$.

Phrased in the language of Galerkin's method, $\{\psi_i\}_{i\in I}$ will constitute the test space; furthermore, one needs to select a trial space V (with the same dimension) in order to get the ultimate numerical approximation of u. As such, this viewpoint has interpreted Problem 1 as a special case of Problem 2, of recovering u, from $[u, \phi_i], i \in I$, via choosing a space V. Often and conveniently, the trial space $V = \text{span } \{\psi_i\}_{i\in I}$ is chosen to be the same as the test space. Under such a choice and after selecting a suitable representative basis $\{\psi_i\}_{i\in I}$ of the linear space V so

that $[\psi_i, \phi_j] = \delta_{ij}$, we can write the final solution in a concise form:

$$u^{\text{ideal}} := \sum_{i \in I} [u, \phi_i] \psi_i \, . \tag{3.1.2}$$

It is the ideal solution (here, "ideal" means that we have not accounted for the computational cost yet) in this setting, both to numerical upscaling and scattered data approximation. In practice, the basis function ψ_i can have global support, and we need a localization step for efficient computation.

As a special case in numerical upscaling, if we choose ϕ_i to be piecewise linear tent functions, then we get the ideal LOD method [181]; if ϕ_i is set to be piecewise constant functions, then we obtain the Gamblet method in [203]. In their contexts, localization of $\{\psi_i\}_{i \in I}$ is achieved via an exponential decay property, and a provable accuracy guarantee has been established by controlling the coarse-graining error of using u^{ideal} to approximate u and the localization error of computing $\{\psi_i\}_{i \in I}$, respectively.

3.1.4 Our Goals

The purposes of this chapter are twofold.

- On the numerical upscaling side, we contribute a further discussion to this family of upscaling methods, concentrating on the fundamental role of a *subsampled lengthscale* (defined in the next subsection) in choosing $\{\phi_i\}_{i \in I}$, with its highly non-trivial consequence on the localization of $\{\psi_i\}_{i \in I}$ and the solution accuracy of u. We will get a novel trade-off between approximation and localization regarding the subsampled scale.

- On the function approximation side, the above recovery method takes advantage of the underlying physical model (3.1.1), combining the merits of data and physics. In addition to contributing a detailed analysis of accuracy and comparisons to numerical upscaling, we will pay close attention to the regime where the subsampled lengthscale is small and approaches zero, in which we provide some numerical evidence that exemplifies, and extends our earlier work on function approximation via subsampled data [44].

Our detailed contributions are outlined in Subsection 3.1.7.

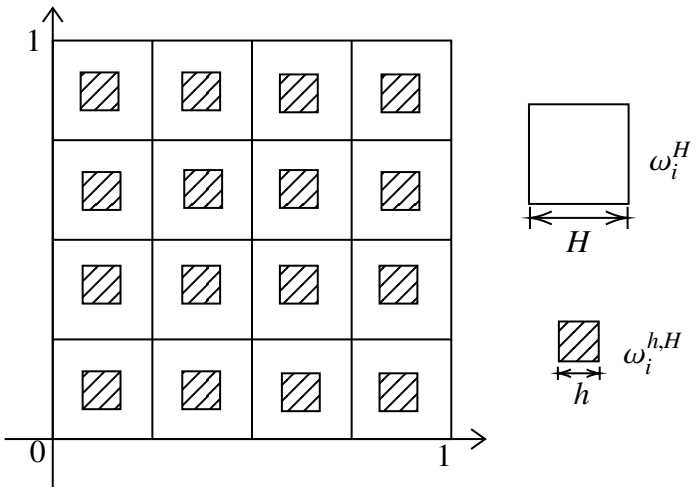


Figure 3.1: Illustration of Subsampled Data: $H = 1/4, h = 1/10$

3.1.5 Subsampled Lengthscales

We begin by introducing the concept of *subsampled data*. For a demonstration of ideas, we work on the domain $\Omega = [0,1]^d$, and it is decomposed uniformly into cubes with side length H; this becomes our coarse grid. Let I be the index set of these cubes such that its cardinality $|I| = 1/H^d$. The measurement function $\phi_i^{h,H}$ (we use superscripts now for notational convenience) for each $i \in I$ is set to be the (L^1 normalized) indicator function of a cube with side length $0 < h \leq H$, centered in the corresponding cube with side length H; see Figure 3.1 for a two dimensional example[1]. For each $i \in I$, these two cubes are denoted by ω_i^H and $\omega_i^{h,H}$ respectively; we assume they are closed sets, i.e., their boundaries are included. We will call H the coarse lengthscale, and h is the *subsampled lengthscale*.

The consideration of this subsampled lengthscale is natural both from the perspectives of function approximation and numerical upscaling. In the former scenario, the measurement data of a field function in physics is often the macroscopic averaged quantity, taking a similar form as $[u, \phi_i^{h,H}]$ for some $h \leq H$. In the latter problem, we have the freedom to choose the upscaled information of the multiscale PDEs, so taking a free parameter h in the approach enables us to analyze the algorithm's behavior more thoroughly. Later on, we will see that the parameter h has a non-trivial influence on the subsequent localization and accuracy of the approximation.

[1]For illustration, the cube $\omega_i^{h,H}$ in the figure is centered in ω_i^H. However, the relative position of the two cubes is not important in our analysis; see the proofs of Theorem 3.2.1 and 3.2.3. The key is that the subsampled Poincaré inequality developed in [44] does not depend on the relative position of the subdomain and the domain.

Note that the choice of ω_i^H and $\omega_i^{h,H}$ being cubes here is for convenience of analysis only; results in this chapter will generalize easily to regular domains with other shapes.

3.1.6 Basis Functions and Localization

Before outlining our main contributions (which are in the next subsection), we make precise here the definition of the basis functions and their localization. Per the discussion in Subsection 3.1 and especially the formula (3.1.2), the basis function $\psi_i^{h,H}$ (we add the superscripts for notational clarity) is the solution of the following variational problem:

$$\begin{aligned} \psi_i^{h,H} = \text{argmin}_{\psi \in H_0^1(\Omega)} \quad & \|\psi\|^2_{H_a^1(\Omega)} \\ \text{subject to} \quad & [\psi, \phi_j^{h,H}] = \delta_{i,j} \ \text{ for } \ j \in I\,, \end{aligned} \tag{3.1.3}$$

where, we have used the notation $\|\psi\|^2_{H_a^1(\Omega)} := \int_\Omega a|\nabla\psi|^2$. This formulation is a consequence of the two properties that are mentioned in Subsection 3.1:

$$\text{(I) span } \{\psi_i^{h,H}\}_{i\in I} = \text{span } \{\mathcal{L}^{-1}\phi_i^{h,H}\}_{i\in I} \quad \text{ and } \quad \text{(II) } [\psi_i^{h,H}, \phi_j^{h,H}] = \delta_{ij}\,.$$

For ease of computation, in practice we will solve a localized version of (3.1.3) instead:

$$\begin{aligned} \psi_i^{h,H,l} = \text{argmin}_{\psi \in H_0^1(\mathrm{N}^l(\omega_i^H))} \quad & \|\psi\|^2_{H_a^1(\mathrm{N}^l(\omega_i^H))} \\ \text{subject to} \quad & [\psi, \phi_j^{h,H}] = \delta_{i,j} \ \text{ for } \ j \in I\,, \end{aligned} \tag{3.1.4}$$

where $l \in \mathbb{N}$ is called the *oversampled layer*. We have $\mathrm{N}^0(\omega_i^H) = \omega_i^H$, and recursively:

$$\mathrm{N}^l(\omega_i^H) := \bigcup \{\omega_j^H, j \in I : \omega_j^H \cap \mathrm{N}^{l-1}(\omega_i^H) \neq \emptyset\}\,. \tag{3.1.5}$$

Then, the level-l localized solution for Problem 2 is

$$u^{\text{loc},l} := \sum_{i\in I} [u, \phi_i^{h,H}]\psi_i^{h,H,l}\,. \tag{3.1.6}$$

By abuse of notation, we will equate $u^{\text{loc},\infty} = u^{\text{ideal}}$. The energy error and L^2 error of this localized solution are written as

$$\begin{aligned} e_1^{h,H,l}(a,u) &= \|u - u^{\text{loc},l}\|_{H_a^1(\Omega)}\,, \\ e_0^{h,H,l}(a,u) &= \|u - u^{\text{loc},l}\|_{L^2(\Omega)}\,. \end{aligned} \tag{3.1.7}$$

For Problem 1, we also get a solution $\tilde{u}^{\text{loc},l}$ by using the localized basis functions $\{\psi_i^{h,H,l}\}_{i\in I}$ and the Galerkin method. This solution is different from $u^{\text{loc},l}$ in general,

unless $l = \infty$, i.e., in the ideal case. The corresponding energy error and L^2 error of $\tilde{u}^{\mathrm{loc},l}$ are denoted by $\tilde{e}_1^{h,H,l}(a,u)$ and $\tilde{e}_0^{h,H,l}(a,u)$.

We call $u^{\mathrm{loc},l}$ the *recovery solution* of Problem 2, and $\tilde{u}^{\mathrm{loc},l}$ the *Galerkin solution* of Problem 1. The computation costs of the two solutions are different—the former only requires solving the basis functions, while the latter also needs to solve an upscaled equation. Their errors in the solution are called the recovery error and Galerkin error, respectively.

Under the above setup, our precise goal in this chapter is to understand how the recovery error and Galerkin error depend on the following three factors:

1. The coarse scale H and subsampled lengthscale h;
2. The oversampled layer l (corresponded to the computational budget);
3. The regularity of function u (in function approximation, it is given as prior information; in multiscale PDEs, it is influenced by the right-hand side f).

Note that the regularity of a function is also intimately connected to the dimension parameter d.

3.1.7 Our Contributions

In the first part of this work, we consider the finite regime of the subsampled lengthscale, i.e., h is a strictly positive number.

- We provide numerical experiments and theoretical analysis of these recovery and Galerkin errors. We show that for a fixed h/H, if $l = O(\log(1/H))$, then both energy errors are of $O(H)$ and both L^2 errors are of $O(H^2)$.
- Further, we decompose the error into two parts: the approximation error of the ideal solution and the localization error. We demonstrate that there is a competition between the two. Roughly, reducing h worsens the former, while improving the latter, for a fixed H and l. This leads to a novel trade-off that was not investigated before—choosing an appropriate h can benefit the final accuracy.
- Moreover, there appears a fundamental difference between $e_0^{h,H,l}(a,u)$ and the other three errors, when $d \geq 2$. For a fixed l and h/H, the former remains

bounded as $H \to 0$, while the other three blow up. We characterize this phenomenon both theoretically and numerically.

In the second part of this work, we consider the small limit regime of h. When $d \geq 2$, the error estimates in the first part blow up as $h \to 0$. To remedy this issue in the context of scattered data approximation, we propose to use a singular weight function in the algorithm. The weight function puts more importance on the subsampled data and avoids the degeneracy, given the target function has improved regularity property around these data. Numerical experiments and theoretical analysis are presented to offer a quantitative explanation of this phenomenon.

3.1.8 Related Works

We review the related works below.

3.1.8.1 Numerical Upscaling

There have been vast literature on numerical upscaling of multiscale PDEs. For our context, i.e., elliptic PDEs with rough coefficients, rigorous theoretical results include Generalized Finite Element Methods (GFEM) [14, 15], Harmonic Coordinates [209], Local Orthogonal Decomposition (LOD) [181, 121, 148, 80, 117, 180], Gamblets related approaches [208, 210, 201, 203, 131, 204], and generalizations of Multiscale Finite Element Methods (MsFEM) [128, 52, 160, 89, 46, 48], etc. Among them, the ones most related to this chapter are LOD and Gamblets; the connection has been explained in Subsection 3.1.3. Indeed, in Gamblets [203, 204], the author has formulated the framework in the perspective of optimal recovery, bridging numerical upscaling to game-theoretical approaches and Gaussian process regressions for function recovery. This formulation connects our Problem 1 and Problem 2 in Subsection 3.1.

A main component in LOD and Gamblets is the localization problem – the ideal multiscale basis functions need to be localized for efficient computation. In this chapter, our localization strategy, as outlined in Subsection 3.1.6, follows from the one in [181, 203]. The main difference is that our measurement function $\phi_i^{h,H}$ contains a subsampled lengthscale parameter, which makes the analysis more delicate. Moreover, in addition to showing a trade-off between approximation errors and localization errors regarding the oversampling parameter l, our setup allows us to discover another trade-off regarding the subsampled lengthscale h: a good choice of h can improve the algorithm in [181, 203]. We also remark that the work [161]

has considered a similar algorithm for convection-dominated diffusion equations, where h is fixed to be the small scale grid size, but the analysis there did not reveal the trade-off here.

3.1.8.2 Function Approximation

Function approximation via scattered data is a classical problem in numerical analysis (interpolation), statistics (non-parametric regression), and machine learning (supervised learning). For the type of scattered data, the most frequently considered one is the pointwise data [288]. The subsampled data introduce an additional small scale parameter h, and are generalizations to pointwise data. Our earlier work [44] performed some analysis on this aspect, and provides some theoretical foundation for this work. The multiscale basis functions constructed for the subsampled data allow us to capture the heterogeneous behaviors of the target function.

The method in Subsection 3.1.3 connects to the graph Laplacian approach in semisupervised learning. In the machine learning literature, the degeneracy issue of graph Laplacians has long been studied, and various approaches have been proposed to remedy this issue. Among them, the one that is most related to our work is the weighted graph Laplacian method [248, 38], which puts more weights around the labeled data to avoid degeneracy. The second part of this work presents some analysis for this type of idea in the context of numerical analysis.

3.1.9 Organization

The rest of this chapter is organized as follows. In Section 3.2 we discuss the regime that $0 < h \leq H$. We present numerical experiments and theoretical analysis of these Galerkin errors in numerical upscaling, and recovery errors in function approximation. In Section 3.3, we consider the regime $h \to 0$, a case that degeneracy may occur. We use a singular weight function to deal with this issue both numerically and theoretically. Section 3.4 contains all the proofs. We summarize, discuss, and conclude this chapter in Section 3.5.

3.2 Finite Regime of Subsampled Lengthscales

In this section, we study the finite regime of h, i.e., $0 < h \leq H$. We start with the ideal solution u^{ideal}, or equivalently $u^{\text{loc},\infty}$, and then move to the localized solution $u^{\text{loc},l}$ and $\tilde{u}^{\text{loc},l}$ for finite l. Experiments are presented first, followed with theoretical analysis. Special attention is paid to the dependence of accuracy on the coarse scale H, subsampled lengthscale h and when in the localized case, the oversampling

parameter l.

3.2.1 Experiments: Ideal Solution

In this subsection, we perform a numerical study of the effect of h in $e_1^{h,H,\infty}(a,u)$ and $e_0^{h,H,\infty}(a,u)$, for $d = 1$ and 2 respectively.

In this ideal case, the recovery solution and Galerkin solution are the same, and in our computation, we directly solve a PDE to get these solutions. Theoretical analysis of these numerical results is given in Subsection 3.2.2.

3.2.1.1 One Dimensional Example

We consider the domain $\Omega = [0,1]$. The rough coefficient $a(x)$ is a sample drawn from the random field

$$\xi = 1 + 0.5 \times \sin\left(\sum_{k=1}^{100} \eta_k \cos(kx) + \zeta_k \sin(kx)\right), \tag{3.2.1}$$

where $\eta_k, \zeta_k, 1 \leq k \leq 100$ are i.i.d. random variables uniformly distributed in $[-0.5, 0.5]$; see the upper left of Figure 3.2 for a single realization. The right-hand side f is drawn from the Gaussian process $\mathcal{N}(0, (-\Delta)^{-0.5-\delta})$ for $\delta = 10^{-2}$; this guarantees $f \in H^t(\Omega)$ for any $t < \delta$ but not $t \geq \delta$; see the upper right of Figure 3.2 for a single realization of this process. Note that this set-up of f ensures that it is roughly an element in $L^2(\Omega)$ and has no apparent higher regularity. This is important because we do not want f to be too regular to influence the results, as our focus is on $f \in L^2(\Omega)$.

In the lower part of Figure 3.2, we output the energy errors and L^2 errors of the ideal solution, $e_1^{h,H,\infty}(a,u)$ and $e_0^{h,H,\infty}(a,u)$, for $H = 2^{-2}, 2^{-3}, ..., 2^{-7}$ and the subsampled ratio $h/H = 1, 1/2, 1/4, 1/8$. The grid size we use to discretize the operator is set to be 2^{-11}. These two figures lead to the following observations:

- For the ideal solution, the energy error decays linearly with respect to the coarse scale H, while the L^2 error decays quadratically.
- Decreasing h leads to a decrease of accuracy.

In the next subsection, we move to a two dimensional example to further confirm these observations.

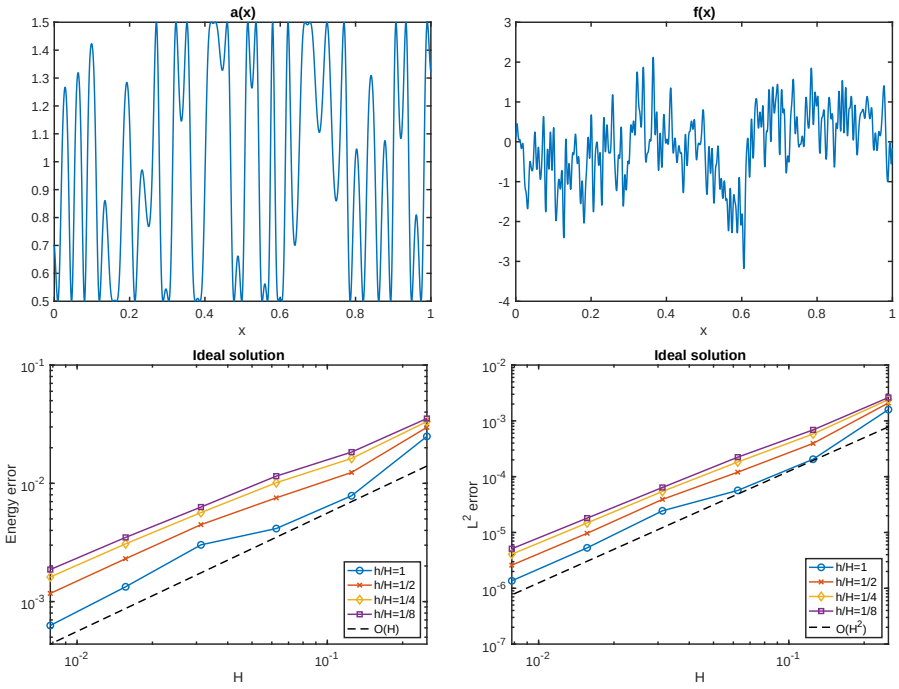


Figure 3.2: 1D example, ideal solution. Upper left: $a(x)$; upper right: $f(x)$; lower left: energy error; lower right: L^2 error.

3.2.1.2 Two Dimensional Example

We consider $\Omega = [0,1]^2$. The coefficient $a(x)$ is chosen as

$$a(x) = \frac{1}{6}\left(\frac{1.1+\sin(2\pi x_1/\epsilon_1)}{1.1+\sin(2\pi x_2/\epsilon_1)} + \frac{1.1+\sin(2\pi x_2/\epsilon_2)}{1.1+\cos(2\pi x_1/\epsilon_2)} + \frac{1.1+\cos(2\pi x_1/\epsilon_3)}{1.1+\sin(2\pi x_2/\epsilon_3)} + \frac{1.1+\sin(2\pi x_2/\epsilon_4)}{1.1+\cos(2\pi x_1/\epsilon_4)} + \frac{1.1+\cos(2\pi x_1/\epsilon_5)}{1.1+\sin(2\pi x_2/\epsilon_5)} + \sin\left(4x_1^2x_2^2\right) + 1\right), \tag{3.2.2}$$

where $\epsilon_1 = 1/5$, $\epsilon_2 = 1/13$, $\epsilon_3 = 1/17$, $\epsilon_4 = 1/31$, $\epsilon_5 = 1/65$. For the right-hand side, we sample two independent one-dimensional process in the last subsection, denoted by $f_1(x_1)$ and $f_2(x_2)$, and we set $f(x) = f_1(x_1)f_2(x_2)$. This guarantees $f \in H^t(\Omega)$ for any $t < \delta$ but not $t \geq \delta$ in two dimensions.

In the upper part of Figure 3.3, we output $a(x)$ and a single realization of $f(x)$. The lower part depicts $e_1^{h,H,\infty}(a,u)$ and $e_0^{h,H,\infty}(a,u)$, for $H = 2^{-2}, 2^{-3}, ..., 2^{-6}$ and the subsampled ratio $h/H = 1, 3/4, 1/2, 1/4$. The grid size we use to discretize the operator is set to be 2^{-8}. These two figures yield the same conclusions as those in the one dimensional case.

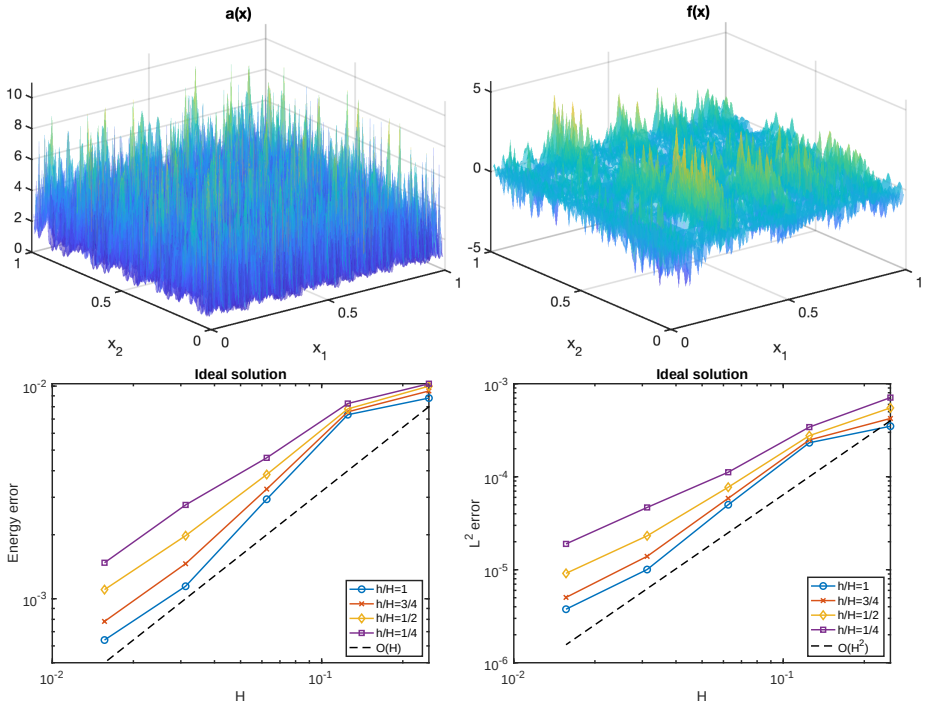


Figure 3.3: 2D example, ideal solution. Upper left: $a(x)$; upper right: $f(x)$; lower left: energy error; lower right: L^2 error.

3.2.2 Analysis: Ideal Solution

In this subsection, we move to the theoretical analysis of the ideal solution, to understand better of the above empirical observations.

For this purpose, we use our earlier results in function approximation via subsampled data [44]. Especially, Theorem 3.3 in [44] implies the following result:

Theorem 3.2.1. *For the ideal solution, it holds that*

$$e_1^{h,H,\infty}(a,u) \leq \frac{1}{\sqrt{a_{\min}}} C_1(d) H \rho_{2,d}(\frac{H}{h}) \|\mathcal{L}u\|_{L^2(\Omega)}\,; \tag{3.2.3}$$

$$e_0^{h,H,\infty}(a,u) \leq \frac{1}{a_{\min}} C_1(d)^2 H^2 \left(\rho_{2,d}(\frac{H}{h})\right)^2 \|\mathcal{L}u\|_{L^2(\Omega)}\,, \tag{3.2.4}$$

where, $C_1(d)$ is a constant that depends on the dimension d only, and for $p, d \geq 1$, the function $\rho_{p,d} : \mathbb{R}_+ \to \mathbb{R}_+$ is defined as:

$$\rho_{p,d}(t) = \begin{cases} 1, & d < p \\ (\log(1+t))^{\frac{d-1}{d}}, & d = p \\ t^{\frac{d-p}{p}}, & d > p\,. \end{cases} \tag{3.2.5}$$

In Theorem 3.2.1, we get the upper bound of $e_1^{h,H,\infty}(a,u)$ and $e_0^{h,H,\infty}(a,u)$. The dependence of this upper bound on h is determined by the function $\rho_{2,d}$. Note that it is a non-decreasing function, so as h decreases, for a fixed H, the ratio H/h increases, and the upper bound will also increase. One exception is when $d = 1$, the upper bound remains constant when h changes, and it is still finite even when h approaches 0. This phenomenon is in sharp contrast with the case $d \geq 2$, where as $h \to 0$, the upper bound blows up to infinity.

The above theoretical implications match what we have observed in the experiments—reducing h leads to a decrease of accuracy, both in $d = 1$ and $d = 2$; moreover, the deterioration of accuracy is more severe in $d = 2$ than $d = 1$.

Therefore, if one is adopting the ideal solution, without considering computational costs, then we would recommend choosing $h = H$, which achieves the best of both worlds with a theoretical guarantee and practical performance.

Remark 3.2.2. *Applying the above recommendation ($h = H$) is straightforward in the context of numerical upscaling—we can choose the suitable upscaled coarse variables. Nevertheless, for scattered data approximation, the data acquisition step also matters. Our analysis suggests that for the sake of accuracy (in the case there is no burden of computational costs), it could be a good idea to make the lengthscale of the coarse data larger; this provides guidance for data collection in such a scenario.*

3.2.3 Experiments: Localized Solution

Solving the ideal solution can be computationally expensive due to the global optimization problem (3.1.3). This is also why we stop at $H = 2^{-6}$ and do not decrease H further in the previous 2D experiments. For better practical algorithms, in this subsection, we move to the localized solution. We start with the numerical experiments for 1D and 2D, followed by theoretical analysis. In these experiments, we use the same functions $a(x)$ and $f(x)$ as in the ideal case.

In the localized scenario, the Galerkin solution in numerical upscaling and the recovery solution in scattered data approximation are different. Thus, we will compute them separately and compare the results. More precisely, for the Galerkin solution, we use the localized basis functions in the Galerkin framework to solve the PDE; for the recovery solution, it is simpler – once the basis functions are computed, we readily get the recovery solution by using the available subsampled data and the formula (3.1.6). For both cases, the ground truth solution u is given as a solution to a PDE.

3.2.3.1 One Dimensional Example

We consider the 1D model in Subsection 3.2.1.1. We compute the Galerkin errors $\tilde{e}_1^{h,H,l}(a,u)$ and $\tilde{e}_0^{h,H,l}(a,u)$ and the recovery errors $e_1^{h,H,l}(a,u)$ and $e_0^{h,H,l}(a,u)$, for $H = 2^{-2}, 2^{-3}, ..., 2^{-7}$, $h/H = 1, 1/2, 1/4, 1/8$ and $l = 2, 4$. The grid size we use to discretize the operator is set to be 2^{-11}.

In Figure 3.4, the oversampling parameter $l = 2$. The upper part depicts the energy and L^2 errors of the Galerkin solution, while the lower part corresponds to that of the recovery solution. From the figure, we observe the following facts:

- Due to localization, the error line of $h/H = 1, 1/2, 1/4$ finally turns up as we make H very small, deviating from what we have observed in the ideal solution. This implies the localization error matters a lot.
- Among the four choices, the case $h/H = 1/8$ that corresponds to the smallest h, behaves the best for small H. It appears that decreasing h may suppress the localization error to certain extent.
- The L^2 error of the recovery solution is more stable and accurate compared to the Galerkin solution, when H is small. Especially, there is no obvious blow-up as H becomes small.

Next, we increase the oversampling parameter to $l = 4$, and output the same set of observables in Figure 3.5. Now, only the case $h/H = 1$ leads to a turning up of the error line, while the other three cases lead to similar error lines as the ideal solution. The best choice among the four becomes $h/H = 1/2$. Thus, as l increases, the localized solution is approaching the ideal one, and choosing a larger h would be good.

3.2.3.2 Two Dimensional Example

In this subsection, we move to a two dimensional example that corresponds to the the ideal case in Subsection 3.2.1.2. As before, we compute the Galerkin errors $\tilde{e}_1^{h,H,l}(a,u)$ and $\tilde{e}_0^{h,H,l}(a,u)$ and the recovery errors $e_1^{h,H,l}(a,u)$ and $e_0^{h,H,l}(a,u)$, for $H = 2^{-2}, 2^{-3}, ..., 2^{-8}$, $h/H = 1, 3/4, 1/2, 1/4$ and $l = 2, 4$. The grid size we use to discretize the operator is set to be 2^{-10}.

We start with $l = 2$, in Figure 3.6. Our observations are as follows:

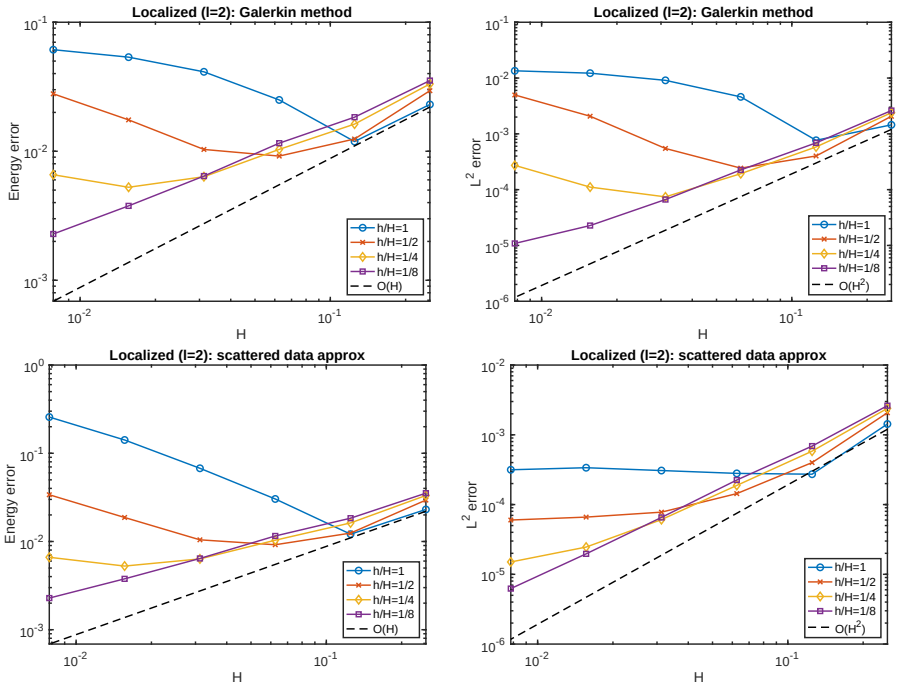


Figure 3.4: 1D example, localized solution $l = 2$. Upper left: $\tilde{e}_1^{h,H,l}(a,u)$; upper right: $\tilde{e}_0^{h,H,l}(a,u)$; lower left: $e_1^{h,H,l}(a,u)$; lower right: $e_0^{h,H,l}(a,u)$.

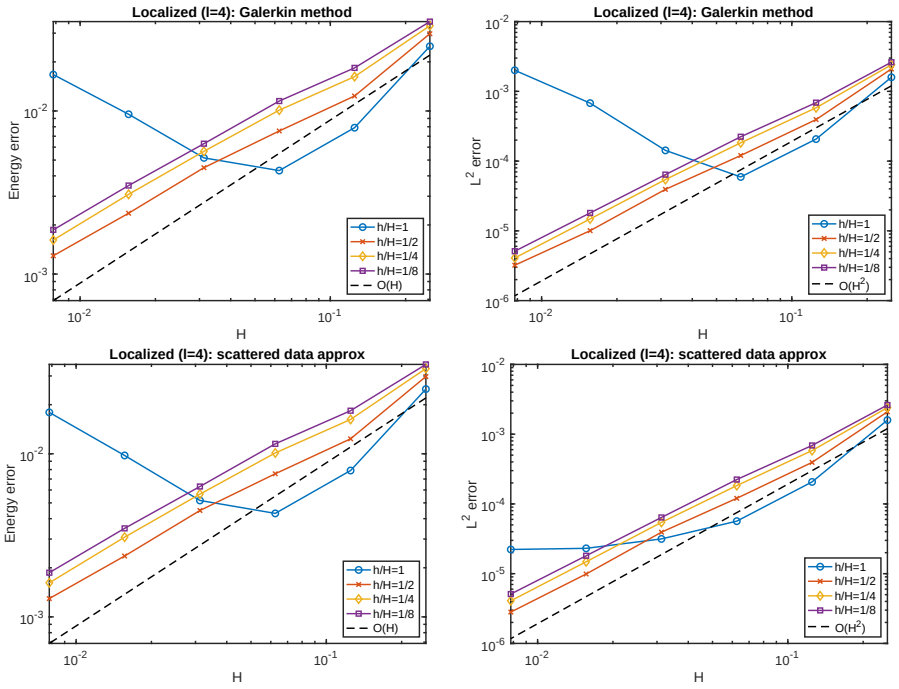


Figure 3.5: 1D example, localized solution $l = 4$. Upper left: $\tilde{e}_1^{h,H,l}(a,u)$; upper right: $\tilde{e}_0^{h,H,l}(a,u)$; lower left: $e_1^{h,H,l}(a,u)$; lower right: $e_0^{h,H,l}(a,u)$.

- All the error lines deviate from the desired $O(H)$ or $O(H^2)$ line to some extent, and among the four choices, the ratio $h/H = 1/2$ performs the best when H is small.
- Compared to the 1D example, the localization errors in 2D are larger, since the deviation from the desired $O(H)$ or $O(H^2)$ line is more apparent.
- The error line exhibits a turning up behavior even for very small $h/H = 1/4$. That means in the 2D case, small h can also lead to large overall errors. This observation indeed matches our theory for the ideal solution, as $\rho_{2,d}(H/h)$ in Theorem 3.2.1 will blow up as $h \to 0$, when $d = 2$.
- When H is small, the L^2 error of the recovery solution in the scattered data approximation is more accurate than the Galerkin solution in numerical upscaling. This phenomenon has also been observed in the 1D example.

Then, we increase the oversampling parameter to $l = 4$. The results are output in Figure 3.7. We observe a better accuracy and more stable behavior of the error lines compared to $l = 2$. Now the best among the four ratios becomes $h/H = 3/4$. Moreover, the relative behaviors of the three cases $h/H = 3/4, 1/2, 1/4$ are very similar to that in the ideal solution, indicating that when $l = 4$, the localization error may be small compared to the approximation error of the ideal solution.

3.2.4 Analysis: Localized Solution

In this subsection, we provide some theoretical analysis for the localized solution. To begin with, we summarize the main observations in the numerical experiments that we want to understand more deeply in our theoretical study.

1. The error lines of the localized solution, $e_1^{h,H,l}(a,u)$, $\tilde{e}_1^{h,H,l}(a,u)$ and also $\tilde{e}_0^{h,H,l}(a,u)$, turn up when H is small, if l is fixed;
2. The localization error appears to become smaller as h decreases; for the overall error of the localized solution, there seems to be a competition between the approximation error of the ideal solution (which increases as h decreases), and the localization error (which decreases as h decreases). The strength of the competition depends on the oversampling parameter l;
3. The L^2 error of the recovery solution is smaller compared to that of the Galerkin solution, i.e., $\tilde{e}_0^{h,H,l}(a,u)$ appears to be larger than $e_0^{h,H,l}(a,u)$, and for the latter, it does not blow up as H becomes small.

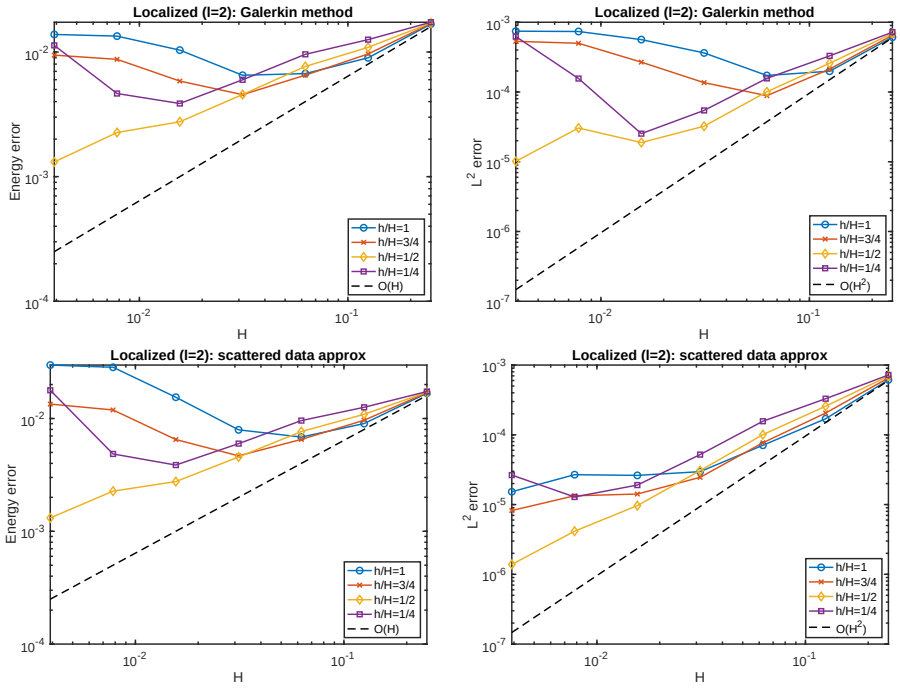


Figure 3.6: 2D example, localized solution $l = 2$. Upper left: $\tilde{e}_1^{h,H,l}(a,u)$; upper right: $\tilde{e}_0^{h,H,l}(a,u)$; lower left: $e_1^{h,H,l}(a,u)$; lower right: $e_0^{h,H,l}(a,u)$.

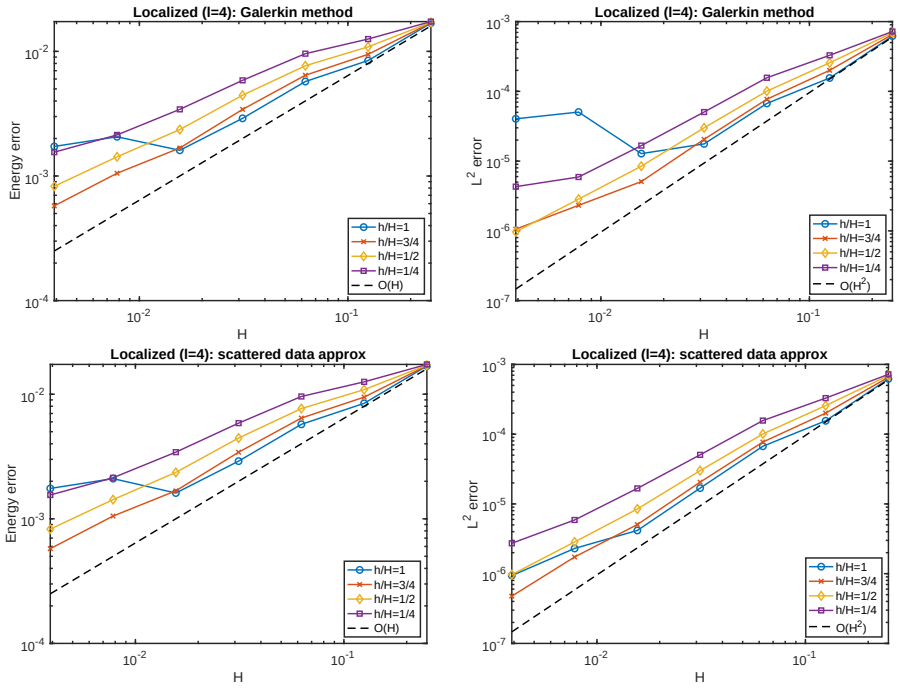


Figure 3.7: 2D example, localized solution $l = 4$. Upper left: $\tilde{e}_1^{h,H,l}(a,u)$; upper right: $\tilde{e}_0^{h,H,l}(a,u)$; lower left: $e_1^{h,H,l}(a,u)$; lower right: $e_0^{h,H,l}(a,u)$.

We will provide reasonable theoretical explanation of these observations. First, we introduce several useful notations.

3.2.4.1 Notations

For any function $v \in H_0^1(\Omega)$, we write

$$\mathsf{P}^{h,H,l} v = \sum_{i \in I} [v, \phi_i^{h,H}] \psi_i^{h,H,l} . \tag{3.2.6}$$

Moreover, we use the convention $\mathsf{P}^{h,H} v = \sum_{i \in I} [v, \phi_i^{h,H}] \psi_i^{h,H}$. These definitions lead to the relation $\mathsf{P}^{h,H,l} \psi_i^{h,H} = \psi_i^{h,H,l}$, which connects the ideal and localized basis functions.

Since we are mainly interested in how the error depends on h, H, l and u, we use $A \lesssim B$ (resp. $A \gtrsim B$) to denote the condition $A \leq CB$ (resp. $A \geq CB$) for some constant C independent of h, H, l and u. If we have both $A \lesssim B$ and $A \gtrsim B$, then we will write $A \simeq B$. We use $\langle \cdot, \cdot \rangle_a$ to denote the a-weighted inner product in $H_0^1(\Omega)$, i.e., $\langle u, v \rangle_a := \int_\Omega a \nabla u \cdot \nabla v$.

3.2.4.2 Analysis

To analyze the error of localized solutions, we first use the triangle inequality:

$$\begin{aligned} e_1^{h,H,l}(a,u) &= \|u - \mathsf{P}^{h,H,l} u\|_{H_a^1(\Omega)} \\ &\leq \|u - \mathsf{P}^{h,H} u\|_{H_a^1(\Omega)} + \|\mathsf{P}^{h,H} u - \mathsf{P}^{h,H,l} u\|_{H_a^1(\Omega)} \\ &\lesssim H \rho_{2,d}(\frac{H}{h}) \|\mathcal{L} u\|_{L^2(\Omega)} + \|\mathsf{P}^{h,H} u - \mathsf{P}^{h,H,l} u\|_{H_a^1(\Omega)}, \end{aligned} \tag{3.2.7}$$

where in the last inequality, we have used the estimate for the ideal solution. The second part $\|\mathsf{P}^{h,H} u - \mathsf{P}^{h,H,l} u\|_{H_a^1(\Omega)}$ is the localization error. Our main goal is to estimate this part of error. For this purpose, we have Theorem 3.2.3 below.

Theorem 3.2.3. *The following results hold:*

1. *(Inverse estimate) For any $v \in$ span $\{\psi_i^{h,H}\}_{i \in I}$ and in each $\omega_j^{h,H}$, $j \in I$, we have the estimate:*

$$\|\nabla \cdot (a \nabla v)\|_{L^2(\omega_j^{h,H})} \leq \frac{\sqrt{a_{\max}} C_2(d)}{h} \|v\|_{H_a^1(\omega_j^{h,H})},$$

where $C_2(d)$ is a constant that depends on d only.

2. *(Exponential decay) For each $i \in I$ and $k \in \mathbb{N}$, we have*

$$\|\psi_i^{h,H}\|^2_{H^1_a(\Omega\setminus \mathrm{N}^k(\omega_i^H))} \leq (\beta(h,H))^k \, \|\psi_i^{h,H}\|^2_{H^1_a(\Omega)}, \tag{3.2.8}$$

where

$$\beta(h,H) = \frac{C_0(d)\sqrt{\frac{a_{\max}}{a_{\min}}}\left(C_1(d)\rho_{2,d}(\frac{H}{h}) + C_1(d)C_2(d)\frac{h}{H}\right)}{C_0(d)\sqrt{\frac{a_{\max}}{a_{\min}}}\left(C_1(d)\rho_{2,d}(\frac{H}{h}) + C_1(d)C_2(d)\frac{h}{H}\right) + 1}. \tag{3.2.9}$$

Here, $C_0(d)$ is a universal constant dependent on d, $C_1(d)$ is the constant in Theorem 3.2.1 while $C_2(d)$ is the constant in the inverse estimate.

3. *(Norm estimate) Suppose for each $i \in I$, $\phi_i^{h,H}$ is L^1 normalized in the sense that $\|\phi_i^{h,H}\|_{L^1(\omega_i^{h,H})} = 1$, then the following estimate holds:*

$$\|\psi_i^{h,H}\|_{H^1_a(\Omega)} \lesssim \frac{1}{\rho_{2,d}(\frac{H}{h})} H^{d/2-1}. \tag{3.2.10}$$

4. *(Localization error per basis function) For each $i \in I$, it holds that*

$$\begin{aligned} &\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \\ &\lesssim H^{d/2-1} \min\left\{(\beta(h,H))^{l/2}, \frac{1}{\rho_{2,d}(\frac{H}{h})}\right\}. \end{aligned} \tag{3.2.11}$$

5. *(Overall localization error) The following error estimate holds:*

$$\begin{aligned} &\|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{H^1_a(\Omega)} \\ &\lesssim \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \times \min\left\{\frac{l^{d/2}}{H}, \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\} \|u\|_{L^\infty(\Omega)}. \end{aligned} \tag{3.2.12}$$

6. *(Overall recovery error) Suppose $d \leq 3$. For the energy recovery error, we have*

$$\begin{aligned} e_1^{h,H,l}(a,u) \lesssim \Bigg(&H\rho_{2,d}(\frac{H}{h}) + \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \\ &\times \min\left\{\frac{l^{d/2}}{H}, \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\}\Bigg) \|\mathcal{L}u\|_{L^2(\Omega)}, \end{aligned} \tag{3.2.13}$$

and for the L^2 recovery error, we have

$$e_0^{h,H,l}(a,u) \lesssim \left((H\rho_{2,d}(\frac{H}{h}))^2 + \min\left\{1, H\rho_{2,d}(\frac{H}{h})\right\} \times \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \times \min\left\{\frac{l^{d/2}}{H}, \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\}\right)\|\mathcal{L}u\|_{L^2(\Omega)}\,. \tag{3.2.14}$$

7. *(Overall Galerkin error) Suppose $d \leq 3$. The energy Galerkin error is upper bounded by the energy recovery error: $\tilde{e}_1^{h,H,l}(a,u) \leq e_1^{h,H,l}(a,u)$. For the L^2 Galerkin error, we have*

$$\tilde{e}_0^{h,H,l}(a,u) \lesssim \left(H\rho_{2,d}(\frac{H}{h}) + \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \times \min\left\{\frac{l^{d/2}}{H}, \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\}\right)^2\|\mathcal{L}u\|_{L^2(\Omega)}\,. \tag{3.2.15}$$

3.2.4.3 Implications

Before we move to the proof part, let us first discuss the implications of this theorem. We focus on the localization error in the final estimates.

- Fix an l and the ratio H/h. Due to (3.2.13) and (3.2.15), the localization error parts in $e_1^{h,H,l}(a,u)$, $\tilde{e}_1^{h,H,l}(a,u)$ and $\tilde{e}_0^{h,H,l}(a,u)$ will blow up as H goes to 0. In contrast, due to (3.2.14), the localization error in $e_0^{h,H,l}(a,u)$ remains bounded in this limit. Indeed, it is bounded by

$$\begin{aligned} &H\rho_{2,d}(\frac{H}{h}) \times (\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}) \times \frac{l^{d/2}}{H}\|\mathcal{L}u\|_{L^2(\Omega)} \\ \leq\; & l^{d/2}\,(\beta(h,H))^{l/2}\left(\rho_{2,d}(\frac{H}{h})\right)^2\|\mathcal{L}u\|_{L^2(\Omega)}\,, \end{aligned}$$

  which does not blow up as $H \to 0$. This reveals a distinguished behavior of $e_0^{h,H,l}(a,u)$ compared to the other three errors, which have been observed in our experiments. Our analysis explains this phenomenon.

- For $e_1^{h,H,l}(a,u)$, our analysis shows that there is a competition between the approximation error of the ideal solution, $H\rho_{2,d}(\frac{H}{h})$ (we omit $\|\mathcal{L}u\|_{L^2(\Omega)}$ for

simplicity), and the localization error

$$\min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}),1\right\}\times\min\left\{\frac{l^{d/2}}{H},\frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\}.$$

Fix an H and l. When $d\geq 2$, since $\lim_{h\to 0}\rho_{2,d}(\frac{H}{h})=\infty$, we have that as $h\to 0$, the approximation error goes to infinity, while the localization error goes to zero. When $d=1$, both two parts of errors remain bounded as $h\to 0$, and thus the competition is less pronounced; this matches what we have observed in our 1D experiments—the effect of reducing h is not as large as in our 2D example.

The existence of competition implies that in general, there should be a value of h that leads to the best error for the fixed H and l. Because the localization error decreases as l increases, this optimal value would also increase for a larger l, as observed in our experiments.

The above phenomenon also applies to other errors, i.e., the recover L^2 error $e_0^{h,H,l}(a,u)$ and the Galerkin errors $\tilde{e}_1^{h,H,l}(a,u)$ and $\tilde{e}_0^{h,H,l}(a,u)$.

- If we fix H/h, and want to have an overall error of $O(H)$ (for energy error) or $O(H^2)$ (for the L^2 error), then our estimates show that

$$l=O(\frac{\log H}{\log\beta(h,H)})$$

suffices for this goal. Note that $\beta(h,H)$ can be treated as a constant (less than 1) when H/h is fixed, so generally $l=O(\log(1/H))$ is enough. Moreover, our experiments demonstrate that we could do much better in practice—a constant value of $l=2$ or 4 behaves well for a wide range of H and h.

The three points above explain the questions that we raised at the beginning of Subsection 3.2.4.

Remark 3.2.4. *Though the presence of* 'min' *in many places of our estimates complicates the formula, they play critical roles in the above explanations, since we need to choose the correct term inside the* 'min' *to get the desired conclusion.*

Remark 3.2.5. *In Theorem 3.2.3, the basis function $\psi_i^{h,H}$ has an exponential decay property; see* (3.2.8)*. The localization error should heavily depend on the decay rate, so obtaining a tight bound of this rate is important here. In our analysis, we get the rate $\beta(h,H)$, which contains a term $\rho_{2,d}(H/h)$ that increases as h decreases*

(when $d \geq 2$), and a term h/H that decreases while h decreases. The two mixed components may suggest a non-monotone behavior of the decay rate. Moreover, when $h \to 0$, we get $\beta(h,H) \to 1$, so the decay appears to deteriorate eventually for small h. On the other hand, it seems intuitive that once h is small, the measurement region $\omega_i^{h,H}$ becomes more localized, and then the decay shall be amplified. To understand this problem better, we conduct a numerical experiment as follows. For the coefficient $a(x)$ in (3.2.2) and $H = 2^{-5}$, we compute the relative localization error $\frac{\|\psi_i^{h,H} - \psi_i^{h,H,l}\|^2_{H^1_a(\Omega)}}{\|\psi_i^{h,H}\|^2_{H^1_a(\Omega)}}$ for $h = 2^{-5}, 2^{-6}, \ldots, 2^{-10}$ and $l = 0, 1, 2, \ldots, 5$. The index i is selected so that ω_i^H is centered in the domain Ω. The result is shown in Fig. 3.8.

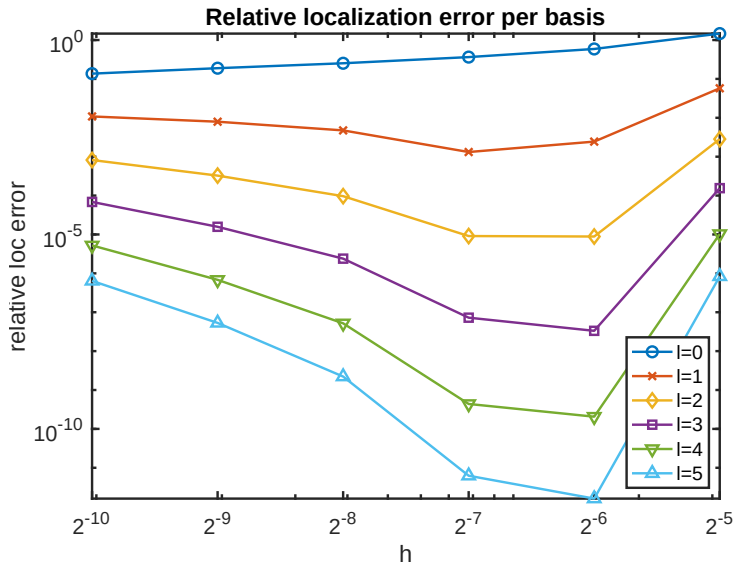


Figure 3.8: Relative localization error per basis function

From the figure, we observe that there is indeed a non-monotone behavior with respect to h in the relative localization error. Among these choices of h and l, we only see a monotone tendency for $l = 0$. For other l, the value h that leads to the minimal relative localization error increases as l increases. For the $a(x)$ and H considered, we can see $h/H = 1/2, 1/4$ lead to small errors in general, which also explains that this choice of h works quite well in our previous experiments. Overall, the above investigation suggests that our bound on the exponential decay and localization error can reasonably predict the behavior in practice. The decay is truly subtle regarding the small parameter h.

Remark 3.2.6. *Our current result does not provide explicit clues on how to choose h according to l and H to achieve the best accuracy. Nonetheless, our experiments have shown that usually $h/H = 3/4$ or $1/2$ behaves well, across a wide range of $H = 2^{-8}, 2^{-7}, ..., 2^{-2}$ and $l = 2, 4$, in the two dimensional problems. Providing more guidance on this aspect, either numerically or theoretically, is left as future work.*

3.2.4.4 Proof Strategy

The results in Theorem 3.2.3 are presented progressively. Our proofs will start from the first and move forward one by one to the seventh. We summarize the main ideas below, together with their connections to existing results in the literature. The detailed proof is in Subsection 3.4.1.

1. The inverse estimate is obtained due to a scaling argument—that is why the subsampled scale h appeared. (Subsection 3.4.1.1)

2. Based on the inverse estimate and the subsampled Poincaré inequality (see Proposition 2.5 in [44]), we can establish the exponential decay property via a Caccioppoli type of argument. The logical line of our proof here is similar to that of the original LOD method (Lemma 3.4 in [181]) and Gamblets (Theorem 3.9 in [203]), while now we need to be careful to make every estimate adaptive to the small scale parameter h. (Subsection 3.4.1.2)

3. For the norm estimate, we construct critical examples whose energy norm leads to a desired upper bound. The critical example here is similar to the one we used before to prove the optimality of the subsampled Poincaré inequality (see Proposition 2.6 in [44]). This type of profile has also been studied in the context of semi-supervised learning; see Theorem 2 in [193]. (Subsection 3.4.1.3)

4. The localization error per basis function is established by combining the exponential decay estimate and the norm estimate. Our results contain two parts inside the 'min' operation. The idea of proving the first part is similar to that of Lemma 3.4 in [181]. The second part is a direct application of the norm estimate. Both parts are important. The first part captures the exponential decay property, while the second part captures the behavior with respect to small h – when $d \geq 2$, this estimate implies the localization error per basis function vanishes as h goes to 0. (Subsection 3.4.1.4)

5. To move from the localization error per basis function to the overall localization error, we also proceed in two directions. The first one follows the idea of proving Lemma 3.5 in [181], leading to an upper bound of $O(l^{d/2}/H)$, which remains bounded as $h \to 0$. On the other hand, we can use simple triangle inequality, which yields an estimate of $O\left(1/\left(H^{d/2+1}\rho_{2,d}(\frac{H}{h})\right)\right)$, which is worse in the power of H than the first one, but can capture the limit as $h \to 0$, i.e., it vanishes as $h \to 0$. The combination of the two leads to the final estimate. (Subsection 3.4.1.5)

6. It is straightforward to go from overall localization error to the energy recovery error by a triangle inequality. For the L^2 recovery error, we can bound it through the energy error in two ways, with or without using the subsampled Poincaré inequality. This leads to a further 'min' operation in the final estimate. (Subsection 3.4.1.6)

7. The energy Galerkin error is upper bounded by the energy recover error according to the Galerkin orthogonality. The L^2 Galerkin error is obtained by the standard Aubin-Nitsche trick. (Subsection 3.4.1.7)

3.3 Small Limit Regime of Subsampled Lengthscales

In the last section, we have made a detailed study of the recovery error and Galerkin error with respect to h, H, and l. We observe that there is a deterioration of accuracy as h becomes small, especially for $d \geq 2$–the benefit of small localization errors by a very small h is overwhelmed by the curse of induced large approximation errors. Due to this reason, in our experiments, we choose the ratio h/H to be not too small: we select $h/H \geq 1/8$ in 1D and $h/H \geq 1/4$ in 2D. Our theoretical analysis also collaborates with these observations, as the function $\rho_{2,d}(H/h)$ that appears in the error estimate will blow up as $h/H \to 0$ for $d \geq 2$.

Therefore, we are advised not to use a very small h. While this is a practical suggestion in the problem of numerical upscaling, since we have the freedom of choosing the upscaled variables and thus can avoid this pathological phenomenon, in the problem of scattered data approximation, we may not have such flexibility due to the prevalent physical constraints for data measurements. As we often encounter recovery problems in high dimensions with scattered data that possibly have a very small lengthscale, e.g., pointwise data, it is natural to ask that whether we could get an accurate recovery even in the $h \to 0$ regime. The analysis above implies that this

goal is not achievable in general for the model problem we have considered. Thus, we need to put stronger assumptions on the function u to be approximated.

Since the degeneracy of accuracy for $d \geq 2$ can be partially attributed to the low regularity of the target function u, that is, when $d \geq 2$, functions in $H^1(\Omega)$ may not have a well-defined pointwise value (according to the Sobolev embedding theorem [83]), a natural idea is to assume u to be more regular. There has been some work in which u is assumed to be in $W^{k,2}(\Omega)$ for some larger k [302]; this assumption ensures the continuity of the function. Alternatively, one can assume $u \in W^{1,p}(\Omega)$ and increase p – when $p > d$, the degeneracy issue disappears; see [75, 252, 151, 37].

The above assumptions of better regularity on u, either via increasing k or p, require to modify the recovery algorithm substantially; in the former, the basis functions are obtained by replacing the $H^1_a(\Omega)$ norm in (3.1.3) by a high order norm, similar to the polyharmonic splines and their rough version [210]; in the latter, the recovery function is obtained by minimizing the $W^{1,p}(\Omega)$ norm subject to the observed data.

Here, to stick to the formulation (3.1.3) and thus the main theme of this chapter, we consider to improve the regularity via choosing a singular weight function $a(x)$. Naturally, in order to make the recovery non-degenerate regarding a vanishing h, we need to put more importance on the coarse data of a small lengthscale h. Thus, we could assume the function is "nearly flat" around the data location by using a singular $a(x)$ such that $\int_\Omega a|\nabla u|^2 < \infty$ – this guarantees the information content of coarse data even for very small h. We will make this intuition more quantitative in this section.

3.3.1 Numerical Experiment

As before, we start with some numerical experiment. We choose $d = 2$ and $\Omega = [0,1]^2$. The ground truth function u is depicted in the upper-left of Figure 3.9. The coarse scale $H = 2^{-2}$, and suppose for now we collect subsampled data with lengthscale $h = H/2 = 2^{-3}$; the grid size h_g is set to be 2^{-7}. In the upper-right of Figure 3.9, we plot the ideal recovery solution by using $a(x) = 1$, the subsampled data $[u, \phi_i^{h,H}], i \in I$ and the ideal basis functions $\{\psi_i^{h,H}\}_{i\in I}$. We observe that to certain extent, the recovery solution can capture the large scale property of u.

Then, we decrease the subsampled lengthscale – we choose $h = 2^{-4} \cdot H = 2^{-6}$. The recovery solution obtained by solving (3.1.3) with $a(x) = 1$ is in the lower-left of Figure 3.9. The degeneracy issue becomes apparent – there are many spikes in

the recovery solution, and the locations of these spikes are the data positions. This confirms our understanding that a small h leads to a degenerate recovery.

Now, we define a weight function as follows. For each local patch $\omega_i^H, i \in I$, its center is denoted by $x_i \in \omega_i^H$. We write $X^H = \bigcup_{i=1}^I \{x_i^H\}$ and $\mathsf{d}(x, X^H)$ is the Euclidean distance from x to the set X^H. The weight function is defined as

$$W(x) = \left(\frac{H}{\mathsf{d}(x, X^H)}\right) \log^2 \left(1 + \frac{H}{\mathsf{d}(x, X^H)}\right). \tag{3.3.1}$$

It is singular at the center of our subsampled data; see Figure 3.10. In the lower-right of Figure 3.9, we construct the recovery solution by solving (3.1.3) with $a(x) = W(x)$. To avoid numerical instability in the experiment, we use a regularized version of the singular weight as follows:

$$W(x; h_g) = \left(\frac{H}{\max\{h_g, \mathsf{d}(x, X^H)\}}\right) \log^2 \left(1 + \frac{H}{\max\{h_g, \mathsf{d}(x, X^H)\}}\right), \tag{3.3.2}$$

where h_g is the grid size. From the figure, we observe that the recovery solution appears much better than the one based on $a(x) = 1$. It captures most of the large scale behaviors. Moreover, it is visually smoother; due to the singular weight function, the impact of the subsampled data does propagate to other points in the domain.

Remark 3.3.1. *The idea of function recovery based on a weight function that puts more importance around the data regions has been used in semisupervised learning and image processing [248], through using a weighted graph Laplacian. Recently, the work [38] proposed a properly weighted Laplacian that attains a well-defined continuous limit. Our earlier work [44] also discussed a similar weighted discovery. In the next subsection, we will provide some theoretical analysis of this recovery based on results in [44], assuming $u(x)$ belongs to a weighted function space.*

3.3.2 Analysis: Weighted Inequality

For simplicity, in dimension $d \geq 2$, we consider the following class of weight functions:

$$W_{\gamma,H}(x) = \left(\frac{H}{\mathsf{d}(x, X^H)}\right)^{d-2+\gamma}, \tag{3.3.3}$$

where $\gamma > 0$. Indeed, the additional log term in (3.3.1) only makes the problem easier, since it makes the function blow up even faster.

We use the same notation as in Subsection 3.2.4.1. Then, we have the following theorem:

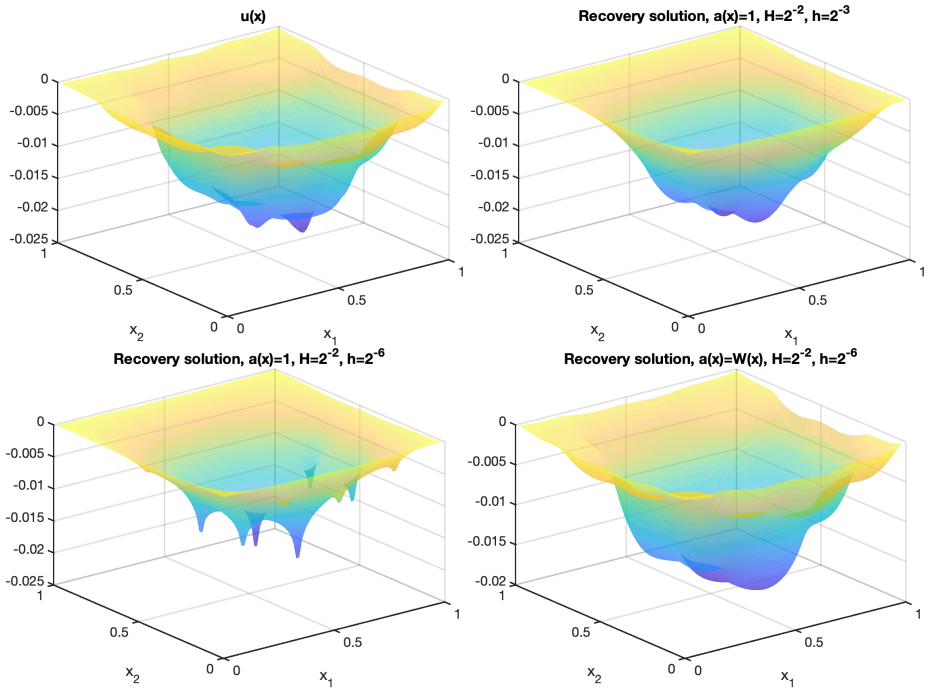


Figure 3.9: Upper left: $u(x)$; upper right: recovery solution, $h/H = 1/2$ and $a(x) = 1$; lower left: recovery solution, $h/H = 1/2^4$ and $a(x) = 1$; lower right: recovery solution, $h/H = 1/2^4$ and $a(x) = W(x)$.

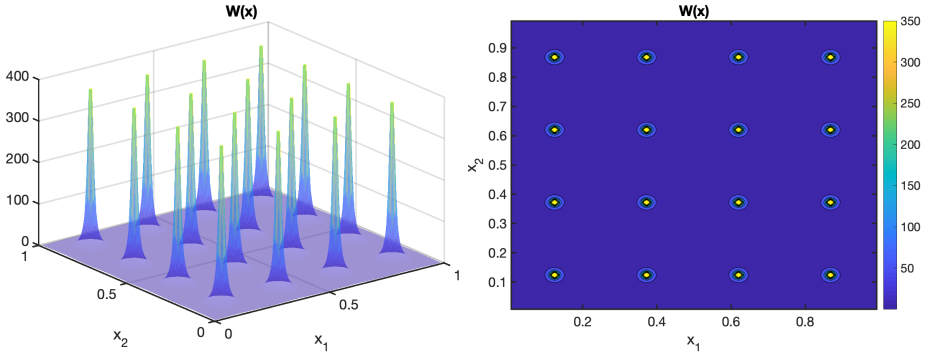


Figure 3.10: Left: figure of $W(x)$; right: contour of $W(x)$

Theorem 3.3.2. *Let $d \geq 2$ and $\gamma > 0$. Fix an H, and we choose $a(x) = W_{\gamma,H}(x)$. Then the following results hold:*

1. *If $\|u\|_{H^1_a(\Omega)} < \infty$, then the L^2 error of the ideal solution satisfies*

$$e_0^{h,H,\infty}(a,u) \lesssim C(\gamma)H\|u\|_{H^1_a(\Omega)}\,. \tag{3.3.4}$$

2. *If $-\nabla\cdot(a\nabla u) = f \in L^2(\Omega)$, then the energy error of the ideal solution satisfies*

$$e_1^{h,H,\infty}(a,u) \lesssim C(\gamma)H\|f\|_{L^2(\Omega)}\,, \tag{3.3.5}$$

and the L^2 error satisfies

$$e_0^{h,H,\infty}(a,u) \lesssim C(\gamma)H^2\|f\|_{L^2(\Omega)}\,. \tag{3.3.6}$$

Here, $C(\gamma)$ represents a positive constant that depends on γ only, and can vary its value from place to place.

The proof is deferred to Subsection 3.4.2. We observe from the theorem that the upper bound of the accuracy is independent of the subsampled scale h, which implies that it is still valid in the small h limit. This is in sharp contrast with the estimates in Theorem 3.2.1, where the upper bound blows up as $h \to 0$. The key here is the use of a singular weight function that puts more importance on the subsampled data.

We also use a numerical experiment to demonstrate this theorem. We choose $d = 2$, $\Omega = [0,1]^2$ and $H = 2^{-2}$. The parameter $\gamma = 1$. We use the mechanism in Subsection 3.2.1.2 to generate a right-hand side $f \in L^2(\Omega)$, and u solves

$$-\nabla\cdot(W_{\gamma,H}\nabla u) = f\,.$$

The grid size is set to be 2^{-8}. We choose $h = 2^{-3}, 2^{-4}, ..., 2^{-7}$. For each h, we collect the data $[u, \phi_i^{h,H}], i \in I$ and compute the ideal recovery solutions by solving (3.1.3) with $a(x) = 1$ and $a(x) = W_{\gamma,H}(x)$ respectively. We output the $H_0^1(\Omega)$ and $L^2(\Omega)$ error of these recovery solutions in Figure 3.11. From this figure, we observe

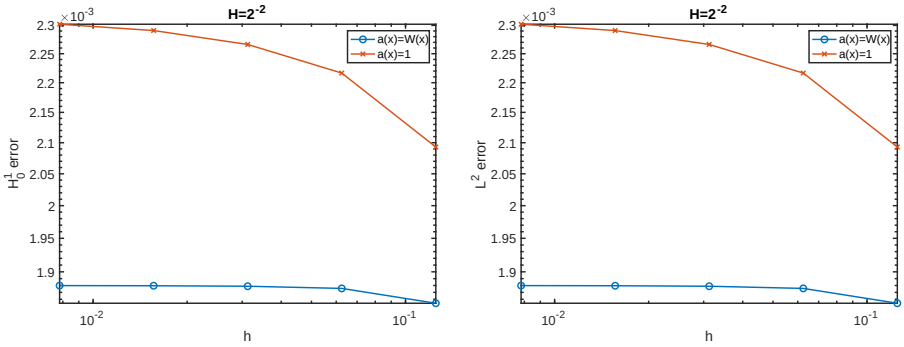


Figure 3.11: The $H_0^1(\Omega)$ and $L^2(\Omega)$ errors for different h, using constant $a(x)$ or singular weighted $a(x)$. Left: $H_0^1(\Omega)$ error; right: $L^2(\Omega)$ error

that the recovery errors using $a(x) = 1$ will increase as h decrease, while those using $a(x) = W_{\gamma,H}(x)$ lead to a flattened curve with respect to h. This matches our theoretical predictions. Since in this example the dimension $d = 2$, the blow-up rate predicted by Theorem 3.2.1 is only logarithmic, so even though h is very small, the overall accuracy is still not too bad.

3.4 Proofs

This section provides all the proofs in this chapter.

3.4.1 Proof of Theorem 3.2.3

There are seven sub-results in this theorem. We prove them one by one.

3.4.1.1 Inverse Estimate

In the domain $\omega_j^{h,H}$, we have $\nabla \cdot (a\nabla v) = c_i \phi_i^{h,H}$ for some $c_i \in \mathbb{R}$. Let $v = v_1 + v_2$ such that

$$\nabla \cdot (a\nabla v_1) = \nabla \cdot (a\nabla v) = c_i \phi_i^{h,H} \text{ in } \omega_j^{h,H}, \quad v_1|_{\partial\omega_j^{h,H}} = 0,$$

and for the second part,

$$\nabla \cdot (a\nabla v_2) = 0 \text{ in } \omega_j^{h,H}, \quad v_2|_{\partial\omega_j^{h,H}} = v|_{\partial\omega_j^{h,H}}.$$

We have the orthogonality: $\int_{\omega_j^{h,H}} a\nabla v_1 \cdot \nabla v_2 = 0$. Thus, it holds that

$$\|v\|_{H_a^1(\omega_j^{h,H})} \geq \|v_1\|_{H_a^1(\omega_j^{h,H})}. \tag{3.4.1}$$

For v_1, we use the elliptic estimate:

$$\|v_1\|_{H_a^1(\omega_j^{h,H})} \geq \frac{1}{\sqrt{a_{\max}}} \|\nabla \cdot (a\nabla v_1)\|_{H^{-1}(\omega_j^{h,H})} = \frac{1}{\sqrt{a_{\max}}} \|c_j \phi_j^{h,H}\|_{H^{-1}(\omega_j^{h,H})}.$$

By a scaling argument, we obtain

$$\|\phi_j^{h,H}\|_{L^2(\omega_j^{h,H})} \leq \frac{C_2(d)}{h} \|\phi_j^{h,H}\|_{H^{-1}(\omega_j^{h,H})},$$

for a constant $C_2(d)$ dependent on d. Then, it follows that

$$\|v_1\|_{H_a^1(\omega_j^{h,H})} \geq \frac{h}{\sqrt{a_{\max}} C_2(d)} \|c_j \phi_j^{h,H}\|_{L^2(\omega_j^{h,H})} = \frac{h}{\sqrt{a_{\max}} C_2(d)} \|\nabla \cdot (a\nabla v)\|_{L^2(\omega_j^{h,H})}. \tag{3.4.2}$$

Combining (3.4.1) and (3.4.2), we arrive at the desired result:

$$\|\nabla \cdot (a\nabla v)\|_{L^2(\omega_j^{h,H})} \leq \frac{\sqrt{a_{\max}} C_2(d)}{h} \|v\|_{H_a^1(\omega_j^{h,H})}.$$

3.4.1.2 Exponential Decay

Fix $i \in I$. For ease of notations, we will write $\psi_i^{h,H}$ by ψ, and $\mathrm{N}^k(\omega_i^H)$ by S_k in this proof.

First, we choose a cut-off function η with value 0 in S_k and value 1 in S_{k+1}^c such that it satisfies $\eta \geq 0$ and $\|\nabla\eta\|_\infty \leq C_0(d)/H$ for some universal constant $C_0(d)$ dependent on d. An example of η could be

$$\eta(x) = \frac{\operatorname{dist}(x, S_k)}{\operatorname{dist}(x, S_k) + \operatorname{dist}(x, S_{k+1}^c)} .$$

Then, we obtain the relation:

$$\|\psi\|^2_{H^1_a(\Omega\setminus S_{k+1})} = \int_{\Omega\setminus S_{k+1}} \nabla\psi \cdot a\nabla\psi \leq \int_\Omega \eta\nabla\psi \cdot a\nabla\psi . \tag{3.4.3}$$

Using some algebra, we have

$$\begin{aligned}\eta\nabla\psi \cdot a\nabla\psi &= \nabla(\eta\psi) \cdot a\nabla\psi - (\nabla\eta) \cdot a\psi\nabla\psi \\ &= \nabla \cdot (\eta\psi a\nabla\psi) - \eta\psi\nabla \cdot (a\nabla\psi) - (\nabla\eta) \cdot a\psi\nabla\psi .\end{aligned}$$

Integrating the above formula in Ω and applying the divergence theorem yields

$$\int_\Omega \eta\nabla\psi \cdot a\nabla\psi \leq \left|\int_\Omega -\eta\psi\nabla \cdot (a\nabla\psi)\right| + \left|\int_\Omega (\nabla\eta) \cdot \psi a\nabla\psi\right| . \tag{3.4.4}$$

For the first term in (3.4.4), we have

$$\begin{aligned}\int_\Omega -\eta\psi\nabla \cdot (a\nabla\psi) &\overset{(a)}{=} \sum_{\omega_j^H \subset S_{k+1}\setminus S_k} \int_{\omega_j^H} -\eta\psi\nabla \cdot (a\nabla\psi) \\ &\overset{(b)}{=} \sum_{\omega_j^H \subset S_{k+1}\setminus S_k} \int_{\omega_j^{h,H}} -\eta\psi\nabla \cdot (a\nabla\psi) \\ &\overset{(c)}{=} \sum_{\omega_j^H \subset S_{k+1}\setminus S_k} \int_{\omega_j^{h,H}} -\big(\eta - \eta(x_j)\big)\,\psi\nabla \cdot (a\nabla\psi) \\ &\overset{(d)}{\leq} \sum_{\omega_j^H \subset S_{k+1}\setminus S_k} \frac{C_0(d)h}{H} \|\psi\|_{L^2(\omega_j^{h,H})} \|\nabla \cdot (a\nabla\psi)\|_{L^2(\omega_j^{h,H})} ,\end{aligned} \tag{3.4.5}$$

where

- in (a), we have used the fact that η is supported in $\Omega\setminus S_k$; moreover, in $\Omega\setminus S_{k+1}$, $\eta = 1$ and $\nabla \cdot (a\nabla\psi) = \sum_j c_j \phi_j^{h,H}$ for some $c_j \in \mathbb{R}$, and we have relied on the property $\int_{\omega_j^H} \phi_j^{h,H}\psi = 0$ for $\omega_j^H \in \Omega\setminus S_{k+1}$;
- in (b), we have used the fact that $\phi_j^{h,H}$ is supported in $\omega_j^{h,H}$;
- in (c), we have relied on the fact $\int_{\omega_j^{h,H}} \phi_j^{h,H}\psi = 0$ for $\omega_j^{h,H} \in \Omega\setminus S_k$ so we can subtract η by the constant $\eta(x_j)$ for x_j being the center of $\omega_j^{h,H}$;

- in (d) we have used the gradient bound on η and the Cauchy-Schwarz inequality.

For the term $\|\nabla \cdot (a\nabla\psi)\|_{L^2(\omega_j^{h,H})}$, we apply the inverse estimate established earlier, which leads to

$$\begin{aligned}
(3.4.5) &\leq \frac{C_0(d)h}{H}\frac{\sqrt{a_{\max}}C_2(d)}{h}\sum_{\omega_j^H\subset S_{k+1}\setminus S_k}\|\psi\|_{L^2(\omega_j^{h,H})}\|\psi\|_{H_a^1(\omega_j^{h,H})}\\
&\overset{(e)}{\leq} \frac{C_0(d)h}{H}\sqrt{a_{\max}}C_2(d)C_1(d)\sum_{\omega_j^H\subset S_{k+1}\setminus S_k}\|\nabla\psi\|_{L^2(\omega_j^{h,H})}\|\psi\|_{H_a^1(\omega_j^{h,H})}\\
&\overset{(f)}{\leq} \frac{C_0(d)C_1(d)C_2(d)h\sqrt{a_{\max}}}{H}\|\nabla\psi\|_{L^2(S_{k+1}\setminus S_k)}\|\psi\|_{H_a^1(S_{k+1}\setminus S_k)}\\
&\leq \frac{C_0(d)C_1(d)C_2(d)h}{H}\sqrt{\frac{a_{\max}}{a_{\min}}}\|\psi\|^2_{H_a^1(S_{k+1}\cap S_k^c)},
\end{aligned}$$

where in (e), we have used the Poincaré inequality, based on the fact $\int_{\omega_j^{h,H}}\psi\phi_j^{h,H}=0$. The constant in the Poincaré inequality can be chosen the same as the one in Theorem 3.2.1, i.e., $C_1(d)$; for details see Proposition 2.5 and Theorem 3.3 in [44]. The step (f) is by the Cauchy-Schwarz inequality.

For the second term in (3.4.4), we have

$$\begin{aligned}
\int_\Omega(\nabla\eta)\cdot\psi a\nabla\psi &= \int_{S_{k+1}\setminus S_k}(\nabla\eta)\cdot\psi a\nabla\psi\\
&= \sum_{\omega_j^H\subset S_{k+1}\setminus S_k}\int_{\omega_j^H}(\nabla\eta)\cdot\psi a\nabla\psi\\
&\leq \frac{C_0(d)\sqrt{a_{\max}}}{H}\sum_{\omega_j^H\subset S_{k+1}\setminus S_k}\|\psi\|_{L^2(\omega_j^H)}\|\psi\|_{H_a^1(\omega_j^H)}\\
&\overset{(g)}{\leq} \frac{C_0(d)\sqrt{a_{\max}}}{H}\sum_{\omega_j^H\subset S_{k+1}\setminus S_k}H\rho_{2,d}(\frac{H}{h})C_1(d)\|\nabla\psi\|_{L^2(\omega_j^H)}\|\psi\|_{H_a^1(\omega_j^H)}\\
&\leq C_0(d)C_1(d)\rho_{2,d}(\frac{H}{h})\sqrt{\frac{a_{\max}}{a_{\min}}}\|\psi\|^2_{H_a^1(S_{k+1}\setminus S_k)},
\end{aligned}$$

where in step (g), we have used the subsampled Poincaré inequality (Proposition 2.5 in [44]) and the fact $\int_{\omega_j^H}\phi_j^{h,H}\psi=0$.

Combining the estimates of the two terms and (3.4.3), we get

$$\|\psi\|^2_{H_a^1(\Omega\setminus S_{k+1})}\leq C_0(d)(C_1(d)\rho_{2,d}(\frac{H}{h})+C_1(d)C_2(d)\frac{h}{H})\sqrt{\frac{a_{\max}}{a_{\min}}}\|\psi\|^2_{H_a^1(S_{k+1}\setminus S_k)}.$$

Writing $\|\psi\|^2_{H^1_a(S_{k+1}\setminus S_k)} = \|\psi\|^2_{H^1_a(\Omega\setminus S_k)} - \|\psi\|^2_{H^1_a(\Omega\setminus S_{k+1})}$, we then arrive at

$$\|\psi\|^2_{H^1_a(\Omega\setminus S_{k+1})} \leq \beta(h,H)\|\psi\|^2_{H^1_a(\Omega\setminus S_k)} \leq \ldots \leq (\beta(h,H))^{k+1}\,\|\psi\|^2_{H^1_a(\Omega)}\,,$$

where

$$\beta(h,H) = \frac{C_0(d)\sqrt{\frac{a_{\max}}{a_{\min}}}\left(C_1(d)\rho_{2,d}(\frac{H}{h}) + C_1(d)C_2(d)\frac{h}{H}\right)}{C_0(d)\sqrt{\frac{a_{\max}}{a_{\min}}}\left(C_1(d)\rho_{2,d}(\frac{H}{h}) + C_1(d)C_2(d)\frac{h}{H}\right) + 1}\,.$$

3.4.1.3 Norm Estimate

Let us recall the definition of $\psi_i^{h,H}$ and $\psi_i^{h,H,l}$ for $l = 0$:

$$\begin{aligned}\psi_i^{h,H} = \text{argmin}_{\psi\in H^1_0(\Omega)} \quad & \|\psi\|^2_{H^1_a(\Omega)} \\ \text{subject to} \quad & [\psi,\phi_j^{h,H}] = \delta_{i,j} \ \text{ for } \ j\in I\,.\end{aligned} \tag{3.4.6}$$

$$\begin{aligned}\psi_i^{h,H,0} = \text{argmin}_{\psi\in H^1_0(\omega_i^H)} \quad & \|\psi\|^2_{H^1_a(\omega_i^H)} \\ \text{subject to} \quad & [\psi,\phi_i^{h,H}] = 1\,.\end{aligned} \tag{3.4.7}$$

Clearly, $\|\psi_i^{h,H}\|_{H^1_a(\Omega)} \leq \|\psi_i^{h,H,0}\|_{H^1_a(\omega_i^H)}$ so it suffices to estimate the latter. Without loss of generality, we can assume ω_i^H is centered at 0, so that $\omega_i^{h,H} = [-h/2,h/2]^d$ and $\omega_i^H = [-H/2,H/2]^d$.

First, we choose $v\in H^1_0(\omega_i^H)$ to be a cut-off function that equals 1 in $[-H/4,H/4]^d$ and equals 0 outside ω_i^H. Moreover, $v\geq 0$ and $\|\nabla v\|_\infty \lesssim 1/H$. Then, we have

$$[v,\phi_i^{h,H}] = \frac{1}{h^d}\int_{[-h/2,h/2]^d} v \simeq 1\,,$$

and

$$\|v\|^2_{H^1_a(\omega_i^H)} \lesssim \int_{\omega_i^H}|\nabla v|^2 \lesssim H^d\cdot\frac{1}{H^2} \lesssim H^{d-2}\,.$$

Define $w = v/[v,\phi_i^{h,H}]$, then w satisfies the constraint in (3.4.7), and $\|w\|_{H^1_a(\omega_i^H)} \lesssim H^{d/2-1}$, which leads to $\|\psi_i^{h,H,0}\|_{H^1_a(\omega_i^H)} \lesssim H^{d/2-1}$. Thus, the case $d=1$ is proved.

Second, we deal with the case $d=2$. Suppose $h\leq H/2$, and we choose

$$v(x) = \begin{cases} 1 - \dfrac{\log\left(1+\frac{4|x|}{h}\right)}{\log\left(1+\frac{H}{h}\right)}, & |x|\leq\dfrac{H}{4} \\ 0, & |x|>\dfrac{H}{4}\,. \end{cases}$$

We have $v(x) \leq 1$, and for $|x| \leq h/4$, $v(x) \geq 1 - \frac{\log(2)}{\log(3)} \gtrsim 1$. Therefore, it holds that

$$[v, \phi_i^{h,H}] = \frac{1}{h^d} \int_{[-h/2,h/2]^d} v \simeq 1 \,.$$

Then, we calculate the energy norm of v as follows:

$$\begin{aligned} \|v\|^2_{H^1_a(\omega_i^H)} &\lesssim \frac{1}{\log^2(1+\frac{H}{h})} \int_{B(0,H/4)} \left(\frac{1}{h+4|x|}\right)^2 \mathrm{d}x \\ &\lesssim \frac{1}{\log^2(1+\frac{H}{h})} \int_0^{H/4} \frac{r}{(4r+h)^2} \,\mathrm{d}r \,. \end{aligned}$$

We write $\int_0^{H/4} \frac{r}{(4r+h)^2} \,\mathrm{d}r = \int_0^{h/2} \frac{r}{(4r+h)^2} \,\mathrm{d}r + \int_{h/2}^{H/4} \frac{r}{(4r+h)^2} \,\mathrm{d}r \lesssim \int_0^{h/2} \frac{1}{h} \,\mathrm{d}r + \int_{h/2}^{H/4} \frac{1}{r} \,\mathrm{d}r$ $\lesssim \log\big(1+\frac{H}{h}\big)$. Thus, it follows that

$$\|v\|_{H^1_a(\omega_i^H)} \lesssim \left(\frac{1}{\log\big(1+\frac{H}{h}\big)}\right)^{1/2} = \frac{1}{\rho_{2,d}(\frac{H}{h})} \,.$$

This concludes the proof for the case $h \leq H/2$. When $h > H/2$, we use the result in the first step $\|v\|_{H^1_a(\omega_i^H)} \lesssim H^{d/2-1} \lesssim 1 \lesssim \frac{1}{\rho_{2,d}(\frac{H}{h})}$. The case $d = 2$ is proved.

Finally, when $d \geq 3$, we choose v in a similar fashion as in the first step, such that $v = 1$ in $[-h/4, h/4]^d$ and $v = 1$ outside $[-h/2, h/2]^d$. Moreover, $v \geq 0$ and $\|\nabla v\|_\infty \lesssim 1/h$. Following the same argument in the first step, we will arrive at

$$\|\psi_i^{h,H,0}\|_{H^1_a(\omega_i^H)} \lesssim h^{d/2-1} = \frac{1}{\rho_{2,d}(\frac{H}{h})} H^{d/2-1} \,,$$

which completes the proof.

3.4.1.4 Localization Per Basis Function

We define a space

$$V^{h,H} := \{v \in H^1_0(\Omega) : [v, \phi_j^{h,H}] = 0, j \in I\} \,.$$

Then, by the optimality of $\psi_i^{h,H}$ and $\psi_i^{h,H,l}$ in their corresponding optimization problems, we have $\left\langle \psi_i^{h,H}, v \right\rangle_a = 0$ for any $v \in V^{h,H}$ and $\left\langle \psi_i^{h,H,l}, v \right\rangle_a = 0$ for any $v \in V^{h,H} \bigcap H^1_0(\mathrm{N}^l(\omega_i^H))$. Thus, $\left\langle \psi_i^{h,H} - \psi_i^{h,H,l}, v \right\rangle_a = 0$ for any $v \in V^{h,H} \bigcap H^1_0(\mathrm{N}^l(\omega_i^H))$.

Then, we define $\chi_i^{h,H} = \psi_i^{h,H} - \psi_i^{h,H,0}$ and $\chi_i^{h,H,l} = \psi_i^{h,H,l} - \psi_i^{h,H,0}$. We have $\psi_i^{h,H} - \psi_i^{h,H,l} = \chi_i^{h,H} - \chi_i^{h,H,l}$ and $\chi_i^{h,H,l} \in V^{h,H} \bigcap H^1_0(\mathrm{N}^l(\omega_i^H))$.

Based on the above fact and the orthogonality, we get

$$\begin{aligned}\|\psi_i^{h,H} - \psi_i^{h,H,l}\|^2_{H^1_a(\Omega)} &= \|\chi_i^{h,H} - \chi_i^{h,H,l}\|^2_{H^1_a(\Omega)} \\ &\leq \|\chi_i^{h,H} - v\|^2_{H^1_a(\Omega)},\end{aligned} \tag{3.4.8}$$

for any $v \in V^{h,H} \bigcap H^1_0(\mathrm{N}^l(\omega_i^H))$. We take

$$v = \eta\chi_i^{h,H} - \mathsf{P}^{h,H,0}(\eta\chi_i^{h,H}),$$

where η is a cut-off function that equals 1 in $\mathrm{N}^{l-1}(\omega_i^H)$ and equals 0 outside $\mathrm{N}^l(\omega_i^H)$. Moreover, $\eta \geq 0$ and $\|\nabla\eta\|_\infty \lesssim 1/H$. This v belongs to $V^{h,H} \bigcap H^1_0(\mathrm{N}^l(\omega_i^H))$ because both $\eta\chi_i^{h,H}$ and $\mathsf{P}^{h,H,0}(\eta\chi_i^{h,H})$ belong to $H^1_0(\mathrm{N}^l(\omega_i^H))$, and by definition, $[\eta\chi_i^{h,H} - \mathsf{P}^{h,H,0}(\eta\chi_i^{h,H}), \phi_j^{h,H}] = 0, j \in I$. Then, it follows that

$$\begin{aligned}\|\chi_i^{h,H} - v\|^2_{H^1_a(\Omega)} &= \|(1-\eta)\chi_i^{h,H} - \mathsf{P}^{h,H,0}\left(\eta\chi_i^{h,H}\right)\|^2_{H^1_a(\Omega)} \\ &= \|(1-\eta)\chi_i^{h,H} - \mathsf{P}^{h,H,0}\left((1-\eta)\chi_i^{h,H}\right)\|^2_{H^1_a(\Omega)},\end{aligned} \tag{3.4.9}$$

where we have used the fact $\mathsf{P}^{h,H,0}\chi_i^{h,H} = 0$. To move further, we need to use the following Lemma:

Lemma 3.4.1. *The operator $\mathsf{P}^{h,H,0}$ is stable under the norm $\|\cdot\|_{H^1_a(\Omega)}$. More precisely, we have for any $w \in H^1_0(\Omega)$, it holds*

$$\|\mathsf{P}^{h,H,0}w\|_{H^1_a(\Omega)} \lesssim \|w\|_{H^1_a(\Omega)}.$$

Proof of Lemma 3.4.1. By definition, $\psi_i^{h,H,0}$ is supported in ω_i^H, and $\mathsf{P}^{h,H,0}w = \sum_{i\in I}[w, \phi_i^{h,H}]\psi_i^{h,H,0}$. Thus, we have

$$\begin{aligned}\|w - \mathsf{P}^{h,H,0}w\|^2_{H^1_a(\Omega)} &= \sum_{i\in I}\int_{\omega_i^H} a\left|\nabla(w - [w,\phi_i^{h,H}]\psi_i^{h,H,0})\right|^2 \\ &\leq \sum_{i\in I}\int_{\omega_i^H} a\,|\nabla w|^2 = \|w\|^2_{H^1_a(\Omega)},\end{aligned} \tag{3.4.10}$$

where we have used the fact that in each ω_i^H, it holds

$$\int_{\omega_i^H} a\nabla(w - [w,\phi_i^{h,H}]\psi_i^{h,H,0}) \cdot \nabla\psi_i^{h,H,0} = 0,$$

according to the definition of $\psi_i^{h,H,0}$. Equation (3.4.10) implies $\mathsf{P}^{h,H,0}$ is stable. □

Using Lemma 3.4.1, we proceed as follows:

$$\begin{aligned}(3.4.9) &\lesssim \|(1-\eta)\chi_i^{h,H}\|^2_{H^1_a(\Omega)} \\ &= \int_{S_l\setminus S_{l-1}} a^2|(\nabla\eta)\chi_i^{h,H}|^2 + \int_{S_l\setminus S_{l-1}} a^2|\eta\nabla\chi_i^{h,H}| + \|\chi_i^{h,H}\|^2_{H^1_a(\Omega\setminus S_l)},\end{aligned} \tag{3.4.11}$$

where we have used the notation $S_l = \mathrm{N}^l(\omega_i^H)$. For the first term in (3.4.11), we have

$$\begin{aligned}\int_{S_l\setminus S_{l-1}} a^2|(\nabla\eta)\chi_i^{h,H}|^2 &= \sum_{\omega_j^H\subset S_l\setminus S_{l-1}} \int_{\omega_j^H} a^2|(\nabla\eta)\chi_i^{h,H}|^2 \\ &\lesssim \sum_{\omega_j^H\subset S_l\setminus S_{l-1}} \frac{1}{H^2}\cdot H^2\left(\rho_{2,d}(\frac{H}{h})\right)^2 \|\chi_i^{h,H}\|^2_{H^1_a(\omega_j^H)} \\ &= \left(\rho_{2,d}(\frac{H}{h})\right)^2 \|\chi_i^{h,H}\|^2_{H^1_a(S_l\setminus S_{l-1})}.\end{aligned} \tag{3.4.12}$$

In the above inequality, we have used the gradient bound of η, the subsampled Poincare inequality (due to the property $[\chi_i^{h,H}, \phi_j^{h,H}] = 0$). Therefore, we obtain

$$\begin{aligned}(3.4.11) &\lesssim (1+\left(\rho_{2,d}(\frac{H}{h})\right)^2)\|\chi_i^{h,H}\|^2_{H^1_a(S_l\setminus S_{l-1})} + \|\chi_i^{h,H}\|^2_{H^1_a(\Omega\setminus S_l)} \\ &\lesssim (1+\left(\rho_{2,d}(\frac{H}{h})\right)^2)\|\chi_i^{h,H}\|^2_{H^1_a(\Omega\setminus S_{l-1})}.\end{aligned} \tag{3.4.13}$$

Using the fact $\|\chi_i^{h,H}\|^2_{H^1_a(\Omega\setminus S_{l-1})} = \|\psi_i^{h,H}\|^2_{H^1_a(\Omega\setminus S_{l-1})}$, the exponential decay property and norm estimate of $\psi_i^{h,H}$, we finally obtain

$$\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \lesssim H^{d/2-1}\left(\beta(h,H)\right)^{l/2}\left(1+\frac{1}{\rho_{2,d}(\frac{H}{h})}\right).$$

On the other hand, we have

$$\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \le \|\psi_i^{h,H}\|_{H^1_a(\Omega)} + \|\psi_i^{h,H,l}\|_{H^1_a(\Omega)} \lesssim H^{d/2-1}\frac{1}{\rho_{2,d}(\frac{H}{h})},$$

due to the norm estimate established before. Thus, finally we obtain

$$\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \lesssim H^{d/2-1}\cdot\min\left\{\left(\beta(h,H)\right)^{l/2}\left(1+\frac{1}{\rho_{2,d}(\frac{H}{h})}\right), \frac{1}{\rho_{2,d}(\frac{H}{h})}\right\}.$$

Note that $1 \le 1+\frac{1}{\rho_{2,d}(\frac{H}{h})} \le 1+\frac{1}{\rho_{2,d}(1)}$, we could further simplify the the upper bound by

$$\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \lesssim H^{d/2-1}\cdot\min\left\{\left(\beta(h,H)\right)^{l/2}, \frac{1}{\rho_{2,d}(\frac{H}{h})}\right\}.$$

3.4.1.5 Overall Localization Error

Let $w = \mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u$, then

$$\|w\|^2_{H^1_a(\Omega)} = \sum_{i\in I}[u,\phi_i^{h,H}]\left\langle w, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle_a . \tag{3.4.14}$$

For each i, to deal with the term $\left\langle w, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle_a$, we introduce a cut-off function η that equals 0 in $\mathrm{N}^l(\omega_i^H)$ and equals 1 in $\Omega\backslash\mathrm{N}^{l+1}(\omega_i^H)$; moreover, $\eta \geq 0$ and $\|\nabla\eta\|_\infty \lesssim 1/H$. We define

$$v = \sum_{\omega_j^H \subset \Omega\backslash\mathrm{N}^l(\omega_i^H)} [\eta w, \phi_j^{h,H}]\psi_j^{h,H,0} \in H^1_0(\Omega\backslash\mathrm{N}^l(\omega_i^H)) .$$

Then $\eta w - v \in V^{h,H} \bigcap H^1_0(\Omega\backslash\mathrm{N}^l(\omega_i^H))$. Thus, we have $\left\langle \eta w - v, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle = 0$ because $\eta w - v$ has a different support with that of $\psi_i^{h,H,l}$, and $\left\langle \psi_i^{h,H}, v\right\rangle_a = 0$ for any $v \in V^{h,H}$; see the first paragraph in Subsection 3.4.1.4, Therefore, we get

$$\begin{aligned}
&\left\langle w, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle_a \\
&= \left\langle w - \eta w + v, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle_a \\
&\leq \left(\|(1-\eta)w\|_{H^1_a(\mathrm{N}^l(\omega_i^H))} + \|v\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H)\backslash\mathrm{N}^l(\omega_i^H))}\right) \|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)},
\end{aligned} \tag{3.4.15}$$

where we have used the fact that v is supported in $\mathrm{N}^{l+1}(\omega_i^H)\backslash\mathrm{N}^l(\omega_i^H)$. Then, by construction of v, we have $\|v\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H)\backslash\mathrm{N}^l(\omega_i^H))} \lesssim \|\eta w\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H)\backslash\mathrm{N}^l(\omega_i^H))}$; the proof of this property is similar to that of Lemma 3.4.1. Now, by using the fact $[w, \phi_j^{h,H}] = 0$ and the subsampled Poincare inequality, we obtain

$$\|(1-\eta)w\|_{H^1_a(\mathrm{N}^l(\omega_i^H))} + \|\eta w\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H)\backslash\mathrm{N}^l(\omega_i^H))} \lesssim \rho_{2,d}(\frac{H}{h})\|w\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H))} .$$

Therefore, $\left\langle w, \psi_i^{h,H} - \psi_i^{h,H,l}\right\rangle_a \lesssim \rho_{2,d}(\frac{H}{h})\|w\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H))}\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)}$. Then combining this estimate with (3.4.14), we arrive at

$$\begin{aligned}
\|w\|^2_{H^1_a(\Omega)} &\lesssim \rho_{2,d}(\frac{H}{h}) \sum_{i\in I}[u,\phi_i^{h,H}]\|w\|_{H^1_a(\mathrm{N}^{l+1}(\omega_i^H))}\|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \\
&\lesssim \rho_{2,d}(\frac{H}{h})\|u\|_{L^\infty(\Omega)} l^{d/2}\|w\|_{H^1_a(\Omega)} \left(\sum_{i\in I}\|\psi_i^{h,H} - \psi_i^{h,H,l}\|^2_{H^1_a(\Omega)}\right)^{1/2},
\end{aligned} \tag{3.4.16}$$

where the last step is by the Cauchy-Schwarz inequality. Combining the above estimate with the result in the last subsection (notice that the cardinality of I is $1/H^d$), we get

$$\|w\|_{H^1_a(\Omega)} \lesssim \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \cdot \frac{l^{d/2}}{H}\|u\|_{L^\infty(\Omega)} . \tag{3.4.17}$$

On the other hand, we can also bound

$$\begin{aligned}\|w\|_{H^1_a(\Omega)} &\leq \sum_{i\in I} |[u,\phi_i^{h,H}]| \cdot \|\psi_i^{h,H} - \psi_i^{h,H,l}\|_{H^1_a(\Omega)} \\ &\lesssim \|u\|_{L^\infty(\Omega)} H^{-d} \cdot H^{d/2-1} \cdot \min\left\{(\beta(h,H))^{l/2}, \frac{1}{\rho_{2,d}(\frac{H}{h})}\right\} \\ &\lesssim \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \cdot \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})} \|u\|_{L^\infty(\Omega)}\,.\end{aligned} \tag{3.4.18}$$

Therefore, we can write

$$\|w\|_{H^1_a(\Omega)} \lesssim \min\left\{(\beta(h,H))^{l/2}\,\rho_{2,d}(\frac{H}{h}), 1\right\} \cdot \min\left\{\frac{l^{d/2}}{H}, \frac{1}{H^{d/2+1}\rho_{2,d}(\frac{H}{h})}\right\} \|u\|_{L^\infty(\Omega)}\,. \tag{3.4.19}$$

3.4.1.6 Overall Recovery Error

When $d \leq 3$, we have $\|u\|_{L^\infty(\Omega)} \lesssim \|\mathcal{L}u\|_{L^2(\Omega)}$; for details see Theorems 8.22 and 8.29 in [99]. Combining the estimates in (3.2.7) and (3.2.12) leads to the estimate of the energy recovery error. For the L^2 recovery error, similar to (3.2.7), we have

$$e_0^{h,H,l}(a,u) \lesssim (H\rho_{2,d}(\frac{H}{h}))^2 \|\mathcal{L}u\|_{L^2(\Omega)} + \|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{L^2(\Omega)}\,. \tag{3.4.20}$$

The second term $\|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{L^2(\Omega)}$ is the L^2 localization error. We can simply bound it by

$$\|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{L^2(\Omega)} \leq \|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{H^1_a(\Omega)}\,. \tag{3.4.21}$$

On the other hand, notice that $[\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u, \phi_i^{h,H}] = 0$ for any $i \in I$, we can use the subsampled Poincaré inequality so that

$$\begin{aligned}\|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|^2_{L^2(\Omega)} &= \sum_{i\in I} \int_{\omega_i^H} |\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u|^2 \\ &\lesssim (H\rho_{2,d}(\frac{H}{h}))^2 \int_{\omega_i^H} a|\nabla(\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u)|^2 \\ &= (H\rho_{2,d}(\frac{H}{h}))^2 \|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|^2_{H^1_a(\Omega)}\,.\end{aligned} \tag{3.4.22}$$

Therefore, we obtain

$$\|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{L^2(\Omega)} \leq \min\left\{1, H\rho_{2,d}(\frac{H}{h})\right\} \|\mathsf{P}^{h,H}u - \mathsf{P}^{h,H,l}u\|_{H^1_a(\Omega)}\,. \tag{3.4.23}$$

Using the estimate of the energy error, we arrive at the final estimate.

3.4.1.7 Overall Galerkin Error

The estimate for the energy Galerkin error is straightforward due to the Galerkin orthogonality. The L^2 error is estimated using the standard Aubin-Nitsche trick in finite element theory, which leads to square of the energy error. This completes the proof.

3.4.2 Proof of Theorem 3.3.2

We start with the first case, i.e., $\|u\|_{H^1_a(\Omega)} < \infty$. By definition,

$$e_0^{h,H,\infty}(a,u) = \|u - \mathsf{P}^{h,H}u\|_{L^2(\Omega)}\,.$$

We have the relation $[u - \mathsf{P}^{h,H}u, \phi_j^{h,H}] = 0$ for any $j \in I$. Thus, using the weighted Poincaré inequality in [44] (Theorem 4.3 and Example 1), we can estimate the error as follows:

$$\begin{aligned}\|u - \mathsf{P}^{h,H}u\|^2_{L^2(\Omega)} &= \sum_{i\in I} \|u - \mathsf{P}^{h,H}u\|^2_{L^2(\omega_i^H)} \\ &\lesssim C(\gamma)^2 H^2 \sum_{i\in I} \|u - \mathsf{P}^{h,H}u\|^2_{H^1_a(\omega_i^H)} \\ &\lesssim C(\gamma)^2 H^2 \|u\|^2_{H^1_a(\Omega)},\end{aligned} \tag{3.4.24}$$

where in the last step, we have used the fact that $\|u - \mathsf{P}^{h,H}u\|_{H^1_a(\Omega)} \le \|u\|_{H^1_a(\Omega)}$ due to the energy orthogonality. The first case is proved.

For the second case, by energy orthogonality of the recovery, we get

$$e_1^{h,H,\infty}(a,u) \le \|u - v\|_{H^1_a(\Omega)}, \tag{3.4.25}$$

for any $v \in \text{span}\,\{\psi_i^{h,H}\}_{i\in I}$. We can write $v = \mathcal{L}^{-1}(\sum_{i\in I} c_i\phi_i^{h,H})$ for some c_i. Then, it holds that

$$\begin{aligned}\|u - v\|^2_{H^1_a(\Omega)} &= [u - v, \mathcal{L}(u - v)] \\ &= [u - v, f - \sum_{i\in I} c_i\phi_i^{h,H}] \\ &= \sum_{i\in I} \int_{\omega_i^H} (u - v)(f - c_i\phi_i^{h,H})\,.\end{aligned} \tag{3.4.26}$$

We choose $c_i = \int_{\omega_i^H} f$, so that

$$\begin{aligned}\sum_{i\in I} \int_{\omega_i^H} (u - v)(f - c_i\phi_i^{h,H}) &= \sum_{i\in I} \int_{\omega_i^H} \left(u - v - \int_{\omega_i^H} (u - v)\phi_i^{h,H}\right) f \\ &\lesssim C(\gamma) \sum_{i\in I} H\|u - v\|_{H^1_a(\omega_i^H)} \|f\|_{L^2(\omega_i^H)} \\ &\le C(\gamma) H \|u - v\|_{H^1_a(\Omega)} \|f\|_{L^2(\Omega)},\end{aligned} \tag{3.4.27}$$

where in the second inequality, we use the Cauchy-Schwarz inequality and the weighted Poincaré inequality (Theorem 4.3 and Example 1 in [44]). Thus, finally we get $\|u - v\|_{H^1_a(\Omega)} \lesssim C(\gamma)H\|f\|_{L^2(\Omega)}$, which implies the desired energy error estimate. The L^2 error estimate is obtained by using the standard Aubin-Nitsche trick in the finite element theory.

3.5 Conclusions

We summarize, discuss, and conclude this chapter in this section.

3.5.1 Summary

We performed a detailed study of a specific approach that connects the problem of numerical upscaling and function approximation, in the context that the target function is a solution to some multiscale elliptic PDEs with rough coefficients. Our main focus is on a subsampled lengthscale that appears in the coarse data of both problems. We investigated, both numerically and theoretically, the effect of h on the recovery errors (for function approximation) and Galerkin errors (for numerical upscaling), given no computational constraints (ideal solution) or limited computational budgets (localized solution with a finite l), and given different regularity assumptions on the target function ($a(x) \in L^\infty(\Omega)$ or a singular $a(x)$). Our results imply that

- There is a trade-off between approximation errors (of ideal solutions) and localization errors (due to finite l) regarding the subsampled lengthscale h, in addition to the oversampling parameter l.
- Due to the finite l caused by our limited computational budget, the Galerkin solution and recovery solution are different in general. The former behaves better in the energy accuracy, while the latter stands out in the L^2 accuracy.
- When the target function is "nearly flat" around the data locations, the subsampled data with a very small h can still contain much coarse scale information. Thus, we would recommend to take our measurements there as a first choice.

The more quantitative descriptions of these main results are established by our numerical experiments and analytic studies based on tools such as the finite element theory, the subsampled Poincaré inequality, and weighted inequalities.

3.5.2 Discussions

There could be multiple future directions:

- A better understanding of the trade-off regarding h and l: how to choose optimal l and h adaptively with respect to u or f. Our current results do not address this question fully.

- Other localization strategies: our localization in Subsection 3.1.6 follows from that in [181, 203], and there are other possibilities, for example, the one in [121] or [148], which leads to error estimates that does not blow up as $H \to 0$. It is of interest to understand how the subsampled lengthscale influences the accuracy in that context.

- Other measurement functions: as we mentioned earlier in Subsection 3.1.5, the choice of $\phi_i^{h,H}$ to be indicator functions in subsampled cubes is only for simplicity of analysis. Thus, results in this paper could be generalized to other types of subsampled measurement functions, for example, subsampled finite element tent functions.

- Generalization to high order models: the approach in Subsection 3.1.3 applies to a general operator $\mathcal{L}$ that can be high order elliptic operators. This also connects to our discussion in Subsection 3.3 regarding a high order model to avoid the degeneracy issues. It is of interest to study the effect of h, l and also the order of the operator $\mathcal{L}$ simultaneously on the recovery and Galerkin errors.

- Coupling of two problems: we have considered a common approach that connects two class of problems. A natural question is about a hybrid model: suppose we have the domain Ω split into two smaller domains Ω_1 and Ω_2. In Ω_1, we have a multiscale PDE $\mathcal{L}u = f$ with known f, and in Ω_2 we have some subsampled data $[u, \phi_i], i \in I$. How shall we take the advantages of the PDE model in Ω_1 and the measured data in Ω_2 to recover an accurate u? This can be a very fundamental problem in combining physics and data science.

3.5.3 Conclusions

Overall, we have explored the connection between numerical upscaling for multiscale PDEs and scattered data approximation for heterogeneous functions, focusing on the roles of a subsampled lengthscale h and the localization parameter l. We

believe it sheds light on the interplay of the lengthscale of coarse data, the computational costs, the regularity of the target function, and the accuracy of approximations and numerical simulations.

Chapter 4

GAUSSIAN PROCESSES FOR SOLVING AND LEARNING PDES AND INVERSE PROBLEMS

In this chapter, we discuss how to use Gaussian processes to solve and learn PDEs and inverse problems. The exposition is based on our work [43] published in *Journal of Computational Physics*, 447:110668, 2021.

4.1 Introduction

Two hundred years ago, modeling a physical problem and solving the underlying differential equations would have required the expertise of Cauchy or Laplace, and it would have been done by hand through a laborious process of discovery. Sixty years ago, the resolution of the underlying differential equation would have been addressed using computers, but modeling and design of the solver would have still been done by hand. Nowadays, there is increasing interest in automating these steps by casting them as machine learning problems. The resulting methods can be divided into two main categories: (1) methods based on variants of artificial neural networks (ANNs) [101], and (2) methods based on kernels and Gaussian Processes (GPs) [290, 246]. In the context of (1) there has been recent activity toward solving nonlinear PDEs, whilst the systematic development of methods of type (2) for nonlinear PDEs has remained largely open. However, methods of type (2) hold potential for considerable advantages over methods of type (1), both in terms of theoretical analysis and numerical implementation. In this work, our goal is to develop a simple kernel/GP framework for solving nonlinear PDEs and related inverse problems (IPs); in particular the methodology we introduce has the following properties:

- the proposed collocation setting for solving nonlinear PDEs and IPs is a direct generalization of optimal recovery kernel methods for linear PDEs [201, 203, 204], and a natural generalization of radial basis function collocation methods [301, 288], and of meshless kernel methods [239];
- theoretically, the proposed method is provably convergent and amenable to rigorous numerical analysis, suggesting new research directions to generalize

the analysis of linear regression methods [288] to the proposed collocation setting for solving nonlinear PDEs;

- computationally, it inherits the complexity of state-of-the-art solvers for dense kernel matrices, suggesting new research to generalize the work of [240], which developed optimal approximate methods for linear regression, to the proposed collocation setting for solving nonlinear PDEs and IPs;
- for IPs the methodology is closely aligned with methodologies prevalent in the PDE-constrained optimization literature [125] and suggests the need for new computational and theoretical analyses generalizing the standard optimization and Bayesian approaches found in [62, 76, 136].

Since ANN methods can also be interpreted as kernel methods with kernels learned from data [134, 202, 292], our framework could also be used for theoretical analysis of such methods.

In Subsection 4.1.1 we summarize the theoretical foundations and numerical implementation of our method in the context of a simple nonlinear elliptic PDE. In Subsection 4.1.2 we give a literature review, placing the proposed methodology in the context of other research at the intersection of machine learning and PDEs. The outline of this chapter is given in Subsection 4.1.3.

4.1.1 Summary of the Proposed Method

For demonstration purposes, we summarize the key ideas of our method for solving a nonlinear second-order elliptic PDE. This PDE will also serve as a running example in Section 4.3 where we present an abstract framework for general nonlinear PDEs.

Let $d \geq 1$ and Ω be a bounded open domain in $\mathbb{R}^d$ with a Lipschitz boundary $\partial\Omega$. Consider the following nonlinear elliptic equation for $u^\star$:

$$\begin{cases} -\Delta u^\star(\mathbf{x}) + \tau\big(u^\star(\mathbf{x})\big) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ u^\star(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega, \end{cases} \tag{4.1.1}$$

where $f : \Omega \to \mathbb{R}, g : \partial\Omega \to \mathbb{R}$ and $\tau : \mathbb{R} \to \mathbb{R}$ are given continuous functions. We assume that f, g, τ are chosen appropriately so that the PDE has a unique classical solution (for abstract theory of nonlinear elliptic PDEs see for example [220, 254]). In Subsection 4.1.1.4 we will present a concrete numerical experiment where $\tau(u) = u^3$ and $g(\mathbf{x}) = 0$.

4.1.1.1 Optimal Recovery

Our proposed method starts with an optimal recovery problem that can also be interpreted as maximum a posterior (MAP) estimation for a GP constrained by a PDE. More precisely, consider a nondegenerate, symmetric, and positive definite kernel function $K : \overline{\Omega}\times\overline{\Omega} \to \mathbb{R}$ where $\overline{\Omega} := \Omega \cup \partial\Omega$. Let $\mathcal{U}$ be the RKHS associated with K and denote the RKHS norm by $\|\cdot\|$. Let $1 \leq M_\Omega < M < \infty$ and fix M points in $\overline{\Omega}$ such that $\mathbf{x}_1, \ldots, \mathbf{x}_{M_\Omega} \in \Omega$ and $\mathbf{x}_{M_\Omega+1}, \ldots, \mathbf{x}_M \in \partial\Omega$. Then, our method approximates the solution $u^\star$ of (4.1.1) with a minimizer $u^\dagger$ of the following optimal recovery problem:

$$\begin{cases} \underset{u\in\mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad -\Delta u(\mathbf{x}_m) + \tau\big(u(\mathbf{x}_m)\big) = f(\mathbf{x}_m), & \text{for } m = 1, \ldots, M_\Omega, \\ \qquad u(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \ldots, M\,. \end{cases} \tag{4.1.2}$$

Here, we assume K is chosen appropriately so that $\mathcal{U} \subset C^2(\Omega) \cap C(\overline{\Omega})$, which ensures the pointwise PDE constraints in (4.1.2) are well-defined.

A minimizer $u^\dagger$ can be interpreted as a MAP estimator of a GP $\xi \sim \mathcal{N}(0, \mathcal{K})$[1](where $\mathcal{K}$ is the integral operator with kernel K) conditioned on satisfying the PDE at the collocation points $\mathbf{x}_m, 1 \leq m \leq M$. Such a view has been introduced for solving linear PDEs in [201, 203] and a closely related approach is studied in [55, Sec. 5.2]; the methodology introduced via (4.1.2) serves as a prototype for generalization to nonlinear PDEs. Here it is important to note that in the nonlinear case the conditioned GP is no longer Gaussian in general; thus the solution of (4.1.2) is a MAP estimator only and is not the conditional expectation, except in the case where $\tau(\cdot)$ is a linear function.

In the next subsections, we show equivalence of (4.1.2) and a finite dimensional constrained optimization problem (4.1.5). This provides existence of a minimizer to (4.1.2), as well as the basis for a numerical method to approximate the minimizer, based on solution of an unconstrained finite-dimensional optimization problem (4.1.6).

4.1.1.2 Finite-Dimensional Representation

The key to our numerical algorithm for solving (4.1.2) is a representer theorem that characterizes $u^\dagger$ via a finite-dimensional representation formula. To achieve this we

[1]This Gaussian prior notation is equivalent to the GP notation $\mathcal{GP}(0, K)$, where K is the covariance function. See further discussions in Subsection 4.3.4.1.

first reformulate (4.1.2) as a two level optimization problem:

$$\begin{cases} \underset{\mathbf{z}^{(1)}\in\mathbb{R}^M,\mathbf{z}^{(2)}\in\mathbb{R}^{M_\Omega}}{\text{minimize}} & \begin{cases} \underset{u\in\mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad u(\mathbf{x}_m) = z_m^{(1)} \text{ and } -\Delta u(\mathbf{x}_m) = z_m^{(2)}, \text{ for } m = 1,\dots,M, \end{cases} \\ \text{s.t.} \quad z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), & \text{for } m = 1,\dots,M_\Omega\,, \\ \qquad z_m^{(1)} = g(\mathbf{x}_m), & \text{for } m = M_\Omega+1,\dots,M\,. \end{cases} \tag{4.1.3}$$

Denote $\phi_m^{(1)} = \delta_{\mathbf{x}_m}$ and $\phi_m^{(2)} = \delta_{\mathbf{x}_m} \circ (-\Delta)$, where $\delta_{\mathbf{x}}$ is the Dirac delta function centered at $\mathbf{x}$. We further use the shorthand notation $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$ for the M and M_Ω-dimensional vectors with entries $\phi_m^{(1)}$ and $\phi_m^{(2)}$ respectively, and $\boldsymbol{\phi}$ for the $(M + M_\Omega)$-dimensional vector obtained by concatenating $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$. Similarly, we write $\mathbf{z}$ for the $(M + M_\Omega)$-dimensional vector concatenating $\mathbf{z}^{(1)}$ and $\mathbf{z}^{(2)}$.

For a fixed $\mathbf{z}$, we can solve the first level optimization problem explicitly due to the representer theorem[2] (see [204, Sec. 17.8]), which leads to the conclusion that

$$u(\mathbf{x}) = K(\mathbf{x},\boldsymbol{\phi})K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z}\,; \tag{4.1.4}$$

here $K(\cdot,\boldsymbol{\phi})$ denotes the $(M+M_\Omega)$-dimensional vector field with entries $\int K(\cdot,\mathbf{x}')\phi_m(\mathbf{x}')\,\mathrm{d}\mathbf{x}'$ and $K(\boldsymbol{\phi},\boldsymbol{\phi})$ is the $(M+M_\Omega)\times(M+M_\Omega)$-matrix with entries $\int K(\mathbf{x},\mathbf{x}')\phi_m(\mathbf{x})\phi_j(\mathbf{x}')\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{x}'$ with the ϕ_m denoting the entries of $\boldsymbol{\phi}$. For this solution, $\|u\|^2 = \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z}$, so we can equivalently formulate (4.1.3) as a finite-dimensional optimization problem:

$$\begin{cases} \underset{\mathbf{z}\in\mathbb{R}^{M+M_\Omega}}{\text{minimize}} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} \quad z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), & \text{for } m = 1,\dots,M_\Omega\,, \\ \qquad z_m^{(1)} = g(\mathbf{x}_m), & \text{for } m = M_\Omega+1,\dots,M\,. \end{cases} \tag{4.1.5}$$

Moreover, using the equation $z_m^{(2)} = f(\mathbf{x}_m) - \tau(z_m^{(1)})$ and the boundary data, we can further eliminate $\mathbf{z}^{(2)}$ and rewrite it as an unconstrained problem:

$$\underset{\mathbf{z}_\Omega^{(1)}\in\mathbb{R}^{M_\Omega}}{\text{minimize}} \ \left(\mathbf{z}_\Omega^{(1)}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)})\right) K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1} \begin{pmatrix} \mathbf{z}_\Omega^{(1)} \\ g(\mathbf{x}_{\partial\Omega}) \\ f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)}) \end{pmatrix}, \tag{4.1.6}$$

where we used $\mathbf{x}_\Omega$ and $\mathbf{x}_{\partial\Omega}$ to denote the interior and boundary points respectively, $\mathbf{z}_\Omega^{(1)}$ denotes the M_Ω-dimensional vector of the z_i for $i = 1,\dots,M_\Omega$ associated to the

[2]This is not the standard RKHS/GP representer theorem [290, Sec. 2.2] in the sense that measurements include the pointwise observation of higher order derivatives of the underlying GP. See [143] and [280, p. xiii] for related spline representation formulas with derivative information.

interior points $\mathbf{x}_\Omega$ while $f(\mathbf{x}_\Omega), g(\mathbf{x}_{\partial\Omega})$ and $\tau(\mathbf{z}_\Omega^{(1)})$ are vectors obtained by applying the corresponding functions to entries of their input vectors. For brevity we have suppressed the transpose signs in the row vector multiplying the matrix from the left in (4.1.6).

The foregoing considerations lead to the following existence result which underpins our numerical method for (4.1.1); furthermore (4.1.6) provides the basis for our numerical implementations. We summarize these facts:

Theorem 4.1.1. *The variational problem* (4.1.2) *has a minimizer of the form* $u^\dagger(\mathbf{x}) = K(\mathbf{x}, \boldsymbol{\phi})K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}^\dagger$*, where* $\mathbf{z}^\dagger$ *is a minimizer of* (4.1.5)*. Furthermore* $u^\dagger(\mathbf{x})$ *may be found by solving the unconstrained minimization problem* (4.1.6) *for* $\mathbf{z}_\Omega^{(1)}$*.*

Proof. Problems (4.1.2), (4.1.3) and (4.1.5) are equivalent. It is therefore sufficient to show that (4.1.5) has a minimizer. Write $\mathbf{z}^\star$ for the vector with entries $z_m^{\star(1)} = u^\star(\mathbf{x}_m)$ and $z_m^{\star(2)} = -\Delta u^\star(\mathbf{x}_m)$. Since $u^\star$ solves the PDE (4.1.1), $\mathbf{z}^\star$ satisfies the constraints on $\mathbf{z}$ in (4.1.5). It follows that the minimization in (4.1.5) can be restricted to the set C of $\mathbf{z}$ that satisfies $\mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z} \leq (\mathbf{z}^\star)^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}^\star$ and the nonlinear constraints. The set C is compact and non-empty: compact because τ is continuous and so the constraint set is closed as it is the pre-image of a closed set, and the intersection of a closed set with a compact set is compact; and nonempty because it contains $\mathbf{z}^*$. Thus the objective function $\mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}$ achieves its minimum in the set C. Once $\mathbf{z}_\Omega^{(1)}$ is found we can extend to the boundary points to obtain $\mathbf{z}^{(1)}$, and use the nonlinear relation between $\mathbf{z}^{(1)}$ and $\mathbf{z}^{(2)}$ to reconstruct $\mathbf{z}^{(2)}$. This gives $\mathbf{z}^\dagger$. □

4.1.1.3 Convergence Theory

The consistency of our method is guaranteed by the convergence of $u^\dagger$ towards $u^\star$ as M, the total number of collocation points, goes to infinity. We first present this result in the case of the nonlinear PDE (4.1.1) and defer a more general version to Subsection 4.3.2. We also give a simple proof of convergence here as it is instructive in understanding why the method works and how the more general result can be established. Henceforth we use $|\cdot|$ to denote the standard Euclidean norm and write $\mathcal{U} \subseteq \mathcal{H}$ to denote $\mathcal{U}$ being continuously embedded in Banach space $\mathcal{H}$.

Theorem 4.1.2. *Suppose that K is chosen so that* $\mathcal{U} \subseteq H^s(\Omega)$ *for some* $s > 2 + d/2$ *and that* (4.1.1) *has a unique classical solution* $u^\star \in \mathcal{U}$*. Write* $u_M^\dagger$ *for a minimizer*

of (4.1.2) *with M distinct collocation points* $\mathbf{x}_1, \dots, \mathbf{x}_M$*. Assume that, as* $M \to \infty$*,*

$$\sup_{\mathbf{x}\in\Omega} \min_{1\le m\le M_\Omega} |\mathbf{x}-\mathbf{x}_m| \to 0 \quad \textit{and} \quad \sup_{\mathbf{x}\in\partial\Omega} \min_{M_{\Omega+1}\le m\le M} |\mathbf{x}-\mathbf{x}_m| \to 0\,.$$

Then, as $M \to \infty$*,* $u_M^\dagger$ *converges towards* $u^\star$ *pointwise in* Ω *and in* $H^t(\Omega)$ *for any* $t < s$*.*

Proof. The proof relies on a simple compactness argument together with the Sobolev embedding theorem [2, 33]. First, as $u^\star$ satisfies the constraint in (4.1.2) and $u_M^\dagger$ is the minimizer, it follows that $\|u_M^\dagger\| \le \|u^\star\|$ for all $M \ge 1$. This implies $\|u_M^\dagger\|_{H^s(\Omega)} \le C\|u^\star\|$ because $\mathcal{U}$ is continuously embedded into $H^s(\Omega)$. Let $t \in (2 + d/2, s)$ so that $H^t(\Omega)$ embeds continuously into $C^2(\Omega) \cap C(\overline{\Omega})$ [2, Thm. 4.12]. Since $H^s(\Omega)$ is compactly embedded into $H^t(\Omega)$, we deduce that there exists a subsequence $\{M_p, p \ge 1\} \subset \mathbb{N}$ and a limit $u_\infty^\dagger \in H^t(\Omega)$ such that $u_{M_p}^\dagger$ converges towards $u_\infty^\dagger$ in the $H^t(\Omega)$ norm, as $p \to \infty$. This also implies convergence in $C^2(\Omega)$ due to the continuous embedding of $H^t(\Omega)$ into $C^2(\Omega) \cap C(\overline{\Omega})$.

We now show that $u_\infty^\dagger$ satisfies the desired PDE in (4.1.1). Denote by $v = -\Delta u_\infty^\dagger + \tau\left(u_\infty^\dagger\right) - f \in C(\Omega)$ and $v_p = -\Delta u_{M_p}^\dagger + \tau\left(u_{M_p}^\dagger\right) - f \in C(\Omega)$. For any $\mathbf{x} \in \Omega$ and $p \ge 1$, use of the triangle inequality shows that

$$\begin{aligned} |v(\mathbf{x})| &\le \min_{1\le m\le M_{p,\Omega}} \left(|v(\mathbf{x}) - v(\mathbf{x}_m)| + \left|v(\mathbf{x}_m) - v_p(\mathbf{x}_m)\right|\right) \\ &\le \min_{1\le m\le M_{p,\Omega}} |v(\mathbf{x}) - v(\mathbf{x}_m)| + \|v - v_p\|_{C(\Omega)}\,, \end{aligned} \tag{4.1.7}$$

where we have used the fact that $v_p(\mathbf{x}_m) = 0$, and $M_{p,\Omega}$ is the number of interior points associated with the total M_p collocation points. Since v is continuous in a compact domain, it is also uniformly continuous. Thus, it holds that $\lim_{p\to\infty} \min_{1\le m\le M_{p,\Omega}} |v(\mathbf{x}) - v(\mathbf{x}_m)| = 0$ since the fill-distance converges to zero by assumption. Moreover, we have that v_p converges to v in the $C(\Omega)$ norm due to the $C^2(\Omega)$ convergence from $u_{M_p}^\dagger$ to $u_\infty^\dagger$. Combining these facts with (4.1.7), and taking $p \to \infty$, we obtain $v(\mathbf{x}) = 0$, and thus $-\Delta u_\infty^\dagger(\mathbf{x}) + \tau\left(u_\infty^\dagger(\mathbf{x})\right) = f(\mathbf{x})$, for $\mathbf{x} \in \Omega$. Following a similar argument, we get $u_\infty^\dagger(\mathbf{x}) = g(\mathbf{x})$ for $\mathbf{x} \in \partial\Omega$. Thus, $u_\infty^\dagger$ is a classical solution to (4.1.1). By assumption, the solution is unique, so we must have $u_\infty^\dagger = u^\star$. As the limit $u_\infty^\dagger$ is independent of the particular subsequence, the whole sequence $u_M^\dagger$ must converge to $u^\star$ in $H^t(\Omega)$. Since $t \in (2 + d/2, s)$, we also get pointwise convergence and convergence in $H^t(\Omega)$ for any $t < s$. □

We note that this convergence theorem requires K to be adapted to the solution space of the PDE so that $u^\star$ belongs to $\mathcal{U}$. In our numerical experiments, we will use a

Gaussian kernel. However, if f or the boundary $\partial\Omega$ are not sufficiently regular, then the embedding conditions $u^\star \in \mathcal{U} \subseteq H^s(\Omega)$ may be satisfied by using kernel as the Green's function of some power of the Laplacian, instead of a Gaussian kernel; the latter induces smoothness on $\mathcal{U}$ which may be incompatible with the regularity of $u^\star$ for irregular f and $\partial\Omega$.

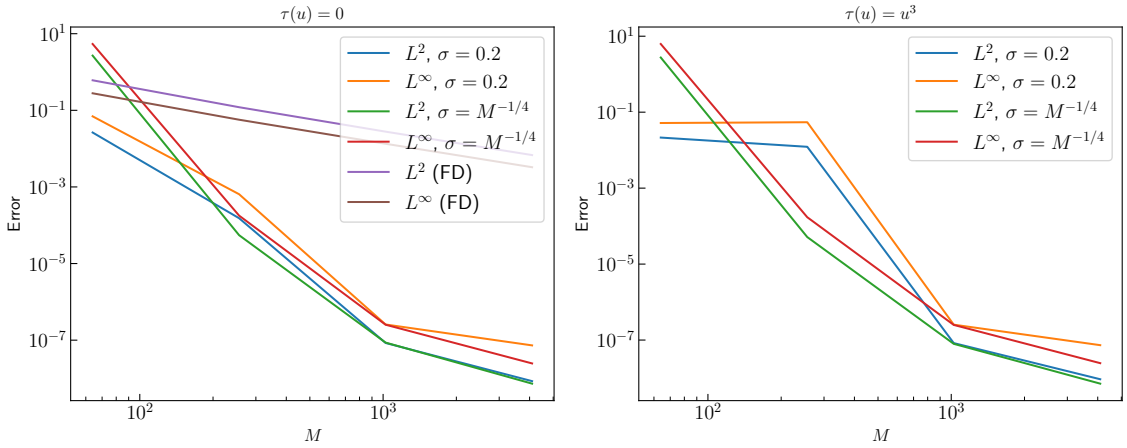


Figure 4.1: L^2 and L^∞ error plots for numerical approximations of $u^\star$, the solution to (4.1.1), as a function of the number of collocation points M. Left: $\tau(u) = 0$; both the kernel collocation method using Gaussian kernel with $\sigma = 0.2$ and $M^{-1/4}$ and the finite difference (FD) method were implemented. Right: $\tau(u) = u^3$; the kernel collocation method using Gaussian kernel with $\sigma = 0.2$ and $M^{-1/4}$ were used. In both cases, an adaptive nugget term with global parameter $\eta = 10^{-13}$ was used to regularize the kernel matrix Θ (see Appendix A.1.1 for details).

4.1.1.4 Numerical Framework

The representation of $u^\dagger$ via the optimization problem (4.1.6) is the cornerstone of our numerical framework for solving nonlinear PDEs. Indeed, efficient solution of (4.1.6), and in turn construction and inversion of the matrix $K(\boldsymbol{\phi}, \boldsymbol{\phi})$, are the most costly steps of our numerical implementation. We summarize several main ingredients of our algorithm below:

- We propose an efficient variant of the Gauss–Newton algorithm in Section 4.3.4.2. Although, in general, (4.1.6) is a nonconvex problem, our algorithm typically converges in between 2 and 10 steps in all the experiments we have conducted.

- In practice we perturb $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ to improve its condition number, and hence the numerical stability of the algorithm, by adding a small diagonal matrix; this perturbation is adapted to the problem at hand, as outlined in Appendix A.1.1; the approach generalizes the idea of a "nugget" as widely used in GP regression.

- To evaluate the cost function in (4.1.6), we pre-compute the Cholesky factorization of the (perturbed) kernel matrix and store it for multiple uses. State-of-the-art linear solvers for dense kernel matrices can be used for this step.

As a demonstration, we present here a simple experiment to show the convergence of our method. We take $d = 2$, $\Omega = (0,1)^2$ and $\tau(u) = 0$ or u^3 together with homogeneous Dirichlet boundary conditions $g(\mathbf{x}) = 0$. The true solution is prescribed to be $u^\star(\mathbf{x}) = \sin(\pi x_1)\sin(\pi x_2) + 4\sin(4\pi x_1)\sin(4\pi x_2)$ and the corresponding right hand side $f(\mathbf{x})$ is computed using the equation.

We choose the Gaussian kernel

$$K(\mathbf{x}, \mathbf{y}; \sigma) = \exp\Big(-\frac{|\mathbf{x} - \mathbf{y}|^2}{2\sigma^2}\Big),$$

with lengthscale parameter σ; we will fix this parameter, but note that the framework is naturally adapted to learning of such hyperparameters. We set $M = 64, 256, 1024, 4096$, sampling the collocation points on a uniform grid of points within Ω. We apply our algorithm to solve the PDEs and compute the L^2 and L^∞ errors to $u^\star$. In the case $\tau(u) = 0$, which corresponds to a linear PDE, we also compare our algorithm with a second-order finite difference (FD) method. For the nonlinear case $\tau(u) = u^3$, we observe that the Gauss–Newton algorithm only needs 2 steps to converge. The errors versus M are depicted in Figure 4.1. The following observations can be made from this figure:

- In the linear case $\tau(u) = 0$, where the corresponding optimization problem (4.1.6) is convex, our algorithm outperforms the FD method in terms of accuracy and rate of convergence. This can be attributed to the fact that the true solution is very smooth, and the Gaussian kernel allows for efficient approximation.

- The choice of the kernel parameter σ has a profound impact on the accuracy and rate of convergence of the algorithm, especially when M is not very large.

This implies the importance of choosing a "good kernel" that is adapted to the solution space of the PDE, and highlights the importance of hyperparameter learning.

- In the nonlinear setting, our algorithm demonstrates similar convergence behavior to the linear setting. Once again, an appropriate choice of σ leads to significant gains in solution accuracy.

4.1.2 Relevant Literature

Machine learning and statistical inference approaches to numerical approximation have attracted a lot of attention in recent years thanks to their remarkable success in engineering applications. Such approaches can be broadly divided into two categories: (1) GP/Kernel-based methods, and (2) ANN-based methods.

GPs and kernel-based methods have long been used in function approximation, regression, and machine learning [290]. As surveyed in [205]:

> *"[kernel approaches can be traced back to] Poincaré's course in Probability Theory [217] and to the pioneering investigations of Sul'din [263], Palasti and Renyi [212], Sard [237], Kimeldorf and Wahba [141] (on the correspondence between Bayesian estimation and spline smoothing/interpolation) and Larkin [157] (on the correspondence between Gaussian process regression and numerical approximation). Although their study initially attracted little attention among numerical analysts [157], it was revived in Information Based Complexity (IBC) [270], Bayesian numerical analysis [69], and more recently in Probabilistic Numerics [119] and Bayesian numerical homogenization [201]."*

For recent reviews of the extensive applications of GP/kernel methods see [54, 205, 264] and [204, Chap. 20]. In particular, they have been introduced to motivate novel methods for solving ordinary differential equations (ODEs) [251, 39, 245] and underlie the collocation approach advocated for parameter identification in ODEs as described in [226]. For PDEs, the use of kernel methods can be traced back to meshless collocation methods with radial basis functions [301, 288, 239]. Furthermore, a recent systematic development towards solving PDEs was initiated in [201, 203, 207, 204] and has lead to the identification of fast solvers for elliptic PDEs and dense kernel matrices [241, 240] with state-of-the-art complexity versus accuracy guarantees. The effectiveness of these methods has been supported by

well-developed theory [204] residing at the interfaces between numerical approximation [280, 288], optimal recovery [188], information-based complexity [270], and GP regression [30], building on the perspective introduced in [69, 157, 188, 269, 280]. In particular there is a one to one correspondence [204, 241] between (1) the locality of basis functions (gamblets) obtained with kernel methods and the *screening effect* observed in kriging [259], (2) Schur complementation and conditioning Gaussian vectors, and (3) the approximate low-rank structure of inverse stiffness matrices obtained with kernel methods and variational properties of Gaussian conditioning. Furthermore, although the approach of modeling a deterministic function (the solution $u^\star$ of a PDE) as a sample from a Gaussian distribution may seem counterintuitive, it can be understood (in the linear setting [204]) as an optimal minimax strategy for recovering $u^\star$ from partial measurements. Indeed, as in Von Neumann's theory of games, optimal strategies are mixed (randomized) strategies and (using quadratic norms to define losses) GP regression (kriging) emerges as an optimal minimax approach to numerical approximation [188, 204].

On the other hand, ANN methods can be traced back to [70, 155, 156, 229, 272] and, although developed for ODEs several decades ago [53, 229], with some of the work generalized to PDEs [149], their systematic development for solving PDEs has been initiated only recently. This systematic development includes the Deep Ritz Method [287], Physics Informed Neural Networks [222] (PINNs), DGN [250], and [115] which employs a probabilistic formulation of the nonlinear PDE via the Feynman-Kac formula. Although the basic idea is to replace unknown functions with ANNs and minimize some loss with respect to the parameters of the ANN to identify the solution, there are by now many variants of ANN approaches, and we refer to [137] for a recent review of this increasingly popular topic at the interface between scientific computing and deep learning. While ANN approaches have been shown to be empirically successful on complex problems (e.g., machine learning physics [225]), they may fail on simple ones [284, 275] if the architecture of the ANN is not adapted to the underlying PDE [283]. Moreover, the theory of ANNs is still in its infancy; most ANN-based PDE solvers lack theoretical guarantees and convergence analyses are often limited to restricted linear instances [249, 284]. Broadly speaking, in comparison with kernel methods, ANN methods have both limited theoretical foundations and unfavorable complexity vs accuracy estimates. We also remark that ANN methods are suitable for the learning of the parametric dependence of solutions of differential equations [303, 29, 163, 164, 31]; however, GP and kernel methods may also be used in this context, and the random feature method provides

a natural approach to implementing such methods in high dimensions [196].

Regardless, the theory and computational framework of kernel methods may naturally be extended to ANN methods to investigate[3] such methods and possibly accelerate them by viewing them as ridge regression with data-dependent kernels and following [241, 240]. To this end, ANN methods can be interpreted as kernel methods with data-dependent kernels in two equivalent ways: (1) as solving PDEs in an RKHS space spanned by a feature map parameterized by the initial layers of the ANN that is learned from data, or, (2) as kernel-based methods with kernels that are parameterized by the inner layers of the network. For instance, [202] shows that Residual Neural Networks [118] (ResNets) are ridge regressors with warped kernels [233, 213]. Given the notorious difficulty of developing numerical methods for nonlinear PDEs [265], it is to some degree surprising that (as suggested by our framework) (A) this difficulty can universally be decomposed into three parts: (1) the compression/inversion of dense kernel matrices, (2) the selection of the kernel, and (3) the minimization of the reduced finite-dimensional optimization problem (4.3.10), and (B) a significant portion of the resulting analysis can be reduced to that of linear regression [204].

Beyond solving PDEs, ANN methods have also been used in data-driven discretizations, and discovery of PDEs that allow for the identification of the governing model [222, 171, 22]; this leads to applications in IPs. Our method, viewed as a GP conditioned on PDE constraints at collocation points, can be interpreted as solving an IP with Bayesian inference and a Gaussian prior [261]. Indeed, if we relax the PDE constraints as in Subsection 4.3.3.2 then a minimizer $u^\dagger$ coincides with a MAP estimator of a posterior measure obtained by viewing the PDE constraints as nonlinear pointwise measurements of a field u with a Gaussian prior $\mathcal{N}(0,\mathcal{K})$. Bayesian IPs with Gaussian priors have been studied extensively (see [62, 57, 261] and references therein). The standard approach for their solution is to discretize the problem using spectral projection or finite element methods, and compute posterior MAP estimators [61] or employ Markov chain Monte Carlo algorithms [58] to simulate posterior samples. Our abstract representation theorem outlined in Section 4.2.1 completely characterizes the MAP estimator of posterior measures with Gaussian priors in settings where the forward map is written as the composition of a nonlinear map with bounded linear functionals acting on the parameter. Indeed, this is precisely the approach that we employ in Section 4.4 to solve IPs with PDE constraints. However,

[3] Beyond the infinite width neural tangent kernel regime [134, 284].

the main difference between our methodology and existing methods in the literature is that we pose the IP as that of recovering the solution of the PDE $u^\dagger$ simultaneously with learning the unknown PDE parameter with independent Gaussian priors on both unknowns.

We now turn to motivation for the GP interpretation. The PDE solvers obtained here are deterministic and could be described from an entirely classical numerical approximation perspective. However we emphasize the GP interpretation for two reasons: (i) it is integral to the derivation of the methods, and (ii) it allows the numerical solution of the PDE to be integrated into a larger engineering pipeline and, in that context, the posterior distribution of the GP conditioned on the PDE at collocation points provides a measure of uncertainty quantification. Using the GP perspective as a pathway to the discovery of numerical methods, was the motivation for the work in [201, 203]. Indeed, as discussed in [201], while the discovery of numerical solvers for PDEs is usually based on a combination of insight and guesswork, this process can be facilitated to the point of being automated, using this GP perspective. For instance, for nonsmooth PDEs, basis functions with near optimal accuracy/localization tradeoff and operator valued wavelets can be identified by conditioning physics informed Gaussian priors on localized linear measurements (e.g., local averages or pointwise values). Physics informed Gaussian priors can, in turn, be identified by (a) filtering uninformed Gaussian priors through the inverse operator [201], or (b) turning the process of computing fast with partial information into repeated zero-sum games with physics informed losses (whose optimal mixed strategies are physics informed Gaussian priors) [203]. The paper [224] generalized (a) to time-dependent PDEs by filtering uninformed priors through linearized PDEs obtained via time stepping. The paper [55] emphasized the probabilistic interpretation of numerical errors obtained from this Bayesian perspective. In particular [55, Sec. 5.2] describes a method identical to the one considered here (and [201]) for linear problems; in the setting of semi-linear PDEs, it is suggested in [55] that latent variables could be employed to efficiently sample from posterior/conditional measures (see also [202] where latent variables were employed to reduce nonlinear optimal recovery problems via two-level optimization as in 4.1.3). Although the methodology proposed in our work agrees with that in [55] for linear problems, the methodology we propose appears to be more general, and differs in the case of nonlinear problems to which both approaches apply.

4.1.3 Outline

The remainder of this chapter is organized as follows. We give an overview of the abstract theory of GPs on Banach spaces in Section 4.2; we establish notation, and summarize basic results and ideas that are used throughout the remainder of the chapter. Section 4.3 is dedicated to the development of our numerical framework for solving nonlinear PDEs with kernel methods; we outline our assumptions on the PDE, present a general convergence theory, discuss our approach to implementation, and present numerical experiments. In Section 4.4 we extend our nonlinear PDE framework to IPs and discuss the implementation differences between the PDE and IP settings, followed by numerical experiments involving a benchmark IP in subsurface flow. Finally, we present additional discussions, results, and possible extensions of our method in Section 4.5. Appendix A.1 is devoted to the small diagonal regularization of kernel matrices ("nugget" term) and outlines general principles as well as specifics for the examples considered in this chapter.

4.2 Conditioning GPs on Nonlinear Observations

In this section, we outline the abstract theory of RKHSs/GPs and their connection to Banach spaces endowed with quadratic norms; this forms the framework for the proposed methodology to solve PDEs and IPs. We start by recalling some basic facts about GPs in Subsection 4.2.1. This is followed in Subsection 4.2.2 by general results pertaining to conditioning of GPs on linear and nonlinear observations, leading to a representer theorem that is the cornerstone of our numerical algorithms. Some of these results may be known to experts, but to the best of our knowledge, they are not presented in the existing literature with the coherent goal of solving nonlinear problems via the conditioning of GPs; hence this material may be of independent interest to the reader.

4.2.1 GPs and Banach Spaces Endowed with a Quadratic Norm

Consider a separable Banach space $\mathcal{U}$ and its dual $\mathcal{U}^t$ with their duality pairing denoted by $[\cdot, \cdot]$. We further assume that $\mathcal{U}$ is endowed with a quadratic norm $\|\cdot\|$, i.e., there exists a linear bijection $\mathcal{K} : \mathcal{U}^t \to \mathcal{U}$ that is symmetric ($[\mathcal{K}\phi, \varphi] = [\mathcal{K}\varphi, \phi]$), positive ($[\mathcal{K}\phi, \phi] > 0$ for $\phi \neq 0$), and such that

$$\|u\|^2 = [\mathcal{K}^{-1}u, u], \qquad \forall u \in \mathcal{U}\,.$$

The operator $\mathcal{K}$ endows $\mathcal{U}$ and $\mathcal{U}^t$ with the following inner products:

$$\langle u, v\rangle := [\mathcal{K}^{-1}u, v], \quad u, v \in \mathcal{U}\,,$$
$$\langle \phi, \varphi\rangle_t := [\phi, \mathcal{K}\varphi], \qquad \phi, \varphi \in \mathcal{U}^t\,.$$

Note that the $\langle\cdot,\cdot\rangle_t$ inner product defines a norm on $\mathcal{U}^t$ that coincides with the standard dual norm of $\mathcal{U}^t$, i.e.,

$$\|\phi\|_t = \sup_{u\in\mathcal{U}, u\neq 0} \frac{[\phi, u]}{\|u\|} = \sqrt{[\phi, \mathcal{K}\phi]}\,.$$

As in [204, Chap. 11], although $\mathcal{U}, \mathcal{U}^t$ are also Hilbert spaces under the quadratic norms $\|\cdot\|$ and $\|\cdot\|_t$, we will keep using the Banach space terminology to emphasize the fact that our dual pairings will not be based on the inner product through the Riesz representation theorem, but on a different realization of the dual space. A particular case of the setting considered here is $\mathcal{U} = H_0^s(\Omega)$ (writing $H_0^s(\Omega)$ for the closure of the set of smooth functions with compact support in Ω with respect to the Sobolev norm $\|\cdot\|_{H^s(\Omega)}$), with its dual $\mathcal{U}^t = H^{-s}(\Omega)$ defined by the pairing $[\phi, v] := \int_\Omega \phi u$ obtained from the Gelfand triple $H^s(\Omega) \subset L^2(\Omega) \subset H^{-s}(\Omega)$. Here $\mathcal{K}$ can be defined as solution map of an elliptic operator[4] mapping $H_0^s(\Omega)$ to $H^{-s}(\Omega)$.

We say that ξ is the *canonical GP* [204, Chap. 17.6] on $\mathcal{U}$ and write $\xi \sim \mathcal{N}(0, \mathcal{K})$, if and only if ξ is a linear isometry from $\mathcal{U}^t$ to a centered Gaussian space (a closed linear space of scalar valued centered Gaussian random variables). The word canonical indicates that the covariance operator of ξ coincides with the bijection $\mathcal{K}$ defining the norm on $\mathcal{U}$. Write $[\phi, \xi]$ for the image of $\phi \in \mathcal{U}^t$ under ξ and note that the following properties hold:

$$\mathbb{E}[\phi, \xi] = 0 \quad \text{and} \quad \mathbb{E}[\phi, \xi][\varphi, \xi] = [\phi, \mathcal{K}\varphi], \quad \forall \phi, \varphi \in \mathcal{U}^t\,.$$

The space $\mathcal{U}$ coincides with the Cameron–Martin space of the GP $\mathcal{N}(0, \mathcal{K})$. In the setting where $\mathcal{K}$ is defined through a covariance kernel K (such as in Subsection 4.1.1 and later in Subsection 4.3.4.2) then $\mathcal{U}$ coincides with the RKHS space of the kernel K [276, Sec. 2.3].

4.2.2 Conditioning GPs with Linear and Nonlinear Observations

Let $\phi_1, \ldots, \phi_N$ be N non-trivial elements of $\mathcal{U}^t$ and define

$$\boldsymbol{\phi} := (\phi_1, \ldots, \phi_N) \in (\mathcal{U}^t)^{\otimes N}\,. \tag{4.2.1}$$

[4] Given $s > 0$, we call an invertible operator $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$ elliptic, if it is positive and symmetric in the sense that $\int_\Omega u\mathcal{L}u \geq 0$ and $\int_\Omega u\mathcal{L}v = \int_\Omega v\mathcal{L}u \geq 0$.

Now consider the canonical GP $\xi \sim \mathcal{N}(0, \mathcal{K})$, then $[\boldsymbol{\phi}, \xi]$ is an $\mathbb{R}^N$-valued Gaussian vector and $[\boldsymbol{\phi}, \xi] \sim \mathcal{N}(0, \Theta)$ where

$$\Theta \in \mathbb{R}^{N \times N}, \qquad \Theta_{i,n} = [\phi_i, \mathcal{K}\phi_n], \quad 1 \leq i, n \leq N\,. \tag{4.2.2}$$

The following proposition characterizes the conditional distribution of GPs under these linear observations; to simplify the statement it is useful to write $\mathcal{K}(\boldsymbol{\phi}, \varphi)$ for the vector with entries $[\phi_i, \mathcal{K}\varphi]$. This type of vectorized notation is used in [204].

Proposition 4.2.1. *Consider a vector* $\mathbf{y} \in \mathbb{R}^N$ *and the canonical GP* $\xi \sim \mathcal{N}(0, \mathcal{K})$. *Then* ξ *conditioned on* $[\boldsymbol{\phi}, \xi] = \mathbf{y}$ *is also Gaussian. Moreover if* Θ *is invertible then* $\mathrm{Law}[\xi | [\boldsymbol{\phi}, \xi] = \mathbf{y}] = \mathcal{N}(u^\dagger, \mathcal{K}_{\boldsymbol{\phi}})$ *with conditional mean defined by* $u^\dagger = \mathbf{y}^T \Theta^{-1} \mathcal{K}\boldsymbol{\phi}$ *and conditional covariance operator defined by* $[\varphi, \mathcal{K}_{\boldsymbol{\phi}}\varphi] = [\varphi, \mathcal{K}\varphi] - \mathcal{K}(\varphi, \boldsymbol{\phi})\Theta^{-1}\mathcal{K}(\boldsymbol{\phi}, \varphi), \forall \varphi \in \mathcal{U}^t$.

Proposition 4.2.1 gives a finite representation of the conditional mean of the GP constituting a representer theorem [204, Cor. 17.12]. Let us define the elements

$$\chi_i := \sum_{n=1}^{N} \Theta^{-1}_{i,n} \mathcal{K}\phi_n\,, \tag{4.2.3}$$

referred to as *gamblets* in the parlance of [203] which can equivalently be characterized as the minimizers of the variational problem

$$\begin{cases} \underset{\chi \in \mathcal{U}}{\text{minimize}} & \|\chi\| \\ \text{s.t.} & [\phi_n, \chi] = \delta_{i,n}, \qquad n = 1, \ldots, N\,. \end{cases} \tag{4.2.4}$$

This fact further enables the variational characterization of the conditional mean $u^\dagger$ directly in terms of the gamblets χ_n.

Proposition 4.2.2. *Let* $u^\dagger = \mathbb{E}[\xi | [\boldsymbol{\phi}, \xi] = \mathbf{y}]$ *as in Proposition 4.2.1. Then* $u^\dagger = \sum_{n=1}^{N} y_n \chi_n$ *is the unique minimizer of*

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} & \|u\| \\ \text{s.t.} & [\phi_n, u] = y_n, \qquad n = 1, \ldots, N\,. \end{cases}$$

Proposition 4.2.2 is the cornerstone of our methodology for solution of nonlinear PDEs. It is also useful for the solution of IPs. For this purpose consider nonlinear

functions $G : \mathbb{R}^N \to \mathbb{R}^I$ and $F : \mathbb{R}^N \to \mathbb{R}^M$ and vectors $\mathbf{o} \in \mathbb{R}^I$ and $\mathbf{y} \in \mathbb{R}^M$ and consider the optimization problem:

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} & \|u\|^2 + \frac{1}{\gamma^2}|G([\boldsymbol{\phi}, u]) - \mathbf{o}|^2 \\ \text{s.t.} & F([\boldsymbol{\phi}, u]) = \mathbf{y}\,, \end{cases} \tag{4.2.5}$$

where $\gamma \in \mathbb{R}$ is a parameter. We will use this formulation of IPs in PDEs, with u concatenating the solution of the forward PDE problem and the unknown parameter; the nonlinear constraint on F enforces the forward PDE and G the observed noisy data. Then a representer theorem still holds for a minimizer of this problem stating that the solution has a finite expansion in terms of the gamblets χ_n:

Proposition 4.2.3. *Suppose* $(\mathbf{o}, \mathbf{y}) \in \mathbb{R}^I \times \mathbb{R}^M$ *are fixed and* Θ *is invertible*[5]*. Then* $u^\dagger \in \mathcal{U}$ *is a minimizer of* (4.2.5) *if and only if* $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ *and the vector* $\mathbf{z}^\dagger$ *is a minimizer of*

$$\begin{cases} \underset{\mathbf{z} \in \mathbb{R}^N}{\text{minimize}} & \mathbf{z}^T \Theta^{-1} \mathbf{z} + \frac{1}{\gamma^2}|G(\mathbf{z}) - \mathbf{o}|^2 \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}\,. \end{cases} \tag{4.2.6}$$

Proof. The proof is nearly identical to the derivation of (4.1.5) presented in Section 4.1.1.2. Simply observe that minimizing (4.2.5) is equivalent to minimizing

$$\underset{\mathbf{z} \in \mathbb{R}^N \,:\, F(\mathbf{z}) = \mathbf{y}}{\text{minimize}} \begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} & \|u\|^2 + \frac{1}{\gamma^2}|G(\mathbf{z}) - \mathbf{o}|^2 \\ \text{s.t.} & [\boldsymbol{\phi}, u] = \mathbf{z}\,. \end{cases} \tag{4.2.7}$$

Then solve the inner optimization problem for a fixed $\mathbf{z}$ and apply Proposition 4.2.1. □

We note that this model assumes independent and identically distributed (i.i.d.) observation noise for the vector $\mathbf{o}$ and can easily be extended to correlated observation noise by replacing the misfit term $\frac{1}{\gamma^2}|G(\mathbf{z}) - \mathbf{o}|^2$ in (4.2.5) with an appropriately weighted misfit term of the form $|\Sigma^{-1/2}(G(\mathbf{z}) - \mathbf{o})|^2$ where Σ denotes the covariance matrix of the observation noise.

Remark 4.2.4. *It is intuitive that a minimizer of the optimization problem we introduce and solve in this work corresponds to a MAP point for the GP* $\xi \sim \mathcal{N}(0, \mathcal{K})$

[5]Relaxing the interpolation constraints renders the invertibility assumption on Θ unnecessary. Nevertheless we keep it for ease of presentation.

conditioned on PDE constraints at the collocation points. To prove this will require extension of the approach introduced in [61], for example, and is left for future work. Here we describe this connection informally in the absence of the equality constraints. Consider the prior measure $\mu_0 = \mathcal{N}(0, \mathcal{K})$ *and consider the measurements* $\mathbf{o} = G([\boldsymbol{\phi}, u]) + \eta, \mathbf{y} = F([\boldsymbol{\phi}, u]) + \eta'$, $\eta \sim \mathcal{N}(0, \gamma^2 I), \eta' \sim \mathcal{N}(0, \beta^2 I)$. *It then follows from Bayes' rule [261] that the posterior measure of* u *given the data* $(\mathbf{o}, \mathbf{y})$ *is identified as the measure*

$$\frac{\mathrm{d}\mu^{(\mathbf{o},\mathbf{y})}}{\mathrm{d}\mu_0}(u) = \frac{1}{Z^{(\mathbf{o},\mathbf{u})}} \exp\left(-\frac{1}{2\gamma^2}|G([\boldsymbol{\phi}, u]) - \mathbf{o}|^2 - \frac{1}{2\beta^2}|F([\boldsymbol{\phi}, u]) - \mathbf{y}|^2\right),$$
$$Z^{(\mathbf{o},\mathbf{y})} := \mathbb{E}_{u\sim\mu_0} \exp\left(-\frac{1}{2\gamma^2}|G([\boldsymbol{\phi}, u]) - \mathbf{o}|^2 - \frac{1}{2\beta^2}|F([\boldsymbol{\phi}, u]) - \mathbf{y}|^2\right).$$

The article [61] showed that the MAP estimators of $\mu^{(\mathbf{o},\mathbf{y})}$ *are solutions to*

$$\underset{u\in\mathcal{U}}{\text{minimize}} \quad \|u\|^2 + \frac{1}{\gamma^2}|G([\boldsymbol{\phi}, u]) - \mathbf{o}|^2 + \frac{1}{\beta^2}|F([\boldsymbol{\phi}, u]) - \mathbf{y}|^2.$$

Letting $\beta \to 0$ *then yields* (4.2.5).

4.3 Solving Nonlinear PDEs

In this section, we present our framework for solution of nonlinear PDEs by conditioning GPs on nonlinear constraints. In Subsection 4.3.1 we outline our abstract setting as well as our assumptions on PDEs of interest; this leads to Corollary 4.3.2 which states an analogue of Proposition 4.2.3 in the PDE setting. We analyze the convergence of our method in Subsection 4.3.2 and discuss two strategies for dealing with the nonlinear PDE constraints in Subsection 4.3.3. Next, we present the details pertaining to numerical implementations of our method, including the choice of kernels and a Gauss–Newton algorithm in Subsection 4.3.4. Finally, we present a set of numerical experiments in Subsection 4.3.5 that demonstrate the effectiveness of our method in the context of prototypical nonlinear PDEs.

4.3.1 Problem Setup

Let us consider a bounded domain $\Omega \subseteq \mathbb{R}^d$ for $d \geq 1$ and a nonlinear PDE of the form

$$\begin{cases} \mathcal{P}(u^\star)(\mathbf{x}) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ \mathcal{B}(u^\star)(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega. \end{cases} \tag{4.3.1}$$

Here $\mathcal{P}$ is a nonlinear differential operator and $\mathcal{B}$ is an appropriate boundary operator with data f, g. Throughout this section and for brevity, we assume that the PDE at hand is well-defined pointwise and has a unique strong solution; extension of

our methodology to weak solutions is left as a future research direction. We then consider $\mathcal{U}$ to be an appropriate quadratic Banach space for the solution $u^\star$ such as a Sobolev space $H^s(\Omega)$ with sufficiently large regularity index $s > 0$.

We propose to solve the PDE (4.3.1) by approximating $u^\star$ by a GP conditioned on satisfying the PDE at a finite set of collocation points in $\overline{\Omega}$ and proceed to approximate the solution by computing the MAP point of such a conditioned GP. More precisely, let $\{\mathbf{x}_i\}_{i=1}^M$ be a collection of points in $\overline{\Omega}$ ordered in such a way that $\mathbf{x}_1, \ldots \mathbf{x}_{M_\Omega} \in \Omega$ are in the interior of Ω while $\mathbf{x}_{M_\Omega+1}, \ldots, \mathbf{x}_M \in \partial\Omega$ are on the boundary. Now let $\mathcal{U}$ be a quadratic Banach space with associated covariance operator $\mathcal{K} : \mathcal{U}^t \to \mathcal{U}$ and consider the optimization problem:

$$\begin{cases} \underset{u\in\mathcal{U}}{\text{minimize}} & \|u\| \\ \text{s.t.} & \mathcal{P}(u)(\mathbf{x}_m) = f(\mathbf{x}_m), \quad \text{for } m = 1, \ldots, M\,, \\ & \mathcal{B}(u)(\mathbf{x}_m) = g(\mathbf{x}_m), \quad \text{for } m = M_\Omega + 1, \ldots, M\,. \end{cases} \tag{4.3.2}$$

In other words, we wish to approximate $u^\star$ with the minimum norm element of the Cameron–Martin space of $\mathcal{N}(0, \mathcal{K})$ that satisfies the PDE and boundary data at the collocation points $\{\mathbf{x}_i\}_{i=1}^M$. In what follows we write $\mathcal{L}(\mathcal{U}; \mathcal{H})$ to denote the space of bounded and linear operators from $\mathcal{U}$ to another Banach space $\mathcal{H}$. We make the following assumption regarding the operators $\mathcal{P}, \mathcal{B}$:

Assumption 4.3.1. *There exist bounded and linear operators $L_1, \ldots, L_Q \in \mathcal{L}(\mathcal{U}; C(\Omega))$ in which $L_1, \ldots, L_{Q_b} \in \mathcal{L}(\mathcal{U}; C(\partial\Omega))$ for some $1 \leq Q_b \leq Q$, and there are maps $P : \mathbb{R}^Q \to \mathbb{R}$ and $B : \mathbb{R}^{Q_b} \to \mathbb{R}$, which may be nonlinear, so that $\mathcal{P}(u)(\mathbf{x})$ and $\mathcal{B}(u)(\mathbf{x})$ can be written as*

$$\begin{aligned} \mathcal{P}(u)(\mathbf{x}) &= P\big(L_1(u)(\mathbf{x}), \ldots, L_Q(u)(\mathbf{x})\big), \quad \forall \mathbf{x} \in \Omega\,, \\ \mathcal{B}(u)(\mathbf{x}) &= B\big(L_1(u)(\mathbf{x}), \ldots, L_{Q_b}(u)(\mathbf{x})\big), \quad \forall \mathbf{x} \in \partial\Omega\,. \end{aligned} \tag{4.3.3}$$

For prototypical nonlinear PDEs the L_q for $1 \leq q \leq Q$ are linear differential operators such as first or second order derivatives while the maps P and B are often simple algebraic nonlinearities. Furthermore, observe that for ease of presentation we are assuming fewer linear operators are used to define the boundary conditions than the operators that define the PDE in the interior.

Example NE (Nonlinear Elliptic PDE)**.** *Recall the nonlinear elliptic PDE* (4.1.1) *and consider the linear operators and nonlinear maps*

$$L_1 : u \mapsto u, \quad L_2 : u \mapsto \Delta u, \quad P(v_1, v_2) = -v_2 + \tau(v_1), \quad B(v_1) = v_1\,,$$

where we took $Q = 2$ and $Q_b = 1$. Then this equation readily satisfies Assumption 4.3.1 whenever the solution is sufficiently regular so that $L_2(u)$ is well-defined pointwise within Ω. ◇

Under Assumption 4.3.1 we can then define the functionals $\phi_m^{(q)} \in \mathcal{U}^t$ by setting

$$\phi_m^{(q)} := \delta_{\mathbf{x}_m} \circ L_q, \quad \text{where} \quad \begin{cases} 1 \le m \le M, & \text{if} \quad 1 \le q \le Q_b, \\ 1 \le m \le M_\Omega, & \text{if} \quad Q_{b+1} \le q \le Q. \end{cases} \tag{4.3.4}$$

We further use the shorthand notation $\boldsymbol{\phi}^{(q)}$ to denote the vector of dual elements $\phi_m^{(q)}$ for a fixed index q. Observe that $\boldsymbol{\phi}^{(q)} \in (\mathcal{U}^t)^{\otimes M}$ if $q \le Q_b$ while $\boldsymbol{\phi}^{(q)} \in (\mathcal{U}^t)^{\otimes M_\Omega}$ if $q > Q_b$. We further write

$$N = MQ_b + M_\Omega(Q - Q_b) \tag{4.3.5}$$

and define

$$\boldsymbol{\phi} := (\boldsymbol{\phi}^{(1)}, \dots, \boldsymbol{\phi}^{(Q)}) \in (\mathcal{U}^t)^{\otimes N}. \tag{4.3.6}$$

To this end, we define the measurement vector $\mathbf{y} \in \mathbb{R}^M$ by setting

$$y_m = \begin{cases} f(\mathbf{x}_m), & \text{if } m \in \{1, \dots, M_\Omega\}, \\ g(\mathbf{x}_m), & \text{if } m \in \{M_\Omega + 1, \dots, M\}, \end{cases} \tag{4.3.7}$$

as well as the nonlinear map

$$\big(F([\boldsymbol{\phi}, u])\big)_m := \begin{cases} P([\phi_m^{(1)}, u], \dots, [\phi_m^{(Q)}, u]) & \text{if } m \in \{1, \dots, M_\Omega\}, \\ B([\phi_m^{(1)}, u], \dots, [\phi_m^{(Q_b)}, u]) & \text{if } m \in \{M_\Omega + 1, \dots, M\}. \end{cases} \tag{4.3.8}$$

We can now rewrite the optimization problem (4.3.2) in the same form as (4.2.5):

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} & \|u\| \\ \text{s.t.} & F([\boldsymbol{\phi}, u]) = \mathbf{y}. \end{cases} \tag{4.3.9}$$

Then a direct appliction of Proposition 4.2.3 yields the following corollary.

Corollary 4.3.2. *Suppose Assumption 4.3.1 holds, $\mathcal{K}$ and Θ are invertible, and define $\boldsymbol{\phi}, F, \mathbf{y}$ as above. Then $u^\dagger$ is a minimizer of* (4.3.2) *if and only if $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ where the χ_n are the gamblets defined according to* (4.2.4) *and $\mathbf{z}^\dagger$ is a minimizer of*

$$\begin{cases} \underset{\mathbf{z} \in \mathbb{R}^N}{\text{minimize}} & \mathbf{z}^T \Theta^{-1} \mathbf{z} \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}. \end{cases} \tag{4.3.10}$$

The above corollary is the foundation of our numerical algorithms for approximation of the solution $u^\dagger$, as Θ^{-1} and the gamblets χ_n can be approximated offline while the coefficients $z_n^\dagger$ can be computed by solving the optimization problem (4.3.10).

To solve (4.3.10) numerically, we will present two different approaches that transform it to an unconstrained optimization problem. Before moving to that in Subsection 4.3.3, we discuss the convergence theory first in the next section.

4.3.2 Convergence Theory

We state and prove a more general version of Theorem 4.1.2 for our abstract setting of PDEs on Banach spaces with quadratic norms stating that a minimizer $u^\dagger$ of (4.3.2) converges to the true solution $u^\star$ under sufficient regularity assumptions and for appropriate choices of the operator $\mathcal{K}$.

Theorem 4.3.3. *Consider the PDE* (4.3.1) *and suppose that* $\mathcal{U} \subset \mathcal{H} \subset C^t(\Omega) \cap C^{t'}(\overline{\Omega})$ *where* $\mathcal{H}$ *is a Banach space such that the first inclusion from the left is given by a compact embedding and* $t \geq t' \geq 0$ *are sufficiently large so that all derivatives appearing in the PDE are defined pointwise for elements of* $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$. *Furthermore assume that the PDE has a unique classical solution* $u^\star \in \mathcal{U}$ *and that, as* $M \to \infty$,

$$\sup_{\mathbf{x}\in\Omega} \min_{1\leq m\leq M_\Omega} |\mathbf{x}-\mathbf{x}_m| \to 0 \quad \textit{and} \quad \sup_{\mathbf{x}\in\partial\Omega} \min_{M_\Omega+1\leq m\leq M} |\mathbf{x}-\mathbf{x}_m| \to 0.$$

Write $u_M^\dagger$ *for a minimizer of* (4.3.2) *with* M *distinct collocation points. Then, as* $M \to \infty$, *the sequence of minimizers* $u_M^\dagger$ *converges towards* $u^\star$ *pointwise in* Ω *and in* $\mathcal{H}$.

Proof. The method of proof is similar to that of Theorem 4.1.2. Indeed, by the same argument as in the first paragraph of the proof for Theorem 4.1.2, there exists a subsequence u_{M_p} that converges to $u_\infty^\dagger$ in $\mathcal{H}$. This also implies convergence in $C^t(\Omega)$ and $C^{t'}(\overline{\Omega})$ due to the assumed continuous embedding of $\mathcal{H}$ into $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$. Since $t \geq t' \geq 0$ are sufficiently large so that all derivatives appearing in the PDE are defined pointwise for elements of $C^t(\Omega) \cap C^{t'}(\partial\Omega)$, we get that $\mathcal{P}u_{M_p}$ converges to $\mathcal{P}u_\infty^\dagger$ in $C(\Omega)$ and $\mathcal{P}u_\infty^\dagger \in C(\Omega)$. As Ω is compact, $u_\infty^\dagger$ is also uniformly continuous in Ω.

For any $\mathbf{x} \in \Omega$ and $p \geq 1$, the triangle inequality shows that

$$\begin{aligned} |\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - f(\mathbf{x})| &\leq \min_{1\leq m\leq M_{p,\Omega}} \left(|\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - \mathcal{P}(u^\dagger)(\mathbf{x}_m)| + |\mathcal{P}(u^\dagger)(\mathbf{x}_m) - \mathcal{P}(u_{M_p})(\mathbf{x}_m)| \right) \\ &\leq \min_{1\leq m\leq M_{p,\Omega}} |\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - \mathcal{P}(u_\infty^\dagger)(\mathbf{x}_m)| + \|\mathcal{P}u_\infty^\dagger - \mathcal{P}u_{M_p}\|_{C(\Omega)}, \end{aligned} \tag{4.3.11}$$

where in the first inequality we have used the fact that $\mathcal{P}(u_{M_p})(\mathbf{x}_m) = f(\mathbf{x}_m)$. Here $M_{p,\Omega}$ is the number of interior points associated with the total M_p collocation points. Taking $p \to \infty$ and using the uniform continuity of $\mathcal{P}u_\infty^\dagger$ and the $C(\Omega)$ convergence from $\mathcal{P}u_{M_p}$ to $\mathcal{P}u_\infty^\dagger$, we derive that $\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) = f(\mathbf{x})$. In a similar manner we can derive $\mathcal{B}(u_\infty^\dagger)(\mathbf{x}) = g(\mathbf{x})$. Thus, the limit $u_\infty^\dagger$ is a classical solution to the PDE. By the uniqueness of the solution we must have $u_\infty^\dagger = u^\star$. Finally, as the limit $u_\infty^\dagger$ is independent of the choice of the subsequence, the whole sequence $u_M^\dagger$ must converge to $u^\star$ pointwise and in $\mathcal{H}$. □

We note that while this theorem does not provide a rate for convergence of $u^\dagger$ towards $u^\star$ it relies on straightforward conditions that are readily verifiable for prototypical PDEs. Typically we choose $t, t' > 0$ large enough so that the PDE operators $\mathcal{P}, \mathcal{B}$ are pointwise defined for the elements of $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$ (e.g., $t >$ order of PDE $+d/2$) and take the space $\mathcal{H}$ to be a Sobolev-type space of appropriate regularity for the inclusion $\mathcal{H} \subset C^t(\Omega) \cap C^{t'}(\partial\Omega)$ to hold; also see the conditions of Theorem 4.1.2 and the subsequent discussion. The compact embedding $\mathcal{U} \subset \mathcal{H}$ can then be ensured by an appropriate choice of the covariance operator $\mathcal{K}$ (or the associated kernel K). However, this choice should result in a sufficiently large space $\mathcal{U}$ that includes the solution $u^\star$ of the PDE. Our conditions on the collocation points $\{\mathbf{x}_m\}_{m=1}^M$ simply ensure that these points form a dense subset of $\overline{\Omega}$ as $M \to \infty$.

4.3.3 Dealing with the Constraints

Now, we turn our attention to the equality constraints in (4.3.10) and present two strategies for elimination or relaxation of these constraints; these transform the optimization problem to an unconstrained one. They are crucial preliminary steps before introducing our numerical framework.

4.3.3.1 Eliminating the Equality Constraints

The equality constraints in (4.3.10) can be eliminated under slightly stronger assumptions on the maps P, B. In particular, suppose that the following assumption holds:

Assumption 4.3.4. *The equations*

$$P(v_1, \dots, v_Q) = y, \qquad B(v_1, \dots, v_{Q_b}) = y\,,$$

can be solved as finite-dimensional algebraic equations, i.e., there exist $\overline{P} : \mathbb{R}^{Q-1} \to \mathbb{R}$ *and* $\overline{B} : \mathbb{R}^{Q_b-1} \to \mathbb{R}$ *so that*

$$v_j = \overline{P}(v_1, \dots, v_{j-1}, v_{j+1}, \dots, v_Q, y), \qquad v_k = \overline{B}(v_1, \dots, v_{k-1}, v_{k+1}, \dots, v_{Q_b}, y)\,, \tag{4.3.12}$$

for selected indices $j \in \{1, \dots, Q\}$ *and* $k \in \{1, \dots, Q_b\}$*. Then for integer* N *defined by* (4.3.5)*, and using the solution maps* $\overline{P}, \overline{B}$*, we can then define a new solution map* $\overline{F} : \mathbb{R}^{N-M} \times \mathbb{R}^M \to \mathbb{R}^N$ *so that*

$$F(\mathbf{z}) = \mathbf{y} \quad \text{if and only if} \quad \mathbf{z} = \overline{F}(\mathbf{w}, \mathbf{y}), \quad \text{for a unique } \mathbf{w} \in \mathbb{R}^{N-M}\,.$$

With this new solution map we can rewrite (4.3.10) as an unconstrained optimization problem.

Corollary 4.3.5. *Let Assumption 4.3.4 and the conditions of Corollary 4.3.2 hold. Then* $u^\dagger$ *is a minimizer of* (4.3.2) *if and only if* $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ *with* $\mathbf{z}^\dagger = F'(\mathbf{w}^\dagger, \mathbf{y})$ *and* $\mathbf{w}^\dagger \in \mathbb{R}^{N-M}$ *is a minimizer of*

$$\underset{\mathbf{w} \in \mathbb{R}^{N-M}}{\text{minimize}} \quad \overline{F}(\mathbf{w}, \mathbf{y})^T \Theta^{-1} \overline{F}(\mathbf{w}, \mathbf{y})\,. \tag{4.3.13}$$

Example NE. *Let us recall that we already eliminated the equality constraints in the context of the PDE* (4.1.1) *through the calculations leading to the unconstrained minimization problem* (4.1.6)*. In that example, we used the calculation*

$$P(v_1, v_2) = -v_2 + \tau(v_1) = y \Leftrightarrow v_2 = \tau(v_1) - y = \overline{P}(v_1, y)\,,$$

that is, we solved Δu *in terms of* $\tau(u)$ *and the source term in the interior of the domain in order to eliminate the PDE constraint. We further imposed the boundary conditions exactly since the boundary map* B *is simply the pointwise evaluation function in that example.*

Alternatively, we could eliminate v_1 *by setting* $v_1 = \tau^{-1}(y + v_2)$*, assuming that* τ^{-1} *has closed form. While both elimination strategies are conceptually valid they may lead to very different optimization problems. The former corresponds to solving for the values of* u *at the collocation points while the latter solves for the values of* Δu *at the interior points under Dirichlet boundary conditions at the boundary collocation points.* ◇

4.3.3.2 Relaxing the Equality Constraints

The choice of the solution maps $\overline{P}, \overline{B}$ in (4.3.12), i.e., the choice of the variable which the equations are solved for, has an impact on the conditioning of (4.3.13); it is not a priori clear that poor conditioning can always be avoided by choice of variables to solve for. Moreover, for certain nonlinear PDEs Assumption 4.3.4 may not hold. In such cases it may be useful to relax the equality constraints in (4.3.9) and instead consider a loss of the following form:

$$\underset{u\in\mathcal{U}}{\text{minimize}} \quad \|u\|^2 + \frac{1}{\beta^2}|F([\boldsymbol{\phi}, u]) - \mathbf{y}|^2\,, \tag{4.3.14}$$

where $\beta^2 > 0$ is a small positive parameter. Likewise (4.3.10) can be relaxed to obtain

$$\underset{\mathbf{z}\in\mathbb{R}^N}{\text{minimize}} \quad \mathbf{z}^T\Theta^{-1}\mathbf{z} + \frac{1}{\beta^2}|F(\mathbf{z}) - \mathbf{y}|^2\,. \tag{4.3.15}$$

Then a similar argument to the proof of Theorem 4.3.3 can be used to show that a minimizer of the relaxed optimization problem for u converges to the solution $u^\star$ of the PDE as the number of collocation points M increases and the parameter β vanishes.

Proposition 4.3.6. *Fix $\beta > 0$. Then the optimization problem* (4.3.14) *has minimizer $u^\dagger_{\beta,M}$ which (assuming Θ to be invertible) may be expressed in the form*

$$u^\dagger_{\beta,M} := \sum_{n=1}^{N} z^\dagger_{\beta,n}\chi_n \in \mathcal{U}\,,$$

where $\mathbf{z}^\dagger_\beta$ denotes a minimizer of (4.3.15)*. Under the assumptions of Theorem 4.3.3 it follows that, as $(\beta, M^{-1}) \to 0$, the relaxed estimator $u^\dagger_{\beta,M}$ converges to $u^\star$ pointwise and in $\mathcal{H}$.*

Proof. By the arguments used in Proposition 4.2.3 the minimizer of (4.3.15) delivers a minimizer of (4.3.14) in the desired form. Since $u^\star$ satisfies $F([\boldsymbol{\phi}, u^\star]) - \mathbf{y} = 0$ we must have $\|u^\dagger_{\beta,M}\| \leq \|u^\star\|$. Then a compactness argument similar to that used in the proof of Theorem 4.3.3, noting that taking $\beta \to 0$ as $M \to \infty$ delivers exact satisfaction of the constraints in the limit, yields the desired result. □

Example NE. *When only part of the constraints $F(\mathbf{z}) = \mathbf{y}$ can be explicitly solved, as is often the case for boundary values, we can also combine the elimination and relaxation approach. Employing the relaxation approach for the interior constraint and the elimination approach for the boundary constraints in* (4.1.1) *amounts to*

replacing the optimization problem (4.1.5)*, which is the analogue of* (4.3.10) *for our running example, with the following problem for a small parameter* $\beta^2 > 0$*:*

$$\begin{cases} \underset{\mathbf{z}\in\mathbb{R}^{M+M_\Omega}}{\text{minimize}} \quad \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} + \dfrac{1}{\beta^2}\left[\displaystyle\sum_{m=1}^{M_\Omega}\left|z_m^{(2)} + \tau(z_m^{(1)}) - f(\mathbf{x}_m)\right|^2\right] \\ \qquad s.t. \quad z_m^{(1)} = g(\mathbf{x}_m), \quad \textit{for } m = M_\Omega + 1, ..., M\,. \end{cases} \tag{4.3.16}$$

We will numerically compare the above approach with the full elimination approach (4.1.6) *in Subsection 4.3.5.1.* ◇

4.3.4 Implementation

We now outline the details of a numerical algorithm for solution of nonlinear PDEs based on the discussions of the previous subsection and in particular Corollary 4.3.2. We discuss the construction of the matrix Θ in Subsection 4.3.4.1 followed by a variant of the Gauss–Newton algorithm in Subsection 4.3.4.2 for solving the unconstrained or relaxed problems outlined in Subsections 4.3.3. We also note that strategies for regularizing the matrix Θ by adding small diagonal ("nugget") terms are collected in Appendix A.1.

4.3.4.1 Constructing Θ

We established through Corollary 4.3.2 that a solution to (4.3.2) can be completely identified by $\mathbf{z}^\dagger$ a minimizer of (4.3.10) as well as the gamblets χ_n. Since here we are concerned with the strong form of the PDE (4.3.1) it is reasonable to assume that at the very least $\mathcal{U} \subset C(\overline{\Omega})$; although we often require higher regularity so that the PDE constraints can be imposed pointwise. This assumption suggests that our GP model for $u^\star$ can equivalently be identified via a covariance kernel function as opposed to the covariance operator $\mathcal{K}$. To this end, given a covariance operator $\mathcal{K}$ define the covariance kernel K (equivalently Green's function of $\mathcal{K}^{-1}$) as

$$K : \overline{\Omega}\times\overline{\Omega} \mapsto \mathbb{R}, \qquad K(\mathbf{x},\mathbf{x}') := [\delta_{\mathbf{x}}, \mathcal{K}\delta_{\mathbf{x}'}]\,. \tag{4.3.17}$$

It is known that the kernel K completely characterizes the GP $\mathcal{N}(0,\mathcal{K})$ under mild conditions [276]; that is $\mathcal{N}(0,\mathcal{K}) \equiv \mathcal{GP}(0,K)$. Let us now consider the matrix Θ in block form

$$\Theta = \begin{bmatrix} \Theta^{(1,1)} & \Theta^{(1,2)} & \dots & \Theta^{(1,Q)} \\ \Theta^{(2,1)} & \Theta^{(2,2)} & \dots & \Theta^{(2,Q)} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta^{(Q,1)} & \Theta^{(Q,2)} & \dots & \Theta^{(Q,Q)} \end{bmatrix}.$$

Using the $L^2(\Omega)$ duality pairing between $\mathcal{U}$ and $\mathcal{U}^t$ we can identify the blocks

$$\Theta^{(q,j)} = K(\boldsymbol{\phi}^{(q)}, \boldsymbol{\phi}^{(j)}),$$

where we used the shorthand notation of Subsection 4.1.1 for the kernel matrix, with the $\boldsymbol{\phi}^{(q)}$ defined as in (4.3.4) and the subsequent discussion. To this end the entries of the $\Theta^{(q,j)}$ take the form

$$\Theta^{(q,j)}_{m,i} = L^{\mathbf{x}}_q L^{\mathbf{x}'}_j K(\mathbf{x}, \mathbf{x}')\big|_{(\mathbf{x},\mathbf{x}')=(\mathbf{x}_m,\mathbf{x}_i)},$$

where we used the superscripts $\mathbf{x}, \mathbf{x}'$ to denote the variables with respect to which the differential operators L_q, L_j act. Note that $\Theta \in \mathbb{R}^{N\times N}$ with $N = MQ_b + M_\Omega(Q - Q_b)$ following the definition of $\boldsymbol{\phi}^{(q)}$ in Subsection 4.3.1.

4.3.4.2 A Gauss–Newton Algorithm

Here we outline a variant of the Gauss–Newton algorithm [197, Sec. 10.3] for solution of the unconstrained optimization problem (4.3.13). Recall our definition of the maps $\overline{P}, \overline{B}$ in (4.3.12) and in turn the map $\overline{F}$. We then propose to approximate a minimizer $\mathbf{w}^\dagger$ of (4.3.13) with a sequence of elements $\mathbf{w}^\ell$ defined iteratively via $\mathbf{w}^{\ell+1} = \mathbf{w}^\ell + \alpha^\ell \delta\mathbf{w}^\ell$, where $\alpha^\ell > 0$ is an appropriate step size while $\delta\mathbf{w}^\ell$ is the minimizer of the optimization problem

$$\underset{\delta\mathbf{w}\in\mathbb{R}^{N-M}}{\text{minimize}} \quad \left(\overline{F}(\mathbf{w}^\ell, \mathbf{y}) + \delta\mathbf{w}^T \nabla\overline{F}(\mathbf{w}^\ell, \mathbf{y})\right)^T \Theta^{-1} \left(\overline{F}(\mathbf{w}^\ell, \mathbf{y}) + \delta\mathbf{w}^T \nabla\overline{F}(\mathbf{w}^\ell, \mathbf{y})\right),$$

and the gradient of $\overline{F}$ is computed with respect to the $\mathbf{w}$ variable only. [6]

This approach can be applied also to solve the relaxed problem (4.3.6) where this time we consider the sequence of approximations $\mathbf{z}^{\ell+1} = \mathbf{z}^\ell + \alpha^\ell \delta\mathbf{z}^\ell$, where $\delta\mathbf{z}^\ell$ is the minimizer of

$$\underset{\delta\mathbf{z}\in\mathbb{R}^N}{\text{minimize}} \quad \left(\mathbf{z}^\ell + \delta\mathbf{z}\right)^T \Theta^{-1} \left(\mathbf{z}^\ell + \delta\mathbf{z}\right) + \frac{1}{\beta^2} \left|F(\mathbf{z}^\ell) + \delta\mathbf{z}^T \nabla F(\mathbf{z}^\ell) - \mathbf{y}\right|^2.$$

Since (4.3.4.2) and (4.3.4.2) are both quadratic in $\delta\mathbf{w}$ and $\delta\mathbf{z}$ respectively, they can be solved exactly and efficiently at each step and the step-size parameters α^ℓ can be fixed or computed adaptively using standard step-size selection techniques [197]. However, in our experiments in Section 4.3.5, we find that both algorithms converge quickly simply by setting $\alpha^\ell = 1$.

[6] Note that our proposed method is nothing more than the standard Gauss–Newton algorithm with Euclidean norm $|\cdot|$ defining the least-squares functional replaced with the weighted norm $|\Theta^{-1/2}\cdot|$ [197, Sec. 10.3].

Example NE. *Let us return once more to the nonlinear elliptic PDE considered in Subsection 4.1.1. Observe that* (4.1.6) *is precisely in the form of* (4.3.13) *and so in order to formulate our Gauss–Newton iterations we need to linearize the vector valued function*

$$\mathbf{w} \mapsto \big(\mathbf{w}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{w})\big)\,,$$

which can easily be achieved by linearizing τ. To this end, we solve (4.3.13) *via the iteration $\mathbf{w}^{\ell+1} = \mathbf{w}^\ell + \alpha^\ell \delta\mathbf{w}^\ell$ where $\delta\mathbf{w}^\ell$ is the minimizer of the functional*

$$\big(\mathbf{w}^\ell+\delta\mathbf{w}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega)-\tau(\mathbf{w}^\ell)-\delta\mathbf{w}^T\nabla\tau(\mathbf{w}^\ell)\big)K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\begin{pmatrix}\mathbf{w}^\ell+\delta\mathbf{w}\\ g(\mathbf{x}_{\partial\Omega})\\ f(\mathbf{x}_\Omega)-\tau(\mathbf{w}^\ell)-\delta\mathbf{w}^T\nabla\tau(\mathbf{w}^\ell)\end{pmatrix}.$$

We also note that the sequence of approximations obtained by the above implementation of Gauss–Newton coincides with successive kernel collocation approximations of the solution of the following particular linearization of the PDE,

$$-\Delta u + u\tau'(u^n) = f - \tau(u^n) + u^n\tau'(u^n)\,, \tag{4.3.18}$$

subject to the Dirichlet boundary conditions. ◇

4.3.4.3 Computational Bottlenecks

The primary computational cost of our method lies in the approximation of the matrix Θ^{-1}. Efficient factorizations and approximations of Θ^{-1} have been studied extensively in the GP regression literature [219] as well as spatial statistics, Kriging and numerical analysis (see [240, 241] and the discussions within). In this work, we do not employ these algorithms and choose instead to use standard $O(N^3)$ algorithms to factorize Θ.

The algorithm introduced in [240] is particularly interesting as it directly approximates the Cholesky factors of Θ^{-1} by querying a subset of the entries of Θ. In fact, that algorithm alleviates the need for a small diagonal regularization ("nugget") term by directly computing the Cholesky factors of Θ^{-1} from the entries of Θ. This could be done by extending the algorithm introduced and analyzed in [240]. This algorithm is based on the identification of an explicit formula for computing approximate Cholesky factors L minimizing the Kullback-Leibler divergence between $\mathcal{N}(0,\Theta^{-1})$ and $\mathcal{N}(0,LL^T)$ given a sparsity constraint on the entries of L. The proposed formula is equivalent to the Vecchia approximation [278] (popular in geostatistics). The

resulting algorithm outlined in [240] computes ϵ approximate Cholesky factors of Θ^{-1} in $O(N \log^{2d}(N/\epsilon))$ complexity by accessing $O(N \log^{d}(N/\epsilon))$ entries of Θ.

Another possible bottleneck is the computation of the gamblets χ_n. The articles [204, 241] show that the gamblets can be approximated with compactly supported functions in complexity $O(N \log^{2d+1}(N/\epsilon))$. We also note that the complexity-vs-accuracy guarantees of [204, 240, 241] have only been established for functionals ϕ_n that are Dirac delta functions and kernels K that are the Green's functions of arbitrary elliptic differential operators (mapping $H^s(\Omega)$ to $H^{-s}(\Omega)$). Extension of those results to functionals ϕ_n considered here is an interesting future direction.

4.3.5 Numerical Experiments for Nonlinear PDEs

In this subsection, we implement our algorithm to solve several nonlinear PDEs, including the nonlinear elliptic equation in Subsection 4.3.5.1, Burgers' equation in Subsection 4.3.5.2 and the regularized Eikonal equation in Subsection 4.3.5.3. For all of these equations, we will start with a fixed M and demonstrate the performance of our algorithm by showing the pattern of collocation points, the loss function history of the Gauss–Newton iteration, and contours of the solution errors. Then, we vary the value of M and study how the errors change with respect to M. We also compare the elimination and relaxation approaches for dealing with the nonlinear constraints.

All the experiments are conducted using Python with the JAX package for automatic differentiation[7]. In particular, we use automatic differentiation to form the kernel matrix Θ that involves derivatives of the kernel function, and to optimize the loss function via the Gauss–Newton method. Details on the choice of small diagonal regularization ("nugget") terms for these experiments are presented in Appendices A.1.2 through A.1.4.

Remark 4.3.7. *In all of the numerical experiments in this section we used a set of collocation points that are drawn randomly from the uniform distribution over the domain Ω, as opposed to the deterministic uniform grid used in Subsection 4.1.1.4. The choice of the random collocation points was made to highlight the flexibility of our methodology. Furthermore, random collocation points are often used in other machine learning algorithms for solution of PDEs such as PINNs [222] and so adopting this approach allows direct comparison with such methods. We*

[7]We use JAX for convenience and all derivatives in our methodology can be computed using standard techniques such as symbolic computation or adjoint methods.

observed empirically that the random grids had similar accuracy to the deterministic uniform grid in all experiments except for Burgers' equation in Subsection 4.3.5.2, where random collocation points outperformed the uniform grid. Understanding this surprising performance gap is an interesting problem related to active learning and the acquisition of collocation points; we do not address this issue here.

Remark 4.3.8. *Float64 data type was employed in the experiments below. This allows the use of small diagonal regularization ("nugget") terms (see Appendix A.1 for details) which do not affect accuracy in the computations described in this work. In contrast, if Float32 data type (the default setting in JAX) is used, we found the need to regularize Θ with larger diagonal terms, leading to an observable accuracy floor.*

4.3.5.1 A Nonlinear Elliptic PDE

We revisit again the nonlinear elliptic equation in (4.1.1). As in Subsection 4.1.1.4, we take $d = 2$, $\Omega = (0,1)^2$ and $\tau(u) = u^3$ together with homogeneous Dirichlet boundary conditions $g(\mathbf{x}) = 0$. The true solution is prescribed to be $u^\star(\mathbf{x}) = \sin(\pi x_1)\sin(\pi x_2) + 4\sin(4\pi x_1)\sin(4\pi x_2)$ and the corresponding right hand side $f(\mathbf{x})$ is computed using the equation. We choose the Gaussian kernel $K(\mathbf{x},\mathbf{y};\sigma) = \exp\left(-\frac{|\mathbf{x}-\mathbf{y}|^2}{2\sigma^2}\right)$ with a lengthscale parameter σ.

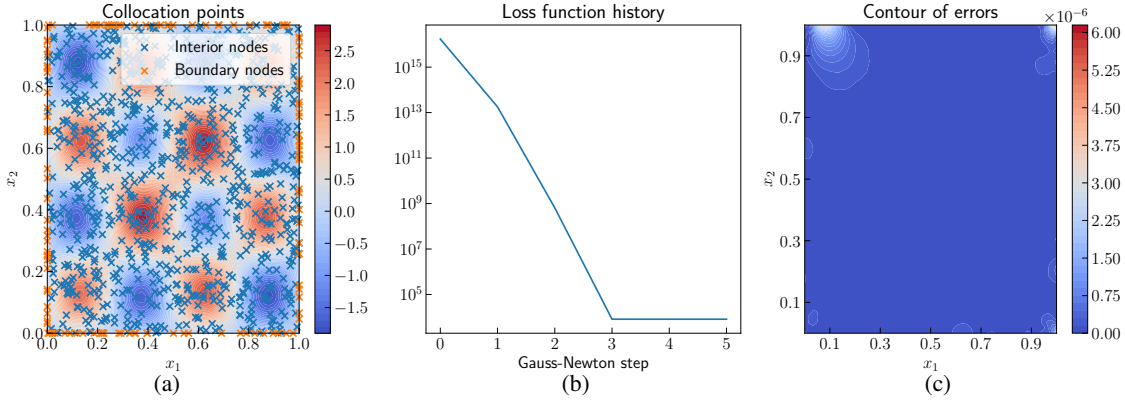


Figure 4.2: Numerical results for the nonlinear elliptic PDE (4.1.1): (a) a sample of collocation points and contours of the true solution; (b) convergence history of the Gauss–Newton algorithm; (c) contours of the solution error. An adaptive nugget term with global parameter $\eta = 10^{-13}$ was employed (see Appendix A.1.2)

.

First, for $M = 1024$ and $M_\Omega = 900$, we randomly sample collocation points in

Ω as shown in Figure 4.2(a). We further show an instance of the convergence history of the Gauss–Newton algorithm in Figure 4.2(b) where we solved the unconstrained optimization problem (4.1.6) after eliminating the equality constraints. We used kernel parameter $\sigma = M^{-1/4}$ and appropriate nugget terms as outlined in Appendix A.1.2. We initiated the algorithm with a Gaussian random initial guess. We observe that only 3 steps sufficed for convergence. In Figure 4.2(c), we show the contours of the solution error. The error in the approximate solution is seen to be fairly uniform spatially, with larger errors observed near the boundary, when $M = 1024$. We note that the main difference between these experiments and those in Subsection 4.1.1.4 is that here we used randomly distributed collocation points while a uniform grid was used previously.

Next, we compare two approaches for dealing with the PDE constraints as outlined in Subsection 4.3.3. We applied both the elimination and relaxation approaches, defined by the optimization problems (4.3.13) and (4.3.15) respectively, for different choices of M. In the relaxation approach, we set $\beta^2 = 10^{-10}$. Here we set $M = 300, 600, 1200, 2400$ and $M_\Omega = 0.9 \times M$. The L^2 and L^∞ errors of the converged Gauss–Newton solutions are shown in Table 4.1. Results were averaged over 10 realizations of the random collocation points. From the table we observe that the difference in solution errors was very mild and both methods were convergent as M increases. We note that in the relaxed setting, convergence is closely tied to our choice of β, and choosing an inadequate value, i.e. too small or too large, can lead to inaccurate solutions. In terms of computational costs, the elimination approaches take 2-3 steps of Gauss–Newton iterations on average, while the relaxation approach needs 5-8 steps. Thus while the elimination strategy appears to be more efficient, we do not observe a significant difference in the order of complexity of the methods for dealing with the constraints, especially when the number of collocation points becomes large.

4.3.5.2 Burgers' Equation

We consider numerical solution of the viscous Burgers equation:

$$\begin{aligned} \partial_t u + u\partial_s u - \nu\partial_s^2 u = 0, \quad \forall (s,t) \in (-1,1) \times (0,1], \\ u(s,0) = -\sin(\pi x), \\ u(-1,t) = u(1,t) = 0\,. \end{aligned} \tag{4.3.19}$$

M	300	600	1200	2400
Elimination: L^2 error	4.84e-02	6.20e-05	2.74e-06	2.83e-07
Elimination: L^∞ error	3.78e-01	9.71e-04	4.56e-05	5.08e-06
Relaxation: L^2 error	1.15e-01	1.15e-04	1.87e-06	1.68e-07
Relaxation: L^∞ error	1.21e+00	1.45e-03	3.38e-05	1.84e-06

Table 4.1: Comparison between the elimination and relaxation approaches to deal with the equality constraints for the nonlinear elliptic PDE (4.1.1). Uniformly random collocation points were sampled with different M and $M_\Omega = 0.9M$. Adaptive nugget terms were employed with the global nugget parameter $\eta = 10^{-12}$ (see Appendix A.1.2). The lengthscale parameter $\sigma = 0.2$. Results were averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 10.

We adopt an approach in which we solve the problem by conditioning a Gaussian process in space-time[8]. In our experiments we take $\nu = 0.02$ and consider $\mathbf{x} = (s, t)$. We write this PDE in the form of (4.3.3) with $Q = 4$ and $Q_b = 1$ with linear operators $L_1(u) = u, L_2(u) = \partial_t u, L_3(u) = \partial_s u, L_4(u) = \partial_s^2 u$ and the nonlinear map $P(v_1, v_2, v_3, v_4) = v_2 + v_1 v_3 - \nu v_4^2$. The boundary part is simply $B(v_1) = v_1$. We then eliminate the equality constraints in our optimization framework following the approach of Subsection 4.3.3.1 using the equation $v_2 = \nu v_4^2 - v_1 v_3$.

We randomly sampled $M = 2400$ with $M_\Omega = 2000$ points in the computational domain $\Omega = [-1, 1] \times [0, 1]$ see Figure 4.3(a), where we also plot contours of the true solution u. The Gauss–Newton algorithm was then applied to solve the unconstrained optimization problem. We computed the true solution from the Cole–Hopf transformation, together with the numerical quadrature. Since the time and space variability of the solution to Burgers' equation are significantly different, we chose an anisotropic kernel

$$K\Big((s,t),(s',t');\sigma\Big) = \exp\Big(-\sigma_1^{-2}(s-s')^2 - \sigma_2^{-2}(t-t')^2\Big)$$

with $\sigma = (1/20, 1/3)$ together with an adaptive diagonal regularization ("nugget") as outlined in Appendix A.1.3.

We plot the Gauss–Newton iteration history in Figure 4.3(b) and observe that 10 steps sufficed for convergence. We compare the converged solution to the true solution and present the contours of the error in Figure 4.3(c). The maximum errors

[8]It would also be possible to look at an incremental in time approach, for example using backward Euler discretization, in which one iteratively in time solves a nonlinear elliptic two point boundary value problem by conditioning a spatial Gaussian process; we do not pursue this here and leave it as a future direction.

occured close to the (viscous) shock at time 1 as expected. In Figure 4.3(d–f), we also compare various time slices of the numerical and true solutions at times $t = 0.2, 0.5, 0.8$ to further highlight the ability of our method in capturing the location and shape of the shock.

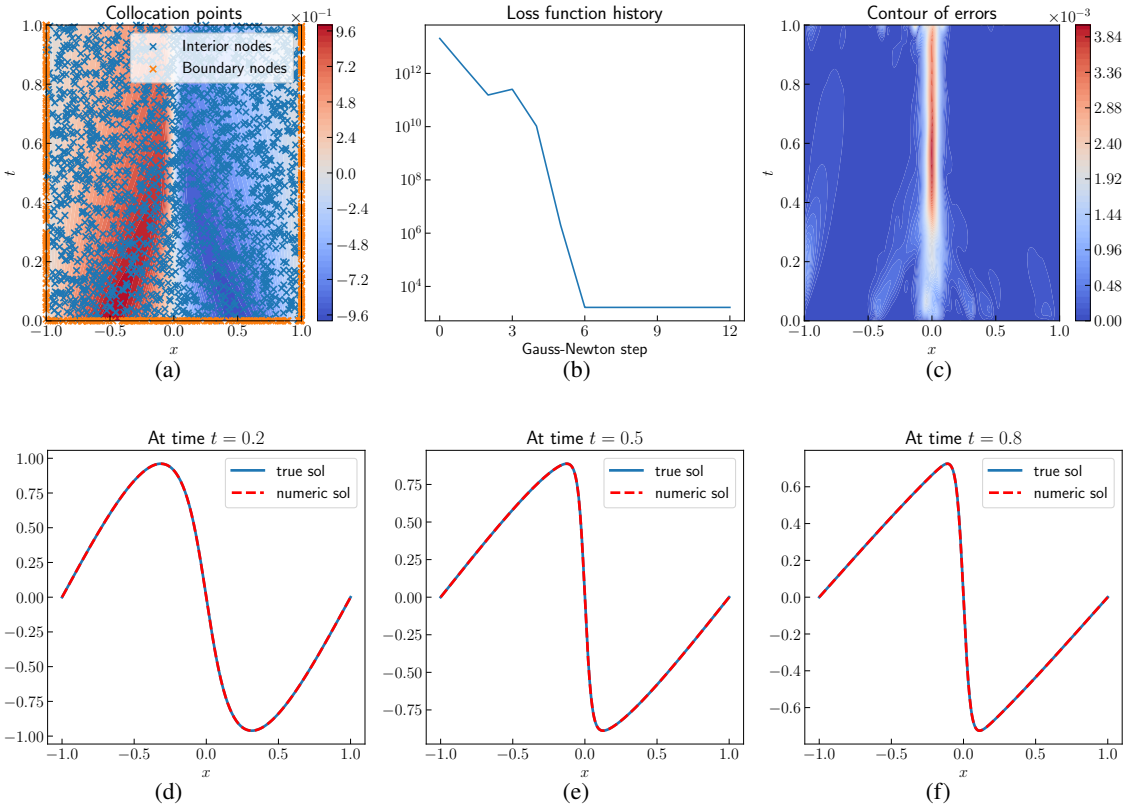


Figure 4.3: Numerical results for Burgers equation (4.3.19): (a) an instance of uniformly sampled collocation points in space-time over contours of the true solution; (b) Gauss–Newton iteration history; (c) contours of the pointwise error of the numerical solution; (d–f) time slices of the numerical and true solutions at $t = 0.2, 0.5, 0.8$. An adaptive nugget term with global parameter $\eta = 10^{-10}$ was employed (see Appendix A.1.3).

Next, we studied the convergence properties of our method as a function of M as shown in Table 4.2. Here, we varied M with a fixed ratio of interior points, $M_\Omega/M = 5/6$. For each experiment we ran 10 steps of Gauss–Newton starting from a Gaussian random initial guess. Results were averaged over 10 realizations of the random collocation points. From the table, we observe that the error decreases very fast as M increases, implying the convergence of our proposed algorithm.

Finally, we note that the accuracy of our method is closely tied to the choice of the

viscosity parameter ν and choosing a smaller value of ν, which in turn results in a sharper shock, can significantly reduce our accuracy. This phenomenon is not surprising since a sharper shock corresponds to the presence of shorter length and time scales in the solution; these in turn, require a more careful choice of the kernel, as well as suggesting the need to carefully choose the collocation points.

M	600	1200	2400	4800
L^2 error	1.75e-02	7.90e-03	8.65e-04	9.76e-05
L^∞ error	6.61e-01	6.39e-02	5.50e-03	7.36e-04

Table 4.2: Space-time L^2 and L^∞ solution errors for the Burgers' equation (4.3.19) for different choices of M with kernel parameters $\sigma = (20, 3)$ and global nugget parameter $\eta = 10^{-5}$ if $M \leq 1200$ and $\eta = 10^{-10}$ otherwise (see Appendix A.1.3). Results were averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 30.

4.3.5.3 Eikonal PDE

We now consider the regularized Eikonal equation in $\Omega = [0, 1]^2$:

$$\begin{cases} |\nabla u(\mathbf{x})|^2 = f(\mathbf{x})^2 + \epsilon \Delta u(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ u(\mathbf{x}) = 0, & \forall \mathbf{x} \in \partial\Omega, \end{cases} \tag{4.3.20}$$

with $f = 1$ and $\epsilon = 0.1$. We write this PDE in the form of (4.3.3) with $Q = 4$ and $Q_b = 1$ and linear operators $L_1(u) = u, L_2(u) = \partial_{x_1} u, L_3(u) = \partial_{x_2} u, L_4(u) = \Delta u$ and nonlinear map $P(v_1, v_2, v_3, v_4) = v_2^2 + v_3^2 - \epsilon v_4$ in the interior of Ω and define the boundary operator identically to Subsection 4.3.5.2. We further eliminate the nonlinear constraints, as outlined in Subsection 4.3.3.1, by solving v_4 in terms of v_2, v_3. To obtain a "true" solution, for the purpose of estimating errors, we employ the transformation $u = -\epsilon \log v$, which leads to the linear PDE $fv - \epsilon^2 \Delta v = 0$; we solve this by a highly-resolved FD method, and we used 2000 uniform grid points in each dimension of the domain leading to the finest mesh that our hardware could handle.

As before, we began with $M = 2400$ collocation points with $M_\Omega = 2160$ interior points. An instance of these collocation points along with contours of the true solution are shown in Figure 4.4(a). We employed a nugget term as outlined in Appendix A.1.4 and used the Gaussian kernel, as in Subsection 4.3.5.1 with $\sigma = M^{-1/4}$. Finally we used the Gauss–Newton algorithm to find the minimizer. We show the convergence history of Gauss–Newton in Figure 4.4(b), observing

that six iterations were sufficient for convergence. In Figure 4.4(c) we show the error contours of the obtained numerical approximation, which appeared to be qualitatively different to Figure 4.2(c) in that the errors were larger in the middle of the domain as well as close to the boundary.

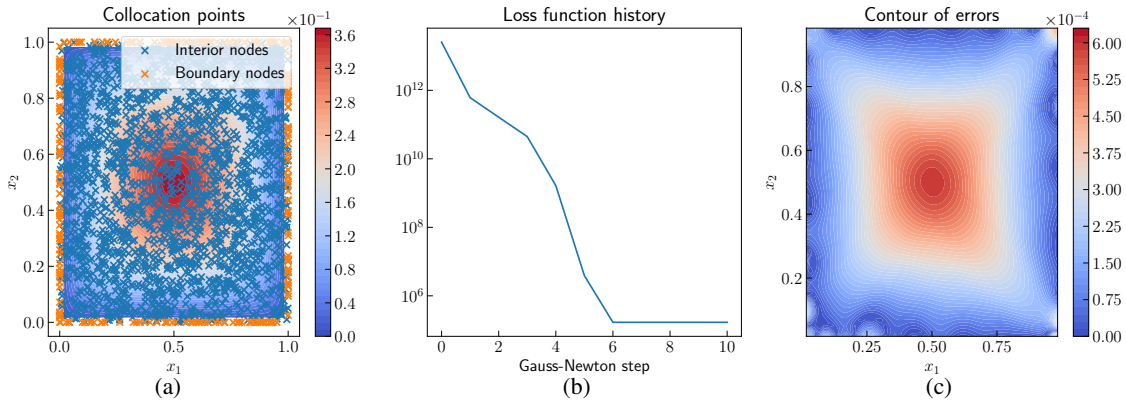


Figure 4.4: Numerical results for the regularized Eikonal equation (4.3.20): (a) an instance of uniformly sampled collocation points over contours of the true solution; (b) Gauss–Newton iteration history; (c) contour of the solution error. An adaptive nugget term with $\eta = 10^{-10}$ was used (see Appendix A.1.4).

Next we performed a convergence study by varying M and computing L^2 and L^∞ errors as reported in Table 4.3 by choosing the lengthscale parameter of the kernel $\sigma = M^{-1/4}$. We used the same nugget terms as in the Burgers' equation (see Appendix A.1.4). Results were averaged over 10 realizations of the random collocation points. Once again we observe clear improvements in accuracy as the number of collocation points increases.

M	300	600	1200	2400
L^2 error	1.01e-01	1.64e-02	2.27e-04	7.78e-05
L^∞ error	3.59e-01	7.76e-02	3.22e-03	1.61e-03

Table 4.3: Numerical results for the regularized Eikonal equation (4.3.20). Uniformly random collocation points were sampled with different M and with fixed ratio $M_\Omega = 0.9M$. An adaptive nugget term was used with global nugget parameter $\eta = 10^{-5}$ if $M \leq 1200$ and $\eta = 10^{-10}$ otherwise (see Appendix A.1.4), together with a Gaussian kernel with lengthscale parameter $\sigma = M^{-1/4}$. Results were averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 20.

4.4 Solving Inverse Problems

We now turn our attention to solution of IPs and show that the methodology of Subsections 4.3.1–4.3.4 can readily be extended to solve such problems with small modifications. We descibe the abstract setting of our IPs in Subsection 4.4.1 leading to Corollary 4.4.4 which is analogous to Proposition 4.2.3 and Corollary 4.3.2 in the setting of IPs. Subsection 4.4.2 outlines our approach for dealing with PDE constraints in IPs and highlights the differences in this setting in comparison to the PDE setting described in Subsection 4.3.3. Subsection 4.4.3 further highlights the implementation differences between the PDE and IP settings while Subsection 4.4.4 presents a numerical experiment concerning an IP in subsurface flow governed by the Darcy flow PDE.

4.4.1 Problem Setup

Consider our usual setting of a nonlinear parameteric PDE in strong form

$$\begin{cases} \mathcal{P}(u^\star; a^\star)(\mathbf{x}) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega\,, \\ \mathcal{B}(u^\star; a^\star)(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega\,. \end{cases} \tag{4.4.1}$$

As before we assume the solution $u^\star$ belongs to a quadratic Banach space $\mathcal{U}$ while $a^\star$ is a parameter belonging to another quadratic Banach space $\mathcal{A}$. Our goal in this subsection is to identify the parameter $a^\star$ from limited observations of the solution $u^\star$. To this end, fix $\psi_1, \ldots, \psi_I \in \mathcal{U}^t$ and define

$$\boldsymbol{\psi} := (\psi_1, \ldots, \psi_I) \in (\mathcal{U}^t)^{\otimes I}\,, \tag{4.4.2}$$

then our goal is to recover $a^\star$ given the noisy observations

$$\mathbf{o} = [\boldsymbol{\psi}, u] + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \gamma^2 I)\,. \tag{4.4.3}$$

We propose to solve this inverse problem by modelling both $u^\star$ and $a^\star$ with canonical GPs on the spaces $\mathcal{U}, \mathcal{A}$ with invertible covariance operators $\mathcal{K} : \mathcal{U}^t \to \mathcal{U}$ and $\widetilde{\mathcal{K}} : \mathcal{A}^t \to \mathcal{A}$ respectively. We then condition these GPs to satisfy the PDE on collocation points $\mathbf{x}_1, \ldots, \mathbf{x}_M \in \overline{\Omega}$ as before and propose to approximate $u^\star, a^\star$ simultaneously via the optimization problem:

$$\begin{cases} \underset{(u,a)\in\mathcal{U}\times\mathcal{A}}{\text{minimize}} & \|u\|_{\mathcal{U}}^2 + \|a\|_{\mathcal{A}}^2 + \dfrac{1}{\gamma^2}\Big|[\boldsymbol{\psi}, u] - \mathbf{o}\Big|^2 & \\ \text{s.t.} & \mathcal{P}(u; a)(\mathbf{x}_m) = f(\mathbf{x}_m), & \text{for } m = 1, \ldots, M_\Omega\,, \\ & \mathcal{B}(u; a)(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \ldots, M\,, \end{cases} \tag{4.4.4}$$

where we used subscripts to distinguish the quadratic norms on the spaces $\mathcal{U}$ and $\mathcal{A}$.

Remark 4.4.1. *In light of Remark 4.2.4 we observe that* (4.4.4) *corresponds to imposing a prior measure on u, a which assumes they are a priori independent. It is straightforward to introduce correlations between the solution u and the parameter a by defining the prior measure directly on the product space $\mathcal{U} \times \mathcal{A}$. This perspective will then lead to an analogous optimization problem to* (4.4.4) *with the same constraints but with the functional*

$$\|(u,a)\|^2_{\mathcal{U}\times\mathcal{A}} + \frac{1}{\gamma^2}|[\boldsymbol{\psi}, u] - \mathbf{o}|^2,$$

where we used $\|\cdot\|_{\mathcal{U}\times\mathcal{A}}$ to denote the RKHS norm of the GP associated with $\mathcal{U}\times\mathcal{A}$.

Remark 4.4.2. *We also note that the Bayesian interpretation of* (4.4.4) *can be viewed as an extension of gradient matching [39, 165] from ODEs to PDEs. Indeed, gradient matching simultaneously approximates the unknown parameters and the solution of an ODE system using a joint GP prior and imposes the ODE as a constraint at finitely many time steps.*

We make analogous assumptions on the form of the operators $\mathcal{P}, \mathcal{B}$ as in Assumption 4.3.1 but this time also involving the parameters a:

Assumption 4.4.3. *There exist bounded and linear operators $L_1, \dots, L_Q \in \mathcal{L}(\mathcal{U}; C(\Omega))$ in which $L_1, \dots, L_{Q_b} \in \mathcal{L}(\mathcal{U}; C(\partial\Omega))$ for some $1 \le Q_b \le Q$, and $\widetilde{L}_1, \dots, \widetilde{L}_J \in \mathcal{L}(\mathcal{A}; C(\overline{\Omega}))$ together with maps $P : \mathbb{R}^{Q+J} \to \mathbb{R}$ and $B : \mathbb{R}^{Q_b+J} \to \mathbb{R}$, which may be nonlinear, so that $\mathcal{P}(u;a)(\mathbf{x})$ and $\mathcal{B}(u;a)(\mathbf{x})$ can be written as*

$$\begin{aligned}
\mathcal{P}(u;a)(\mathbf{x}) &= P\big(L_1(u)(\mathbf{x}), \dots, L_Q(u)(\mathbf{x}); \widetilde{L}_1(a)(\mathbf{x}), \dots \widetilde{L}_J(a)(\mathbf{x})\big), \quad \forall \mathbf{x} \in \Omega, \\
\mathcal{B}(u;a)(\mathbf{x}) &= B\big(L_1(u)(\mathbf{x}), \dots, L_{Q_b}(u)(\mathbf{x}); \widetilde{L}_1(a)(\mathbf{x}), \dots \widetilde{L}_J(a)(\mathbf{x})\big), \quad \forall \mathbf{x} \in \partial\Omega.
\end{aligned} \tag{4.4.5}$$

Similarly to the L_q, the $\widetilde{L}_j$ operators are also linear differential operators in case of prototypical PDEs while the maps P, B remain as simple algebraic nonlinearities. Let us briefly consider an IP in subsurface flow and verify the above assumption.

Example DF (Darcy flow IP)**.** *Let $\Omega = (0,1)^2$ and consider the Darcy flow PDE with Dirichlet boundary conditions*

$$\begin{cases} -div\left(\exp(a)\nabla u\right)(\mathbf{x}) = f(\mathbf{x}), & \mathbf{x} \in \Omega, \\ u(\mathbf{x}) = 0, & \mathbf{x} \in \partial\Omega. \end{cases}$$

We wish to approximate $a \in C^1(\overline{\Omega})$ given noisy pointwise observations of u at a set of points $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_I$. Thus, we take $\psi_i = \delta_{\tilde{\mathbf{x}}_i}$. By expanding the PDE we obtain

$$-div\,(\exp(a)\nabla u) = -\exp(a)\,(\nabla a \cdot \nabla u + \Delta u)\,,$$

and so we simply choose $Q = 3$, $Q_b = 1$ and $J = 2$ with the linear operators

$$L_1(u) = u, \quad L_2(u) = \nabla u, \quad L_3(u) = \Delta u, \quad \widetilde{L}_1(a) = a, \quad \widetilde{L}_2(a) = \nabla a\,.$$

We can then satisfy Assumption 4.4.5 by taking

$$P(v_1, \mathbf{v}_2, v_3; v_4, \mathbf{v}_5) = -\exp(v_4)\,(\mathbf{v}_5 \cdot \mathbf{v}_2 + v_3)\,, \qquad B(v_1; v_4, \mathbf{v}_5) = v_1\,, \tag{4.4.6}$$

where we have slightly abused notation by letting $L_2, \widetilde{L}_2$ be vector valued and defining P, B to take vectors as some of their inputs. ◇

As in Subsection 4.3.1 we now define the functionals $\phi_m^{(q)} \in \mathcal{U}^t$ for $m = 1, \dots, M$ and $q = 1, \dots, Q$ according to (4.3.4) and (4.3.6) with $N = MQ_b + M_\Omega(Q - Q_b)$. Similarly we define the functionals $\widetilde{\phi}_m^{(j)} \in \mathcal{A}^t$ as

$$\widetilde{\phi}_m^{(j)} := \delta_{\mathbf{x}_m} \circ \widetilde{L}_j, \qquad \text{for} \qquad m = 1, \dots, M, \text{ and } j = 1, \dots, J\,, \tag{4.4.7}$$

together with the vectors

$$\widetilde{\boldsymbol{\phi}}^{(j)} = (\widetilde{\phi}_1^{(j)}, \dots \widetilde{\phi}_M^{(j)}) \in (\mathcal{A}^t)^{\otimes M} \quad \text{and} \quad \widetilde{\boldsymbol{\phi}} = (\widetilde{\boldsymbol{\phi}}^{(1)}, \dots, \widetilde{\boldsymbol{\phi}}^{(J)}) \in (\mathcal{A}^t)^{\otimes \widetilde{N}}, \tag{4.4.8}$$

where $\widetilde{N} := MJ$. Similarly to (4.3.8) define the map

$$\big(F([\boldsymbol{\phi}, u]_{\mathcal{U}}; [\widetilde{\boldsymbol{\phi}}, a]_{\mathcal{A}})\big)_m :=$$
$$\begin{cases} P([\phi_m^{(1)}, u]_{\mathcal{U}}, \dots, [\phi_m^{(Q)}, u]_{\mathcal{U}}; [\widetilde{\phi}_m^{(1)}, a]_{\mathcal{A}}, \dots, [\widetilde{\phi}_m^{(J)}, a]_{\mathcal{A}}) & \text{if } m \in \{1, \dots, M_\Omega\}\,, \\ B([\phi_m^{(1)}, u]_{\mathcal{U}}, \dots, [\phi_m^{(Q_b)}, u]_{\mathcal{U}}; [\widetilde{\phi}_m^{(1)}, a]_{\mathcal{A}}, \dots, [\widetilde{\phi}_m^{(J)}, a]_{\mathcal{A}}) & \text{if } m \in \{M_\Omega + 1, \dots, M\}\,, \end{cases}$$

where we used subscripts to distinguish the duality pairings between $\mathcal{U}, \mathcal{U}^t$ and the pairing between $\mathcal{A}, \mathcal{A}^t$. With this new notation we can finally rewrite (4.4.4) in the familiar form

$$\begin{cases} \underset{(u,a) \in \mathcal{U} \times \mathcal{A}}{\text{minimize}} & \|u\|_{\mathcal{U}}^2 + \|a\|_{\mathcal{A}}^2 + \dfrac{1}{\gamma^2} |\boldsymbol{\psi}(u) - \mathbf{o}|^2 \\ \text{s.t.} & F\big([\boldsymbol{\phi}, u]_{\mathcal{U}}; [\widetilde{\boldsymbol{\phi}}, a]_{\mathcal{A}}\big) = \mathbf{y}, \end{cases} \tag{4.4.9}$$

with the PDE data vector $\mathbf{y} \in \mathbb{R}^M$ defined in (4.3.7).

We can now apply Proposition 4.2.3 with the canonical GP defined on the product space $\mathcal{U} \times \mathcal{A}$ and with a block diagonal covariance operator $\mathcal{K} \otimes \widetilde{\mathcal{K}}$ to obtain a representer theorem for minimizer of (4.4.9). We state this result as a corollary below after introducing some further notation. Define the vector $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_{N+I}) \in (\mathcal{U}^t)^{\otimes(I+N)}$, with entries [9]

$$\varphi_n := \begin{cases} \psi_n, & \text{if } n = 1, \ldots, I, \\ \phi_{n-I}, & \text{if } n = I+1, \ldots, I+N, \end{cases}$$

as well as the matrices $\Theta \in \mathbb{R}^{(I+N)\times(I+N)}$ and $\widetilde{\Theta} \in \mathbb{R}^{\widetilde{N}\times\widetilde{N}}$ with entries

$$\Theta_{i,n} = [\varphi_i, \mathcal{K}\varphi_n]_{\mathcal{U}} \qquad \text{and} \qquad \widetilde{\Theta}_{i,n} = [\widetilde{\phi}_i, \widetilde{\mathcal{K}}\widetilde{\phi}_n]_{\mathcal{A}}.$$

As in (4.2.3) we define the gamblets

$$\chi_i = \sum_{n=1}^{N+I} \Theta^{-1}_{i,n} \mathcal{K}\varphi_n, \qquad \text{and} \qquad \widetilde{\chi}_i = \sum_{n=1}^{\widetilde{N}} \widetilde{\Theta}^{-1}_{i,n} \widetilde{\mathcal{K}}\widetilde{\phi}_n.$$

Then Proposition 4.2.3 yields the following corollary.

Corollary 4.4.4. *Suppose Assumption 4.4.3 holds and that the covariance operators $\mathcal{K}$ and $\widetilde{\mathcal{K}}$ as well as the matrices Θ and $\widetilde{\Theta}$ are invertible. Then $(u^\dagger, a^\dagger) \in \mathcal{U} \times \mathcal{A}$ is a minimizer of* (4.3.2) *if and only if*

$$u^\dagger = \sum_{n=1}^{I+N} z_n^\dagger \chi_n, \qquad \textit{and} \qquad a^\dagger = \sum_{n=1}^{\widetilde{N}} \widetilde{z}_n^\dagger \widetilde{\chi}_n,$$

where the vectors $\mathbf{z}^\dagger, \widetilde{\mathbf{z}}^\dagger$ are minimizers of

$$\begin{cases} \underset{(\mathbf{z},\widetilde{\mathbf{z}})\in(\mathbb{R}^{I+N}\times\mathbb{R}^{\widetilde{N}})}{\text{minimize}} & \mathbf{z}^T\Theta^{-1}\mathbf{z} + \widetilde{\mathbf{z}}^T\widetilde{\Theta}^{-1}\widetilde{\mathbf{z}} + \dfrac{1}{\gamma^2}|\Pi^I\mathbf{z} - \mathbf{o}|^2 \\ \text{s.t.} & F(\Pi_N\mathbf{z}; \widetilde{\mathbf{z}}) = \mathbf{y}, \end{cases} \tag{4.4.10}$$

where $\Pi^I : \mathbb{R}^{I+N} \to \mathbb{R}^I$ is the projection that extracts the first I entries of a vector while $\Pi_N : \mathbb{R}^{I+N} \to \mathbb{R}^N$ is the projection that extracts the last N entries.

4.4.2 Dealing with the Constraints

The equality constraints in (4.4.10) can be dealt with using the same strategies as in Subsection 4.3.3. Indeed, as in Subsection 4.3.3.2, we can readily relax these

[9] Note that we are concatenating the I measurement functionals defining the data for the IP with the N functionals used to define the PDE at the collocation points.

constraints to obtain the optimization problem

$$\underset{(\mathbf{z},\widetilde{\mathbf{z}})\in(\mathbb{R}^{I+N}\times\mathbb{R}^{\widetilde{N}})}{\text{minimize}} \quad \mathbf{z}^T\Theta^{-1}\mathbf{z} + \widetilde{\mathbf{z}}^T\widetilde{\Theta}^{-1}\widetilde{\mathbf{z}} + \frac{1}{\gamma^2}|\Pi^I\mathbf{z} - \mathbf{o}|^2 + \frac{1}{\beta^2}\,|F(\Pi_N\mathbf{z};\widetilde{\mathbf{z}}) - \mathbf{y}|^2\,, \tag{4.4.11}$$

for a small parameter $\beta^2 > 0$. Elimination of the constraints as in Subsection 4.3.3.1 is slightly more delicate, but is sometimes possible. Suppose there exists a solution map $\overline{F} : \mathbb{R}^{N+\widetilde{N}-M} \times \mathbb{R}^M \to \mathbb{R}^{N+\widetilde{N}}$ so that

$$F(\Pi_N\mathbf{z};\widetilde{\mathbf{z}}) = \mathbf{y} \quad \text{if and only if} \quad (\Pi_N\mathbf{z},\widetilde{\mathbf{z}}) = \overline{F}(\mathbf{w},\mathbf{y}) \qquad \text{for a unique } \mathbf{w} \in \mathbb{R}^{N+\widetilde{N}-M}.$$

Then solving (4.4.10) is equivalent to solving the unconstrained problem

$$\underset{(\mathbf{v},\mathbf{w})\in\mathbb{R}^I\times\mathbb{R}^{N+\widetilde{N}-M}}{\text{minimize}} \quad (\mathbf{v},\overline{F}(\mathbf{w},\mathbf{y}))\begin{bmatrix}\Theta^{-1} & 0\\ 0 & \widetilde{\Theta}^{-1}\end{bmatrix}\begin{pmatrix}\mathbf{v}\\ \overline{F}(\mathbf{w},\mathbf{y})\end{pmatrix} + \frac{1}{\gamma^2}|\mathbf{v} - \mathbf{o}|^2, \tag{4.4.12}$$

and setting $\Pi^I\mathbf{z}^\dagger = \mathbf{v}^\dagger$ and $(\Pi_N\mathbf{z}^\dagger,\widetilde{\mathbf{z}}) = \overline{F}(\mathbf{w}^\dagger,\mathbf{y})$.

4.4.3 Implementation

Both of the problems (4.4.11) and (4.4.12) can be solved using the same techniques outlined in Subsection 4.3.4 except that we now have a higher dimensional solution space. Below we briefly describe the main differences between the implementation of the PDE and IP solvers.

4.4.3.1 Constructing Θ and $\widetilde{\Theta}$

We propose to construct the matrices $\Theta, \widetilde{\Theta}$ using appropriate kernels K, for the solution u of the PDE, and $\widetilde{K}$, for the parameter a identically to Subsection 4.3.4.1. Our minimum requirements on $K, \widetilde{K}$ is sufficient regularity for the pointwise constraints in (4.4.4) to be well-defined. Since we have limited and noisy data in the inverse problem setting, it is not possible for us to recover the exact solution $(u^\star, a^\star)$ in general and so the kernels $K, \widetilde{K}$ should be chosen to reflect our prior assumptions on the unknown parameter and the solution of the PDE at that parameter value.

4.4.4 Numerical Experiments for Darcy Flow

In this subsection, we apply our method to an IP involving the Darcy flow PDE. We consider the IP outlined in Example DF with the true coefficient $a^\star(\mathbf{x})$ satisfying

$$\exp\Big(a^\star(\mathbf{x})\Big) = \exp\Big(\sin(2\pi x_1) + \sin(2\pi x_2)\Big) + \exp\Big(-\sin(2\pi x_1) - \sin(2\pi x_2)\Big)\,,$$

and the right hand side source term is $f = 1$. We randomly sampled $M = 500$ collocation points with $M_\Omega = 400$ in the interior. From these 400 interior points,

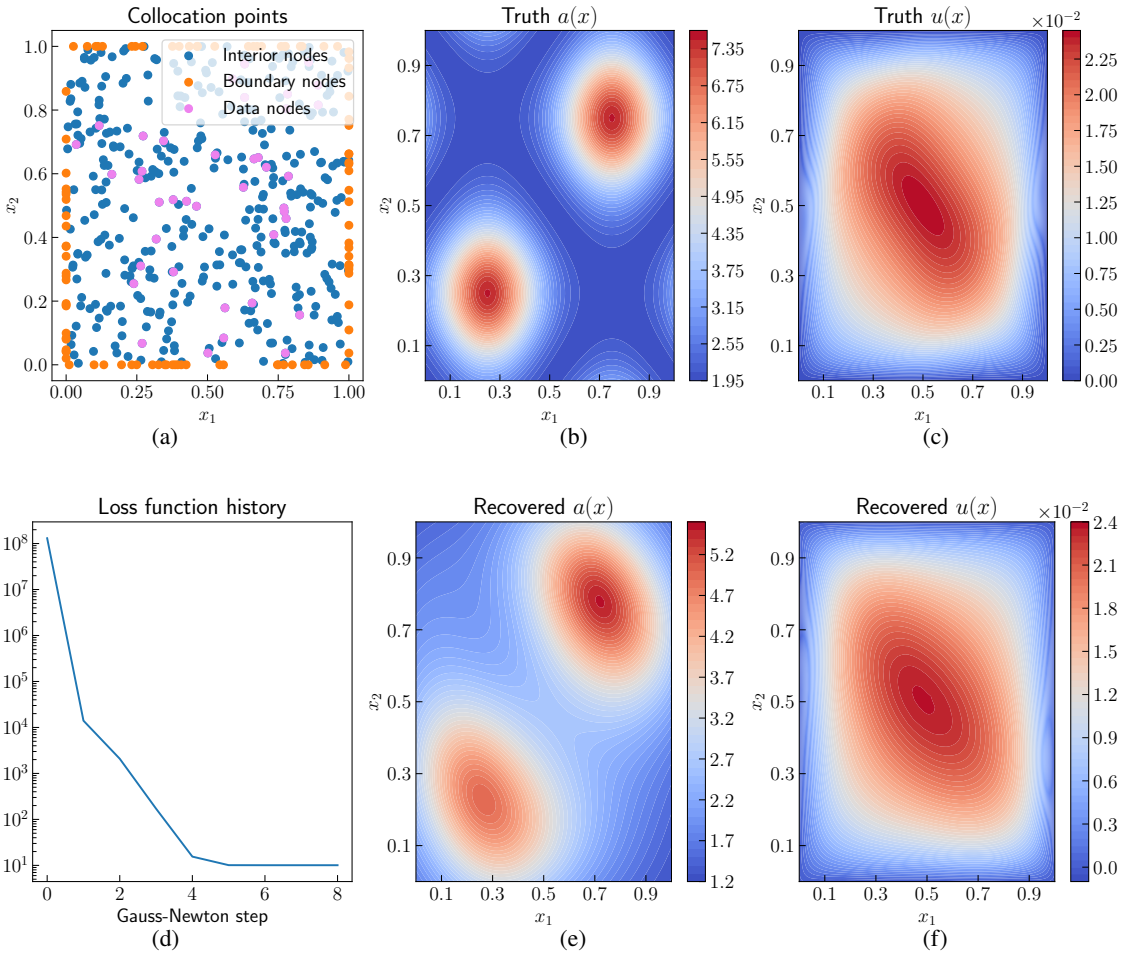


Figure 4.5: Numerical results for the inverse Darcy flow: (a) an instance of uniformly sampled collocation points and data points; (d) Gauss–Newton iteration history; (b) true a; (e) recovered a; (c) true u; (f) recovered u. Adaptive diagonal regularization ("nugget") terms were added to the kernel matrix, with parameters $\eta = \tilde{\eta} = 10^{-5}$ as outlined in Appendix A.1.5.

we randomly chose $I = 40$ points and observed the values of $u(\mathbf{x})$ at those points as the data for the IP. The values of $u(\mathbf{x})$ for this purpose were generated by first solving the equation with the true coefficient on a uniform grid and then using linear interpolation to get the solution at the observation points. We further added independent Gaussian noise $\mathcal{N}(0, \gamma^2 I)$ with noise standard deviation $\gamma = 10^{-3}$ to these observations. In dealing with the nonlinear constraint shown in Example DF, we eliminated the variable v_3 using the relation in (4.4.6).

We chose Gaussian kernels for both u and a with the same lengthscale parameter $\sigma = 0.2$ and adaptive diagonal ("nugget") terms were added to the kernel matrices,

with parameters $\eta = \tilde{\eta} = 10^{-5}$, to regularize Θ and $\widetilde{\Theta}$ as outlined in Appendix A.1.5. In Figure 4.5 we show the experimental results for recovering both a and u. From the figure, we observe that the Gauss–Newton iterations converged after 6 steps. Moreover, the recovered a and u are reasonably accurate, i.e. they capture the shape of the truth, given the limited amount of observation information available.

4.5 Conclusions

We have introduced a kernel/GP framework for solving nonlinear PDEs and IPs centered around the idea of approximating the solution of a given PDE with a MAP estimator of a GP conditioned on satisfying the PDE at a set of collocation points. Theoretically, we exhibited a nonlinear representer theorem which finite-dimensionalizes the MAP estimation problem and proved the convergence of the resulting solution towards the truth as the number of collocation points goes to infinity, under some regularity assumptions. Computationally, we demonstrated that the solution can be found by solving a finite-dimensional optimization problem with quadratic loss and nonlinear constraints. We presented two methods for dealing with the nonlinear constraints, namely the elimination and relaxation approaches. An efficient variant of the Gauss–Newton algorithm was also proposed for solving the resulting unconstrained optimization problem, where an adaptive nugget term was employed for regularization together with offline Cholesky factorizations of the underlying kernel matrices. We demonstrated that the proposed algorithm performs well in a wide range of prototypical nonlinear problems such as a nonlinear elliptic PDE, Burgers' equation, a regularized Eikonal equation, and the identification of the permeability field in Darcy flow.

Chapter 5

SPARSE CHOLESKY FACTORIZATION FOR SOLVING PDES VIA GAUSSIAN PROCESSES

In this chapter, we introduce a sparse Cholesky factorization algorithm to scale up the GP method for solving PDEs. The exposition is based on our work [50].

5.1 Introduction

Machine learning and probabilistic inference [191] have become increasingly popular due to their ability to automate the solution of computational problems. Gaussian processes (GPs) [290] are a promising approach for combining the theoretical rigor of traditional numerical algorithms with the flexible design of machine learning solvers [201, 203, 224, 54, 43]. They also have deep connections to kernel methods [247, 239], neural networks [194, 158, 134], and meshless methods [239, 301]. This chapter studies the computational efficiency of GPs in solving nonlinear PDEs, where we need to deal with dense kernel matrices with entries obtained from pointwise values and derivatives of the covariance kernel function of the GP. The methodology developed here may also be applied to other contexts where derivative information of a GP or function is available, such as in Bayes optimization [294].

5.1.1 The problem

The GP method in [43] transforms every nonlinear PDE into the following quadratic optimization problem with nonlinear constraints:

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbb{R}^N} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}\,, \end{cases} \tag{5.1.1}$$

where F and $\mathbf{y}$ encode the PDE and source/boundary data. $K(\boldsymbol{\phi},\boldsymbol{\phi}) \in \mathbb{R}^{N\times N}$ is a positive definite kernel matrix whose entries are $k(\mathbf{x}_i,\mathbf{x}_j)$ or (a linear combination of) derivatives of the kernel function such as $\Delta_{\mathbf{x}} k(\mathbf{x}_i,\mathbf{x}_j)$. Here $\mathbf{x}_i,\mathbf{x}_j \in \mathbb{R}^d$ are some sampled collocation points in space. Entries like $k(\mathbf{x}_i,\mathbf{x}_j)$ arise from Diracs measurements while entries like $\Delta_{\mathbf{x}} k(\mathbf{x}_i,\mathbf{x}_j)$ come from derivative measurements of the GP. For more details of the methodology, see Section 5.2. Computing with the dense matrix $K(\boldsymbol{\phi},\boldsymbol{\phi})$ naïvely results in $O(N^3)$ space/time complexity.

5.1.2 Contributions

This chapter presents an algorithm that runs with complexity $O(N \log^d(N/\epsilon))$ in space and $O(N \log^{2d}(N/\epsilon))$ in time, which outputs a permutation matrix P_{perm} and a sparse upper triangular matrix U with $O(N \log^d(N/\epsilon))$ nonzero entries, such that

$$\|K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} - P_{\text{perm}}^T U U^T P_{\text{perm}}\|_{\text{Fro}} \leq \epsilon\,, \tag{5.1.2}$$

where $\|\cdot\|_{\text{Fro}}$ is the Frobenius norm. We elaborate the algorithm in Section 5.3.

Our error analysis, presented in Section 5.4, requires sufficient Diracs measurements to appear in the domain. The setting of the rigorous result is for a class of kernel functions that are Green functions of differential operators such as the Matérn-like kernels, although the algorithm is generally applicable. The analysis is based on the interplay between linear algebra, Gaussian process conditioning, screening effects, and numerical homogenization, which shows the *exponential decay/near-sparsity* of the inverse Cholesky factor of the kernel matrix after permutation.

To solve (5.1.1), one efficient method is to linearize the constraint and solve a sequential quadratic programming problem. This leads to a linear system that involves a reduced kernel matrix $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k) := DF(\mathbf{z}^k) K(\boldsymbol{\phi}, \boldsymbol{\phi}) (DF(\mathbf{z}^k))^T$ at each iterate $\mathbf{z}^k$; see Section 5.5. For this reduced kernel matrix, there are insufficient Diracs measurements in the domain, and we no longer have the above theoretical guarantee for its sparse Cholesky factorization. Nevertheless, we can still apply the algorithm with a slightly different permutation and couple it with preconditioned conjugate gradient (pCG) methods to solve the linear system. Our experiments demonstrate that nearly constant steps of pCG suffice for convergence.

For many nonlinear PDEs, we observe that the above sequential quadratic programming approach converges in $O(1)$ steps. Consequently, our algorithm leads to a near-linear space/time complexity solver for general nonlinear PDEs, assuming it converges. The assumption of convergence depends on the selection of kernels and the property of the PDE, and we demonstrate it numerically in solving nonlinear elliptic, Burgers, and Monge-Ampère equations; see Section 5.6.

5.1.3 Related work

5.1.3.1 Machine learning PDEs

Machine learning methods, such as those based on neural networks (NNs) and GPs, have shown remarkable promise in automating scientific computing, for instance in solving PDEs. Recent developments in this field include operator learning using

prepared solution data [163, 29, 196, 172] and learning a single solution without any solution data [115, 222, 43, 138]. This work focuses on the latter. NNs provide an expressive function representation. Empirical success has been widely reported in the literature. However, the training of NNs often requires significant tuning and takes much longer than traditional solvers [106]. Considerable research efforts have been devoted to stabilizing and accelerating the training process [150, 282, 284, 64, 299].

GP and kernel methods are based on a more interpretable and theoretically grounded function representation rooted in the Reproducing Kernel Hilbert Space (RKHS) theory [288, 25, 204]; with hierarchical kernel learning [292, 206, 51, 60], these representations can be made expressive as well. Nevertheless, working with dense kernel matrices is common, which often limits scalability. In the case of PDE problems, these matrices may also involve partial derivatives of the kernels [43], and fast algorithms for such matrices are less developed compared to the derivative-free counterparts.

5.1.3.2 Fast solvers for kernel matrices

Approximating dense kernel matrices (denoted by Θ) is a classical problem in scientific computing and machine learning. Most existing methods focus on the case where Θ only involves the pointwise values of the kernel function. These algorithms typically rely on low-rank or sparse approximations, as well as their combination and multiscale variants. Low-rank techniques include Nyström's approximations [291, 192, 42], rank-revealing Cholesky factorizations [107], inducing points via a probabilistic view [219], and random features [221]. Sparsity-based methods include covariance tapering [92], local experts (see a review in [167]), and approaches based on precision matrices and stochastic differential equations [166, 230, 236, 235]. Combining low-rank and sparse techniques can lead to efficient structured approximation [293] and can better capture short and long-range interactions [234]. Multiscale and hierarchical ideas have also been applied to seek for a full-scale approximation of Θ with a *near-linear* complexity. They include $\mathcal{H}$ matrix [110, 112, 111] and variants [162, 9, 10, 152, 189] that rely on the low-rank structure of the off-diagonal block matrices at different scales; wavelets-based methods [28, 100] that use the sparsity of Θ in the wavelet basis; multiresolution predictive processes [139]; and Vecchia approximations [278, 140] and sparse Cholesky factorizations [241, 240] that rely on the approximately sparse correlation conditioned on carefully

ordered points.

For Θ that contains derivatives of the kernel function, several work [81, 211, 66] has utilized structured approximation to scale up the computation; no rigorous accuracy guarantee is proved. The inducing points approach [295, 187] has also been explored; however since this method only employs a low-rank approximation, the accuracy and efficiency can be limited.

5.1.3.3 Screening effects in spatial statistics

Notably, the sparse Cholesky factorization algorithm in [240], formally equivalent to Vecchia's approximation [278, 140], achieves a state-of-the-art complexity $O(N \log^d(N/\epsilon))$ in space and $O(N \log^{2d}(N/\epsilon))$ in time for a wide range of kernel functions, with a rigorous theoretical guarantee. This algorithm is designed for kernel matrices with derivative-free entries and is connected to the screening effect in spatial statistics [259, 256]. The screening effect implies that approximate conditional independence of a spatial random field is likely to occur, under suitable ordering of points. The line of work [203, 204, 241] provides quantitative exponential decay results for the conditional covariance in the setting of a coarse-to-fine ordering of data points, laying down the theoretical groundwork for [240].

A fundamental question is how the screening effect behaves when derivative information of the spatial field is incorporated, and how to utilize it to extend sparse Cholesky factorization methods to kernel matrices that contain derivatives of the kernel. The screening effect studied within this new context can be useful for numerous applications where derivative-type measurements are available.

5.2 Solving Nonlinear PDEs via GPs

In this section, we review the GP framework in [43] for solving nonlinear PDEs. We will use a prototypical nonlinear elliptic equation as our running example to demonstrate the main ideas, followed by more complete recipes for general nonlinear PDEs.

Consider the following nonlinear elliptic PDE:

$$\begin{cases} -\Delta u + \tau(u) = f & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{5.2.1}$$

where τ is a nonlinear scalar function and Ω is a bounded open domain in $\mathbb{R}^d$ with a Lipschitz boundary. We assume the equation has a strong solution in the classical

sense.

5.2.1 The GP framework

The first step is to sample M_Ω collocation points in the interior and $M_{\partial\Omega}$ on the boundary such that

$$\mathbf{x}_\Omega = \{\mathbf{x}_1, ..., \mathbf{x}_{M_\Omega}\} \subset \Omega \quad \text{and} \quad \mathbf{x}_{\partial\Omega} = \{\mathbf{x}_{M_\Omega+1}, ..., \mathbf{x}_M\} \subset \partial\Omega\,,$$

where $M = M_\Omega + M_{\partial\Omega}$. Then, by assigning a GP prior to the unknown function u with mean 0 and covariance function $K : \overline{\Omega} \times \overline{\Omega} \to \mathbb{R}$, the method aims to compute the maximum a posterior (MAP) estimator of the GP given the sampled PDE data, which leads to the following optimization problem

$$\begin{cases} \underset{u\in\mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad -\Delta u(\mathbf{x}_m) + \tau(u(\mathbf{x}_m)) = f(\mathbf{x}_m), & \text{for } m = 1, \ldots, M_\Omega\,, \\ \qquad u(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \ldots, M\,. \end{cases} \tag{5.2.2}$$

Here, $\|\cdot\|$ is the Reproducing Kernel Hilbert Space (RKHS) norm corresponding to the kernel/covariance function K.

Regarding consistency, once K is sufficiently regular, the above solution will converge to the exact solution of the PDE when $M_\Omega, M_{\partial\Omega} \to \infty$; see Theorem 1.2 in [43]. The methodology can be seen as a nonlinear generalization of many radial basis function based meshless methods [239] and probabilistic numerics [201, 54].

5.2.2 The finite dimensional problem

The next step is to transform (5.2.2) into a finite-dimensional problem for computation. We first introduce some notations:

- Notations for *measurements*: We denote the measurement functions by

  $$\phi_m^{(1)} = \delta_{\mathbf{x}_m}, 1 \le m \le M \quad \text{and} \quad \phi_m^{(2)} = \delta_{\mathbf{x}_m} \circ \Delta, 1 \le m \le M_\Omega\,,$$

  where $\delta_{\mathbf{x}}$ is the Dirac delta function centered at $\mathbf{x}$. They are in $\mathcal{U}^*$, the dual space of $\mathcal{U}$, for sufficiently regular kernel functions.

  Further, we use the shorthand notation $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$ for the M and M_Ω-dimensional vectors with entries $\phi_m^{(1)}$ and $\phi_m^{(2)}$ respectively, and $\boldsymbol{\phi}$ for the N-dimensional vector obtained by concatenating $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$, where $N = M + M_\Omega$.

- Notations for *primal dual pairing*: We use $[\cdot,\cdot]$ to denote the primal dual pairing, such that for $u \in \mathcal{U}, \phi_m^{(1)} = \delta_{\mathbf{x}_m} \in \mathcal{U}^*$, it holds that $[u, \phi_m^{(1)}] = u(\mathbf{x}_m)$. Similarly $[u, \phi_m^{(2)}] = \Delta u(\mathbf{x}_m)$ for $\phi_m^{(2)} = \delta_{\mathbf{x}_m} \circ \Delta \in \mathcal{U}^*$. For simplicity of presentation, we oftentimes abuse the notation to write the primal-dual pairing in the L^2 integral form: $[u, \phi] = \int u(\mathbf{x})\phi(\mathbf{x})\,\mathrm{d}\mathbf{x}$.

- Notations for *kernel matrices*: We write $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ as the $N \times N$-matrix with entries $\int K(\mathbf{x}, \mathbf{x}')\phi_m(\mathbf{x})\phi_j(\mathbf{x}')\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{x}'$ where ϕ_m denotes the entries of $\boldsymbol{\phi}$. Here, the integral notation shall be interpreted as the primal-dual pairing as above.

  Similarly, $K(\mathbf{x}, \boldsymbol{\phi})$ is the N dimensional vector with entries $\int K(\mathbf{x}, \mathbf{x}')\phi_j(\mathbf{x}')\,\mathrm{d}\mathbf{x}'$. Moreover, we adopt the convention that if the variable inside a function is a set, it means that this function is applied to every element in this set; the output will be a vector or a matrix. As an example, $K(\mathbf{x}_\Omega, \mathbf{x}_\Omega) \in \mathbb{R}^{M_\Omega \times M_\Omega}$.

Then, based on a generalization of the representer theorem [43], the minimizer of (5.2.2) attains the form

$$u^\dagger(\mathbf{x}) = K(\mathbf{x}, \boldsymbol{\phi}) K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} \mathbf{z}^\dagger ,$$

where $\mathbf{z}^\dagger$ is the solution to the following finite dimensional quadratic optimization problem with nonlinear constraints

$$\begin{cases} \underset{\mathbf{z} \in \mathbb{R}^{M+M_\Omega}}{\text{minimize}} & \mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} \mathbf{z} \\ \text{s.t.} & -z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), \quad \text{for } m = 1, \dots, M_\Omega , \\ & z_m^{(1)} = g(\mathbf{x}_m), \qquad\qquad \text{for } m = M_\Omega + 1, \dots, M . \end{cases} \tag{5.2.3}$$

Here, $\mathbf{z}^{(1)} \in \mathbb{R}^M$, $\mathbf{z}^{(2)} \in \mathbb{R}^{M_\Omega}$ and $\mathbf{z}$ is the concatenation of them. For this specific example, we can write down $K(\mathbf{x}, \boldsymbol{\phi})$ and $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ explicitly:

$$\begin{aligned} K(\mathbf{x}, \boldsymbol{\phi}) &= \big(K(\mathbf{x}, \mathbf{x}_\Omega), K(\mathbf{x}, \mathbf{x}_{\partial\Omega}), \Delta_{\mathbf{y}} K(\mathbf{x}, \mathbf{x}_\Omega)\big) \in \mathbb{R}^{1 \times N} , \\ K(\boldsymbol{\phi}, \boldsymbol{\phi}) &= \begin{pmatrix} K(\mathbf{x}_\Omega, \mathbf{x}_\Omega) & K(\mathbf{x}_\Omega, \mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{y}} K(\mathbf{x}_\Omega, \mathbf{x}_\Omega) \\ K(\mathbf{x}_{\partial\Omega}, \mathbf{x}_\Omega) & K(\mathbf{x}_{\partial\Omega}, \mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{y}} K(\mathbf{x}_{\partial\Omega}, \mathbf{x}_\Omega) \\ \Delta_{\mathbf{x}} K(\mathbf{x}_\Omega, \mathbf{x}_\Omega) & \Delta_{\mathbf{x}} K(\mathbf{x}_\Omega, \mathbf{x}_{\partial\Omega}) & \Delta_{\mathbf{x}}\Delta_{\mathbf{y}} K(\mathbf{x}_\Omega, \mathbf{x}_\Omega) \end{pmatrix} \in \mathbb{R}^{N \times N} . \end{aligned} \tag{5.2.4}$$

Here, $\Delta_{\mathbf{x}}, \Delta_{\mathbf{y}}$ are the Laplacian operator for the first and second arguments of k, respectively. Clearly, evaluating the loss function and its gradient requires us to deal with the dense kernel matrix $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ with entries comprising *derivatives* of k.

5.2.3 The general case

For general PDEs, the methodology leads to the optimization problem

$$\begin{cases} \min\limits_{u\in\mathcal{U}} & \|u\| \\ \text{s.t.} & \text{PDE constraints at } \{\mathbf{x}_1,\dots,\mathbf{x}_M\}\in\overline{\Omega}, \end{cases}$$

and the equivalent finite dimensional problem

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbb{R}^N} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & F(\mathbf{z})=\mathbf{y}, \end{cases} \tag{5.2.5}$$

where $\boldsymbol{\phi}$ is the concatenation of Diracs measurements and derivative measurements of u; they are induced by the PDE at the sampled points. The function F encodes the PDE, and the vector $\mathbf{y}$ encodes the right hand side and boundary data. Again, it is clear that the computational bottleneck lies in the part $K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}$.

Remark 5.2.1. *Here, we use "derivative measurement" to mean a functional in $\mathcal{U}^*$ whose action on a function in $\mathcal{U}$ leads to a linear combination of its derivatives. Mathematically, suppose the highest order of derivatives is J, then the corresponding derivative measurement at point $\mathbf{x}_m$ can be written as $\phi=\sum_{|\gamma|\le J} a_\gamma \delta_{\mathbf{x}_m}\circ D^\gamma$ with the multi-index $\gamma=(\gamma_1,\dots,\gamma_d)\in\mathbb{N}^d$ and $|\gamma|:=\sum_{k=1}^d\gamma_k\le J$. Here $D^\gamma:=D^{\gamma_1}_{\mathbf{x}^{(1)}}\cdots D^{\gamma_d}_{\mathbf{x}^{(d)}}$ is a $|\gamma|$-th order differential operator, and we use the notation $\mathbf{x}=(\mathbf{x}^{(1)},\dots,\mathbf{x}^{(d)})$. We require linear independence between these measurements to ensure $K(\boldsymbol{\phi},\boldsymbol{\phi})$ is invertible.* ◊

5.3 The Sparse Cholesky Factorization Algorithm

In this section, we present a sparse Cholesky factorization algorithm for $K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}$. Theoretical results will be presented in Section 5.4 based on the interplay between linear algebra, Gaussian process conditioning, screening effects in spatial statistics, and numerical homogenization.

In Subsection 5.3.1, we summarize the state-of-the-art sparse Cholesky factorization algorithm for kernel matrices with derivative-free measurements. In Subsection 5.3.2, we discuss an extension of the idea to kernel matrices with derivative-type measurements, which are the main focus of this chapter. The algorithm presented in Subsection 5.3.2 leads to near-linear complexity evaluation of the loss function and its gradient in the GP method for solving PDEs. First-order methods thus become scalable. We then extend the algorithm to second-order optimization methods (e.g., the Gauss-Newton method) in Section 5.5.

5.3.1 The case of derivative-free measurements

We start the discussion with the case where $\boldsymbol{\phi}$ contains Diracs-type measurements only.

Consider a set of points $\{\mathbf{x}_i\}_{i\in I} \subset \Omega$, where $I = \{1, 2, ..., M\}$ as in Subsection 5.2.1. We assume the points are *scattered*; to quantify this, we have the following definition of homogeneity:

Definition 5.3.1. *The homogeneity parameter of the points* $\{\mathbf{x}_i\}_{i\in I} \subset \Omega$ *conditioned on a set* A *is defined as*

$$\delta(\{\mathbf{x}_i\}_{i\in I}; \mathsf{A}) = \frac{\min_{\mathbf{x}_i,\mathbf{x}_j\in I} \operatorname{dist}(\mathbf{x}_i, \{\mathbf{x}_j\} \cup \mathsf{A})}{\max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, \{\mathbf{x}_i\}_{i\in I} \cup \mathsf{A})}.$$

When $\mathsf{A} = \emptyset$*, we also write* $\delta(\{\mathbf{x}_i\}_{i\in I}) := \delta(\{\mathbf{x}_i\}_{i\in I}; \emptyset)$.

Throughout this chapter, we assume $\delta(\{\mathbf{x}_i\}_{i\in I}) > 0$. One can understand that a larger $\delta(\{\mathbf{x}_i\}_{i\in I})$ makes the distribution of points more homogeneous in space. It can also be useful to consider $\mathsf{A} = \partial\Omega$ if one wants the points not too close to the boundary.

Let $\boldsymbol{\phi}$ be the collection of $\delta_{\mathbf{x}_i}, 1 \le i \le M$; all of them are Diracs-type measurements and thus derivative-free. In [240], a sparse Choleksy factorization algorithm was proposed to factorize $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}$. We summarize this algorithm (with a slight modification[1]) in the following three steps: reordering, sparsity pattern, and Kullback-Leibler (KL) minimization.

5.3.1.1 Reordering

The first step is to reorder these points from *coarse to fine* scales. It can be achieved by the maximum-minimum distance ordering (maximin ordering) [108]. We define a generalization to conditioned maximin ordering as follows:

Definition 5.3.2 (Conditioned Maximin Ordering)**.** *The maximin ordering conditioned on a set* A *for points* $\{\mathbf{x}_i, i \in I\}$ *is obtained by successively selecting the point* $\mathbf{x}_i$ *that is furthest away from* A *and the already picked points. If* A *is an empty set, then we select an arbitrary index* $i \in I$ *as the first to start. Otherwise, we choose the first index as*

$$i_1 = \arg\max_{i\in I} \operatorname{dist}(\mathbf{x}_i, \mathsf{A}).$$

[1] The method in [240] was presented to get the lower triangular Cholesky factors. We present the method for solving the upper triangular Cholesky factors since it gives a more concise description. As a consequence of this difference, in the reordering step, we are led to a reversed ordering compared to that in [240].

For the first q indices already chosen, we choose

$$i_{q+1} = \arg\max_{i \in I \setminus \{i_1,\ldots,i_q\}} \operatorname{dist}(\mathbf{x}_i, \{\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_q}\} \cup \mathsf{A}) .$$

Usually we set $\mathsf{A} = \partial\Omega$ or $\emptyset$. We introduce the operator $P : I \to I$ to map the order of the measurements to the index of the corresponding points, i.e., $P(q) = i_q$. One can define the *lengthscale* of each ordered point as

$$l_i = \operatorname{dist}(\mathbf{x}_{P(i)}, \{\mathbf{x}_{P(1)}, \ldots, \mathbf{x}_{P(i-1)}\} \cup \mathsf{A}) . \tag{5.3.1}$$

Let $\Theta = K(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\phi}}) \in \mathbb{R}^{N \times N}$ be the kernel matrix after reordering the measurements in $\boldsymbol{\phi}$ to $\tilde{\boldsymbol{\phi}} = (\phi_{P(1)}, \ldots, \phi_{P(M)})$; we have $N = M$ in this setting. An important observation is that the Cholesky factors of Θ and Θ^{-1} could exhibit *near-sparsity* under the maximin ordering. Indeed, as an example, suppose $\Theta^{-1} = U^\star U^{\star T}$ where $U^\star$ is the upper Cholesky factor. Then in Figure 5.1, we show the magnitude of $U^\star_{ij}, i \le j$ for a Matérn kernel, where $j = 1000, 2000$; the total number of points is $M = 51^2$. It is clear from the figure that the entries decay very fast when the points move far away from the current jth ordered point.

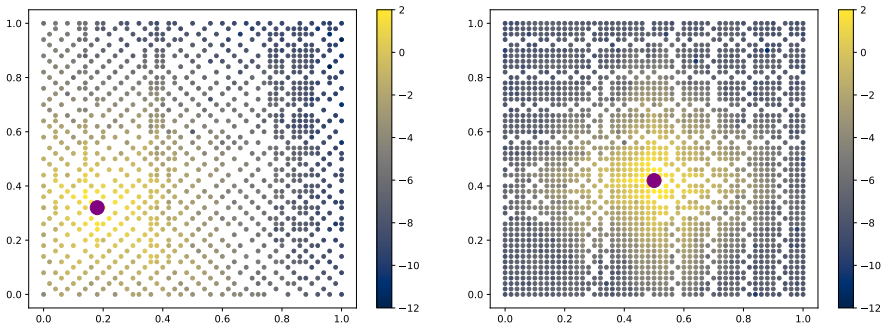


Figure 5.1: Demonstration of screening effects in the context of Diracs measurements using the Matérn kernel with $\nu = 5/2$ and lengthscale 0.3. The data points are equidistributed in $[0,1]^2$ with grid size $h = 0.02$. In the left figure, we display the 1000th point (the big point) in the maximin ordering with $\mathsf{A} = \emptyset$, where all points ordered before it (i.e., $i < 1000$) are colored with intensity according to the corresponding $|U^\star_{ij}|$. The right figure is generated in the same manner but for the 2000th point in the ordering.

Remark 5.3.3. *One may wonder why such a coarse-to-fine reordering leads to sparse Cholesky factors. In fact, we can interpret entries of $U^\star$ as the conditional covariance of some GP. More precisely, consider the GP $\xi \sim \mathcal{GP}(0, K)$. Then, by*

definition, the Gaussian random variables $Y_i := [\xi, \tilde{\phi}_i] \sim \mathcal{N}(0, K(\tilde{\phi}_i, \tilde{\phi}_i))$. *We have the following relation:*

$$\frac{U^\star_{ij}}{U^\star_{jj}} = (-1)^{i\neq j} \frac{\mathrm{Cov}[Y_i, Y_j | Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i | Y_{1:j-1\setminus\{i\}}]}, \quad i \leq j\,. \tag{5.3.2}$$

Here we used the MATLAB notation such that $Y_{1:j-1\setminus\{i\}}$ *corresponds to* $\{Y_q : 1 \leq q \leq j-1, q \neq i\}$. *Proof of this formula can be found in Appendix B.4.3.*

Formula (5.3.2) *links the values of* $U^\star$ *to the conditional covariance of a GP. In spatial statistics, it is well-known from empirical evidence that conditioning a GP on coarse-scale measurements results in very small covariance values between finer-scale measurements. This phenomenon, known as screening effects, has been discussed in works such as [259, 256]. The implication is that conditioning on coarse scales screens out fine-scale interactions.*

As a result, one would expect the corresponding Cholesky factor to become sparse upon reordering. Indeed, the off-diagonal entries exhibit exponential decay. A rigorous proof of the quantitative decay can be found in [241], where the measurements consist of Diracs functionals only, and the kernel function is the Green function of some differential operator subject to Dirichlet boundary conditions. The proof of Theorem 6.1 in [241] effectively implies that

$$|U^\star_{ij}| \leq C_1 l_M^{C_2} \exp\left(-\frac{\mathrm{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C_1 l_j}\right) \tag{5.3.3}$$

for some generic constants C_1, C_2 *depending on the domain, kernel function, and homogeneity parameter of the points. We will prove such decay also holds when derivative-type measurements are included, in Section 5.4 under a novel ordering. It is worth mentioning that our analysis also provides a much simpler proof for* (5.3.3). ◊

5.3.1.2 Sparsity pattern

With the ordering determined, our next step is to identify the sparsity pattern of the Cholesky factor under the ordering.

For a tuning parameter $\rho \in \mathbb{R}^+$, we select the upper-triangular sparsity set $S_{P,l,\rho} \subset I \times I$ as

$$S_{P,l,\rho} = \{(i,j) \subset I \times I : i \leq j,\, \mathrm{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)}) \leq \rho l_j\}\,. \tag{5.3.4}$$

The choice of the sparsity pattern is motivated by the quantitative exponential decay result mentioned in Remark 5.3.3. Here in the subscript, P stands for the ordering, l

is the lengthscale parameter associated with the ordering, and ρ is a hyperparameter that controls the size of the sparsity pattern. We sometimes drop the subscript to simplify the notation when there is no confusion. We note that the cardinality of the set $S_{P,l,\rho}$ is bounded by $O(N\rho^d)$, through a ball-packing argument (see Appendix B.2).

Remark 5.3.4. *The maximin ordering and the sparsity pattern can be constructed with computational complexity $O(N\log^2(N)\rho^d)$ in time and $O(N\rho^d)$ in space; see Algorithm 4.1 and Theorem 4.1 in [240].* ◊

5.3.1.3 KL minimization

With the ordering and sparsity pattern identified, the last step is to use KL minimization to compute the best approximate sparse Cholesky factors given the pattern.

Define the set of sparse upper-triangular matrices with sparsity pattern $S_{P,l,\rho}$ as

$$\mathcal{S}_{P,l,\rho} := \{A \in \mathbb{R}^{N\times N} : A_{ij} \neq 0 \Rightarrow (i,j) \in S_{P,l,\rho}\}\,. \tag{5.3.5}$$

For each column j, denote $s_j = \{i : (i,j) \in S_{P,l,\rho}\}$. The cardinality of the set s_j is denoted by $\#s_j$.

The KL minimization step seeks to find

$$U = \arg\min_{\hat{U}\in\mathcal{S}_{P,l,\rho}} \mathrm{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(\hat{U}\hat{U}^T)^{-1})\right)\,. \tag{5.3.6}$$

It turns out that the above problem has an explicit solution

$$U_{s_j,j} = \frac{\Theta^{-1}_{s_j,s_j}\mathbf{e}_{\#s_j}}{\sqrt{\mathbf{e}^T_{\#s_j}\Theta^{-1}_{s_j,s_j}\mathbf{e}_{\#s_j}}}\,, \tag{5.3.7}$$

where $\mathbf{e}_{\#s_j}$ is a standard basis vector in $\mathbb{R}^{\#s_j}$ with the last entry being 1 and other entries equal 0. Here, $\Theta^{-1}_{s_j,s_j} := (\Theta_{s_j,s_j})^{-1}$. The proof of this explicit formula follows a similar approach to that of Theorem 2.1 in [240], with the only difference being the use of upper Cholesky factors. A detailed proof is provided in Appendix B.3. It is worth noting that the optimal solution is equivalent to the Vecchia approximation used in spatial statistics; see discussions in [240].

With the KL minimization, we can find the best approximation measured in the KL divergence sense, given the sparsity pattern. The computation is embarrassingly parallel, noting that the formula (5.3.7) are independent for different columns.

Remark 5.3.5. *For the algorithm described above, the computational complexity is upper-bounded by* $O(\sum_{1\leq i\leq N}(\#s_j)^3)$ *in time and* $O(\#S + \max_{1\leq i\leq N}(\#s_j)^2)$ *in space when using dense Cholesky factorization to invert* Θ_{s_j,s_j}.

When using the sparsity pattern $S_{P,l,\rho}$*, we can obtain* $\#s_j = O(\rho^d)$ *and* $\#S = O(N\rho^d)$ *via a ball-packing argument (see Appendix B.2). This yields a complexity of* $O(N\rho^{3d})$ *in time and* $O(N\rho^d)$ *in space.* ◊

Remark 5.3.6. *The concept of supernodes [240], which relies on an extra parameter* λ*, can be utilized to group the sparsity pattern of nearby measurements and create an aggregate sparsity pattern* $S_{P,l,\rho,\lambda}$*. This technique reduces computation redundancy and improves the arithmetic complexity of the KL minimization to* $O(N\rho^{2d})$ *in time (see Appendix B.1). In this chapter, we consistently employ this approach.* ◊

Remark 5.3.7. *In [240], it was shown in Theorem 3.4 that* $\rho = O(\log(N/\epsilon))$ *suffices to get an* ϵ*-approximate factor for a large class of kernel matrices, so the complexity of the KL minimization is* $O(N\log^{2d}(N/\epsilon))$ *in time and* $O(N\log^d(N/\epsilon))$ *in space. Note that the ordering and aggregate sparsity pattern can be constructed in time complexity* $O(N\log^2(N)\rho^d)$ *and space complexity* $O(N\rho^d)$*; the complexity of this construction step is usually of a lower order compared to that of the KL minimization. Moreover, this step can be pre-computed.* ◊

5.3.2 The case of derivative measurements

The last subsection discusses the sparse Cholesky factorization algorithm for kernel matrices that are generated by derivative-free measurements. When using GPs to solve PDEs and inverse problems, $\boldsymbol{\phi}$ can contain derivative measurements, which are the main focus of this chapter. This subsection aims to deal with such scenarios.

5.3.2.1 The nonlinear elliptic PDE case

To begin with, we will consider the example in Subsection 5.2.2, where we have Diracs measurements $\phi_m^{(1)} = \delta_{\mathbf{x}_m}$ for $1 \leq m \leq M$, and Laplacian-type measurements $\phi_m^{(2)} = \delta_{\mathbf{x}_m} \circ \Delta$ for $1 \leq m \leq M_\Omega$. Our objective is to extend the algorithm discussed in the previous subsection to include these derivative measurements.

An important question we must address is the ordering of these measurements. Specifically, should we consider the Diracs measurements before or after the derivative-type measurements? To explore this question, we conduct the following experiment.

First, we order all derivative-type measurements, $\phi_m^{(2)}$ for $1 \le m \le M_\Omega$, in an arbitrary manner. We then follow this ordering with any Diracs measurement, labeled $M_\Omega + 1$ in the order. For this measurement, we plot the magnitude of the corresponding Cholesky factor of Θ^{-1}, i.e., $|U^\star_{ij}|$ for $i \le j$ and $j = M_\Omega + 1$, similar to the approach taken in Figure 5.1. The results are shown in the left part of Figure 5.2.

Unfortunately, we do not observe an evident decay in the left of Figure 5.2. This may be due to the fact that, even when conditioned on the Laplacian-type measurements, the Diracs measurements can still exhibit long-range interactions with other measurements. This is because there are degrees of freedom of harmonic functions that are not captured by Laplacian-type measurements, and thus, the correlations may not be effectively screened out.

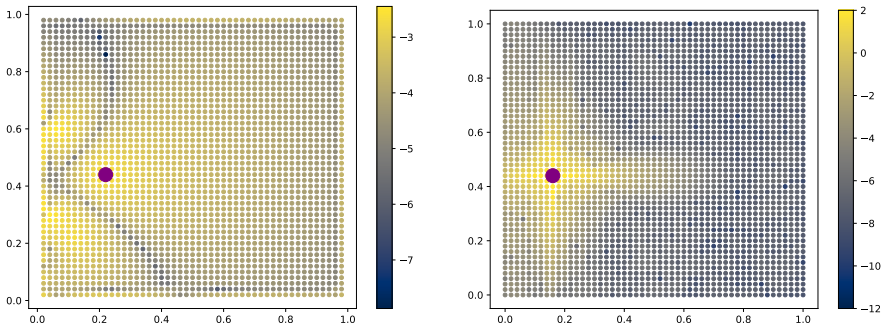


Figure 5.2: Demonstration of screening effects in the context of derivative-type measurements using the Matérn kernel with $\nu = 5/2$ and lengthscale 0.3. The data points are equidistributed in $[0,1]^2$ with grid size $h = 0.02$. In the left figure, we order the Laplacian measurements first and then select a Diracs measurement which is the big point. The points are colored with intensity according to $|U^\star_{ij}|$. In the right figure, we order the Dircas measurements first and then select a Laplacian measurement; we display things in the same manner as the left figure.

Alternatively, we can order the Dirac measurements first and then examine the same quantity as described above for any Laplacian measurement. This approach yields the right part of Figure 5.2, where we observe a fast decay as desired. This indicates that the derivative measurements should come after the Dirac measurements, or equivalently, that the derivative measurements should be treated as *finer scales* compared to the pointwise measurements.

With the above observation, we can design our new ordering as follows. For the nonlinear elliptic PDE example in Subsection 5.2.2, we order the Diracs measurements $\phi_m^{(1)} = \delta_{\mathbf{x}_m}, 1 \le m \le M$ first using the maximin ordering with $\mathsf{A} = \emptyset$ mentioned

earlier. Then, we add the derivative-type measurements $\delta_{\mathbf{x}_m} \circ \Delta, 1 \leq m \leq M_\Omega$ in arbitrary order to the ordering.

Again, for our notations, we use $P : I_N \to I$ to map the index of the ordered measurements to the index of the corresponding points. Here $I_N := \{1, 2, ..., N\}$, $N = M + M_\Omega$ and the cardinality of I is M. We define the lengthscales of the ordered measurements to be

$$l_i = \begin{cases} \text{dist}(\mathbf{x}_{P(i)}, \{\mathbf{x}_{P(1)}, ..., \mathbf{x}_{P(i-1)}\} \cup \mathsf{A}), & \text{if } i \leq M, \\ l_M, & \text{otherwise.} \end{cases} \tag{5.3.8}$$

We will justify the above choice of lengthscales in our theoretical study in Section 5.4.

With the ordering and the lengthscales determined, we can apply the same steps in the last subsection to identify sparsity patterns:

$$S_{P,l,\rho} = \{(i, j) \subset I_N \times I_N : i \leq j, \text{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)}) \leq \rho l_j\} \subset I_N \times I_N \,. \tag{5.3.9}$$

Then, we can use KL minimization as in Subsection 5.3.1.3 (see (5.3.5), (5.3.6), and (5.3.7)) to find the optimal sparse factors under the pattern. This leads to our sparse Cholesky factorization algorithm for kernel matrices with derivative-type measurements.

Remark 5.3.8. *Similar to Remark 5.3.6, the above KL minimization step (with the idea of supernodes to aggregate the sparsity pattern) can be implemented in time complexity $O(N\rho^{2d})$ and space complexity $O(N\rho^d)$, for a parameter $\rho \in \mathbb{R}^+$ that determines the size of the sparsity set.* ◊

We present some numerical experiments to demonstrate the accuracy of such an algorithm. In Figure 5.3, we show the error measured in the KL divergence sense, namely $\text{KL}\left(\mathcal{N}(0, \Theta) \parallel \mathcal{N}(0, (U^\rho U^{\rho T})^{-1})\right)$ where U^ρ is the computed sparse factor. The figures show that the KL error decays exponentially fast regarding ρ. The rate is faster for less smooth kernels, and for the same kernel, the rate remains the same when there are more physical points. In the left of Figure 5.4, we show the CPU time of the algorithm, which scales nearly linearly regarding the number of points.

5.3.2.2 General case

We present the algorithm discussed in the last subsection for general PDEs. In the general case (5.2.5), we need to deal with $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ where $\boldsymbol{\phi}$ is the concatenation

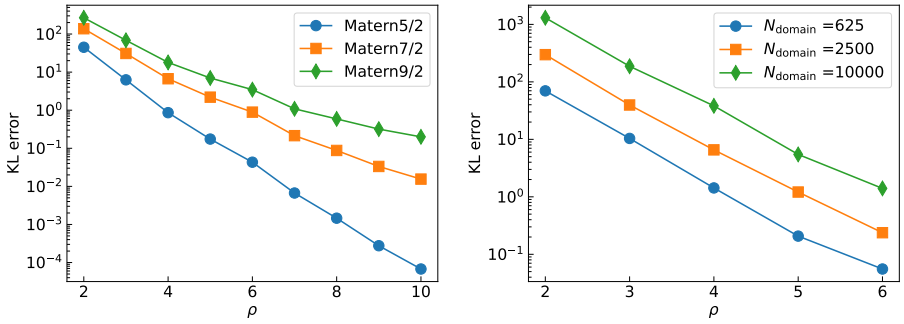


Figure 5.3: Demonstration of the accuracy of the sparse Cholesky factorization for $K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}$ in the nonlinear elliptic PDE example. In the left figure, we choose Matérn kernels with $\nu = 5/2, 7/2, 9/2$ and lengthscale $l = 0.3$; the physical points are fixed to be equidistributed in $[0,1]^2$ with grid size $h = 0.05$; we plot the error measured in the KL sense with regard to different ρ. In the right figure, we fix the Matérn kernels with $\nu = 5/2$ and lengthscale $l = 0.3$. We vary the number of physical points, which are equidistributed with grid size $h = 0.04, 0.02, 0.01$; thus $N_{\text{domain}} = 625, 2500, 10000$ correspondingly.

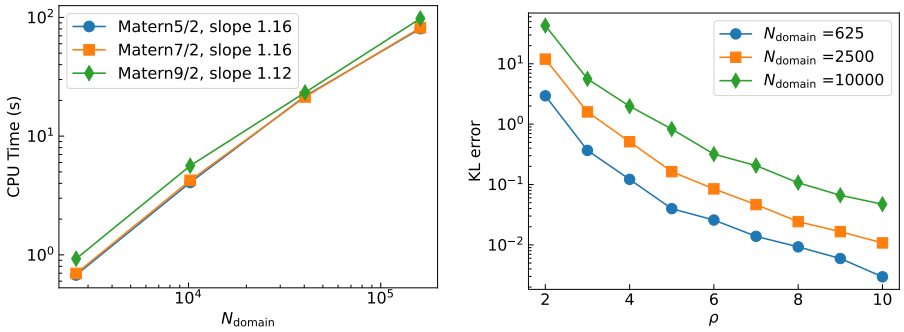


Figure 5.4: In the left figure, we choose Matérn kernels with $\nu = 5/2, 7/2, 9/2$ and lengthscale $l = 0.3$; the physical points are equidistributed in $[0,1]^2$ with different grid sizes; we plot the CPU time of the factorization algorithm for $K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}$, using the personal computer MacBook Pro 2.4 GHz Quad-Core Intel Core i5. In the right figure, we study the sparse Cholesky factorization for the reduced kernel matrix $K(\boldsymbol{\phi}^k,\boldsymbol{\phi}^k)^{-1}$. We fix the Matérn kernels with $\nu = 5/2$ and lengthscale $l = 0.3$. We vary the number of physical points, which are equidistributed with grid size $h = 0.04, 0.02, 0.01$. We plot the KL error with regard to different ρ.

of Diracs measurements and derivative-type measurements that are derived from the PDE. Suppose the number of physical points is M; they are $\{\mathbf{x}_1, ..., \mathbf{x}_M\}$ and the index set is denoted by $I = \{1, ..., M\}$. Without loss of generality, we can assume $\boldsymbol{\phi}$ contains Diracs measurements $\delta_{\mathbf{x}_m}$ at all these points and some derivative measurements at these points up to order $J \in \mathbb{N}$. See the definition of derivative

measurements in Remark 5.2.1. The reason we can assume Diracs measurements are in $\boldsymbol{\phi}$ is that one can always add $0u(\mathbf{x})$ to the PDE if there are no terms involving $u(\mathbf{x})$. The presence of these Diracs measurements is the key to get provable guarantee of the algorithm; for details see Section 5.4.

Denote the total number of measurements by N, as before. We order the Diracs measurements $\delta_{\mathbf{x}_m}, 1 \leq m \leq M$ first using the maximin ordering with $\mathsf{A} = \emptyset$. Then, we add the derivative-type measurements in an arbitrary order to the ordering.

Similar to the last subsection, we use the notation $P : I_N \to I$ to map the index of the ordered measurements to the index of the corresponding points; here $I_N := \{1, 2, ..., N\}$. The lengthscales of the ordered measurements are defined via (5.3.8). With the ordering, one can identify the sparsity pattern as in (5.3.9) (and aggregate it using supernodes as discussed in Remark 5.3.6) and use the KL minimization (5.3.5)(5.3.6)(5.3.7) to compute ϵ-approximate factors the same way as before. We outline the general algorithmic procedure in Algorithm 1.

Algorithm 1 Sparse Cholesky factorization for $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}$

1: **Input**: Measurements $\boldsymbol{\phi}$, kernel function K, sparsity parameter ρ, supernodes parameter λ
2: **Output**: U^ρ, P_{perm} s.t. $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} \approx P_{\text{perm}}^T U^\rho U^{\rho T} P_{\text{perm}}$
3: Reordering and sparsity pattern: we first order the Diracs measurements using the maximin ordering. Next, we order the derivative measurements in an arbitrary order. This process yields a permutation matrix denoted by P_{perm}, such that $P_{\text{perm}}\boldsymbol{\phi} = \tilde{\boldsymbol{\phi}}$, and lengthscales l for each measurement in $\tilde{\boldsymbol{\phi}}$. Under the ordering, we construct the aggregate sparsity pattern $S_{P,l,\rho,\lambda}$ based on the chosen values of ρ and λ.
4: KL minimization: solve (5.3.6) with $\Theta = K(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\phi}})$, by (5.3.7), to obtain U^ρ
5: **return** U^ρ, P_{perm}

The complexity is of the same order as in Remark 5.3.8; the hidden constant in the complexity estimate depends on J, the maximum order of the derivative measurements. We will present theoretical analysis for the approximation accuracy in Section 5.4, which implies that $\rho = O(\log(N/\epsilon))$ suffices to provide ϵ-approximation for a large class of kernel matrices.

5.4 Theoretical Study

In this section, we perform a theoretical study of the sparse Cholesky factorization algorithm in Subsection 5.3.2 for $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}$.

5.4.1 Set-up for rigorous results

We present the setting of kernels, physical points, and measurements for which we will provide rigorous analysis of the algorithm.

Kernel

We first describe the domains and the function spaces. Suppose Ω is a bounded convex domain in $\mathbb{R}^d$ with a Lipschitz boundary. Without loss of generality, we assume $\operatorname{diam}(\Omega) \leq 1$; otherwise, we can scale the domain. Let $H_0^s(\Omega)$ be the Sobolev space in Ω with order $s \in \mathbb{N}$ derivatives in L^2 and zero traces. Let the operator

$$\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$$

satisfy Assumption 5.4.1. This assumption is the same as in Section 2.2 of [204].

Assumption 5.4.1. *The following conditions hold for $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$:*

(i) symmetry: $[u, \mathcal{L}v] = [v, \mathcal{L}u]$*;*

(ii) positive definiteness: $[u, \mathcal{L}u] > 0$ *for* $\|u\|_{H_0^s(\Omega)} > 0$*;*

(iii) boundedness:

$$\|\mathcal{L}\| := \sup_u \frac{\|\mathcal{L}u\|_{H^{-s}(\Omega)}}{\|u\|_{H_0^s(\Omega)}} < \infty, \quad \|\mathcal{L}^{-1}\| := \sup_u \frac{\|\mathcal{L}^{-1}u\|_{H_0^s(\Omega)}}{\|u\|_{H^{-s}(\Omega)}} < \infty\,;$$

(iv) locality: $[u, \mathcal{L}v] = 0$ *if u and v have disjoint supports.*

We assume $s > d/2$ so Sobolev's embedding theorem shows that $H_0^s(\Omega) \subset C(\Omega)$, and thus $\delta_{\mathbf{x}} \in H^{-s}(\Omega)$ for $\mathbf{x} \in \Omega$. We consider the kernel function to be the Green function $K(\mathbf{x}, \mathbf{y}) := [\delta_{\mathbf{x}}, \mathcal{L}^{-1}\delta_{\mathbf{y}}]$. An example of $\mathcal{L}$ could be $(-\Delta)^s$; we use the zero Dirichlet boundary condition to define $\mathcal{L}^{-1}$, which leads to a Matérn-like kernel.

Physical points

Consider a scattered set of points $\{\mathbf{x}_i\}_{i\in I} \subset \Omega$, where $I = \{1, 2, ..., M\}$ as in Subsection 5.3.1; the homogeneity parameter of these points is assumed to be positive:

$$\delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega) = \frac{\min_{\mathbf{x}_i, \mathbf{x}_j \in I} \operatorname{dist}(\mathbf{x}_i, \{\mathbf{x}_j\} \cup \partial\Omega)}{\max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, \{\mathbf{x}_i\}_{i\in I} \cup \partial\Omega)} > 0\,.$$

This condition ensures that the points are scattered homogeneously. Here we set $\mathsf{A} = \partial\Omega$ since we consider zero Dirichlet's boundary condition and no points will be on the boundary. The accuracy in our theory will depend on $\delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega)$.

Measurements

The setting is the same as in Subsection 5.3.2.2. We assume $\boldsymbol{\phi}$ contains Diracs measurements at *all* of the scattered points, and it also contains derivative-type measurements at some of these points up to order $J \in \mathbb{N}$. We require $J < s - d/2$ so that the Sobolev embedding theorem guarantees these derivative measurements are well-defined.

For simplicity of analysis, we assume all the measurements are of the type $\delta_{\mathbf{x}_i} \circ D^\gamma$ with the multi-index $\gamma = (\gamma_1, ..., \gamma_d) \in \mathbb{N}^d$ and $|\gamma| := \sum_{k=1}^d \gamma_k \leq J$; here $D^\gamma = D^{\gamma_1}_{\mathbf{x}^{(1)}} \cdots D^{\gamma_d}_{\mathbf{x}^{(d)}}$; see Remark 5.2.1. Note that when $|\gamma| = 0$, $\delta_{\mathbf{x}_i} \circ D^\gamma$ corresponds to Diracs measurements. The total number of measurements is denoted by N.

Note that the aforementioned assumption does not apply to the scenario of Laplacian measurements in the case of a nonlinear elliptic PDE example. This exclusion is solely for the purpose of proof convenience, as it necessitates linear independence of measurements. However, similar proofs can be applied to Laplacian measurements once linear independence between the measurements is ensured.

5.4.2 Theory

Under the setting in Subsection 5.4.1, we consider the ordering $P : \{1, 2, ..., N\} \to \{1, 2, ..., M\}$ described in Subsection 5.3.2.2. Recall that for this P, we first order the Diracs measurements using the maximin ordering conditioned on $\partial\Omega$ (since there are no boundary points); then, we follow the ordering with an arbitrary order of the derivative measurements. The lengthscale parameters are defined via

$$l_i = \begin{cases} \operatorname{dist}(\mathbf{x}_{P(i)}, \{\mathbf{x}_{P(1)}, ..., \mathbf{x}_{P(i-1)}\} \cup \partial\Omega), & \text{if } i \leq M, \\ l_M, & \text{otherwise.} \end{cases}$$

We write $\Theta = K(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\phi}}) \in \mathbb{R}^{N \times N}$, which is the kernel matrix after reordering the measurements in $\boldsymbol{\phi}$ to $\tilde{\boldsymbol{\phi}} = (\phi_{P(1)}, ..., \phi_{P(N)})$.

Theorem 5.4.2. *Under the setting in Subsection 5.4.1 and the above given ordering P, we consider the upper triangular Cholesky factorization $\Theta^{-1} = U^\star U^{\star T}$. Then, for $1 \leq i \leq j \leq N$, we have*

$$\left| \frac{U^\star_{ij}}{U^\star_{jj}} \right| \leq C l_j^{-2s} \exp\left(-\frac{\operatorname{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C l_j} \right) \quad \text{and} \quad |U^\star_{jj}| \leq C l_M^{-s+d/2},$$

where C is a generic constant that depends on $\Omega, \delta(\{\mathbf{x}_i\}_{i \in I}; \partial\Omega), d, s, J, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$.

The proof for Theorem 5.4.2 can be found in Appendix B.4.1. The proof relies on the interplay between GP regression, linear algebra, and numerical homogenization. Specifically, we use (5.3.2) to represent the ratio $U^\star_{ij}/U^\star_{jj}$ as the normalized conditional covariance of a GP. Our technical innovation is to connect this normalized term to the conditional expectation of the GP, leading to the identity

$$\left|\frac{U^\star_{ij}}{U^\star_{jj}}\right| = \left|\frac{\mathrm{Cov}[Y_i, Y_j|Y_{1:j-1\setminus i}]}{\mathrm{Var}[Y_i|Y_{1:j-1\setminus i}]}\right| = \left|\mathbb{E}[Y_j|Y_i = 1, Y_{1:j-1\setminus i} = 0]\right|,$$

where $Y \sim \mathcal{N}(0,\Theta)$. This conditional expectation directly connects to the operator-valued wavelets, or *Gamblets*, in the numerical homogenization literature [203, 204]. We can apply PDE tools to establish the decay result of Gamblets. Remarkably, the connection to the conditional expectation simplifies the analysis for general measurements, compared to the more lengthy proof based on exponential decay matrix algebra in [241] for Diracs measurements only.

In our setting, we need additional analytic results regarding the derivative measurements to prove the exponential decay of the Gamblets, which is one of the technical contributions of this work. Finally, for $|U^\star_{jj}|$, we obtain the estimate by bounding the lower and upper eigenvalues of Θ. For details, see Appendix B.4.1 and B.5.1.

With Theorem 5.4.2, we can establish that $|U^\star_{ij}|$ is exponentially small when (i,j) is outside the sparsity set $S_{P,l,\rho}$. This property enables us to show that the sparse Cholesky factorization algorithm leads to provably accurate sparse factors when $\rho = O(\log(N/\epsilon))$. See Theorem 5.4.3 for details.

Theorem 5.4.3. *Under the setting in Theorem 5.4.2, suppose U^ρ is obtained by the KL minimization* (5.3.6) *with the sparsity parameter $\rho \in \mathbb{R}^+$. Then, there exists a constant depending on $\Omega, \delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega), d, s, J, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$, such that if $\rho \geq C\log(N/\epsilon)$, we have*

$$\mathrm{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(U^\rho U^{\rho T})^{-1})\right) + \|\Theta^{-1} - U^\rho U^{\rho T}\|_{\mathrm{Fro}} + \|\Theta - (U^\rho U^{\rho T})^{-1}\|_{\mathrm{Fro}} \leq \epsilon,$$

where $\epsilon < 1$ and $\|\cdot\|_{\mathrm{Fro}}$ is the Frobenius norm.

The proof can be found in Appendix B.4.2. It is based on the KL optimality of U^ρ and a comparison inequality between KL divergence and Frobenious norm shown in Lemma B.8 of [240].

Remark 5.4.4. *Theorem 5.4.3 will still hold when the idea of supernodes in Remark 5.3.6 is used since it only makes the sparsity pattern larger.* ◊

5.5 Second Order Optimization Methods

Using the algorithm in Subsection 5.3.2, we get a sparse Cholesky factorization for $K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}$, and thus we have a fast evaluation of the loss function in (5.2.3) (and more generally in (5.2.5)) and its gradient. Therefore, first-order methods can be implemented efficiently.

In [43], a second-order Gauss-Newton algorithm is used to solve the optimization problem and is observed to converge very fast, typically in 3 to 8 iterations. In this subsection, we discuss how to make such a second-order method scalable based on the sparse Cholesky idea. As before, we first illustrate our ideas on the nonlinear elliptic PDE example (5.2.2) and then describe the general algorithm.

5.5.1 Gauss-Newton iterations

For the nonlinear elliptic PDE example, the optimization problem we need to solve is (5.2.3). Using the equation $z_m^{(2)} = \tau(z_m^{(1)}) - f(\mathbf{x}_m)$ and the boundary data, we can eliminate $\mathbf{z}^{(2)}$ and rewrite (5.2.3) as an unconstrained problem:

$$\underset{\mathbf{z}_\Omega^{(1)}\in\mathbb{R}^{M_\Omega}}{\text{minimize}} \ \left(\mathbf{z}_\Omega^{(1)}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)})\right) K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1} \begin{pmatrix} \mathbf{z}_\Omega^{(1)} \\ g(\mathbf{x}_{\partial\Omega}) \\ f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)}) \end{pmatrix}, \qquad (5.5.1)$$

where $\mathbf{z}_\Omega^{(1)}$ denotes the M_Ω-dimensional vector of the z_i for $i = 1,\dots,M_\Omega$ associated to the interior points $\mathbf{x}_\Omega$ while $f(\mathbf{x}_\Omega), g(\mathbf{x}_{\partial\Omega})$ and $\tau(\mathbf{z}_\Omega^{(1)})$ are vectors obtained by applying the corresponding functions to entries of their input vectors. To be clear, the expression in (5.5.1) represents a weighted least-squares optimization problem, and the transpose signs in the row vector multiplying the matrix have been suppressed for notational brevity. In [43], a Gauss-Newton method has been proposed to solve this problem. This method linearizes the nonlinear function τ at the current iteration and solves the resulting quadratic optimization problem to obtain the next iterate.

We present the Gauss-Newton algorithm in a slightly different but equivalent way that is more convenient for exposition. To that end, we consider the general formulation in (5.2.5). In the nonlinear elliptic PDE example, we have $\boldsymbol{\phi} = (\delta_{\mathbf{x}_\Omega}, \delta_{\mathbf{x}_{\partial\Omega}}, \delta_{\mathbf{x}_\Omega}\circ\Delta)$ where $\delta_{\mathbf{x}_\Omega}$ denotes the collection of Diracs measurements $(\delta_{\mathbf{x}_1},\dots,\delta_{\mathbf{x}_{M_\Omega}})$; the definition of $\delta_{\mathbf{x}_{\partial\Omega}}$ and $\delta_{\mathbf{x}_\Omega}\circ\Delta$ follows similarly. We also write correspondingly $\mathbf{z} = (\mathbf{z}_\Omega, \mathbf{z}_{\partial\Omega}, \mathbf{z}_\Omega^\Delta) \in \mathbb{R}^N$ with $N = M + M_\Omega$. Then $F(\mathbf{z}) = (-\mathbf{z}_\Omega^\Delta + \tau(\mathbf{z}_\Omega), \mathbf{z}_{\partial\Omega}) \in \mathbb{R}^M$ and $\mathbf{y} = (f(\mathbf{x}_\Omega), g(\mathbf{x}_{\partial\Omega})) \in \mathbb{R}^M$, such that $F(\mathbf{z}) = \mathbf{y}$.

The Gauss-Netwon iterations for solving (5.5.1) is equivalent to the following se-

quential quadratic programming approach for solving (5.2.5): for $k \in \mathbb{N}$, assume $\mathbf{z}^k$ obtained, then $\mathbf{z}^{k+1}$ is given by

$$\begin{aligned} \mathbf{z}^{k+1} = \underset{\mathbf{z}\in\mathbb{R}^N}{\arg\min} \quad & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} \quad & F(\mathbf{z}^k) + DF(\mathbf{z}^k)(\mathbf{z}-\mathbf{z}^k) = \mathbf{y}, \end{aligned} \tag{5.5.2}$$

where $DF(\mathbf{z}^k) \in \mathbb{R}^{M\times N}$ is the Jacobian of F at $\mathbf{z}^k$. The above is a quadratic optimization with a linear constraint. Using Lagrangian multipliers, we get the explicit formula of the solution: $\mathbf{z}^{k+1} = K(\boldsymbol{\phi},\boldsymbol{\phi})(DF(\mathbf{z}^k))^T\gamma$, where $\gamma \in \mathbb{R}^M$ solves the linear system

$$\left(DF(\mathbf{z}^k)K(\boldsymbol{\phi},\boldsymbol{\phi})(DF(\mathbf{z}^k))^T\right)\gamma = \mathbf{y} - F(\mathbf{z}^k) + DF(\mathbf{z}^k)\mathbf{z}^k\,. \tag{5.5.3}$$

Now, we introduce the *reduced* set of measurements $\boldsymbol{\phi}^k = DF(\mathbf{z}^k)\boldsymbol{\phi}$. For the nonlinear elliptic PDE, we have

$$\boldsymbol{\phi}^k = (-\delta_{\mathbf{x}_\Omega}\circ\Delta + \tau(\mathbf{z}^k_\Omega)\cdot\delta_{\mathbf{x}_\Omega}, \delta_{\mathbf{x}_{\partial\Omega}}),$$

where $\tau(\mathbf{z}^k_\Omega)\cdot\delta_{\mathbf{x}_\Omega} := (\tau(\mathbf{z}^k_1)\delta_{\mathbf{x}_1}, ..., \tau(\mathbf{z}^k_{M_\Omega})\delta_{\mathbf{x}_{M_\Omega}})$. Then, we can equivalently write the solution as $\mathbf{z}^{k+1} = K(\boldsymbol{\phi},\boldsymbol{\phi})(DF(\mathbf{z}^k))^T\gamma$ where γ satisfies

$$K(\boldsymbol{\phi}^k,\boldsymbol{\phi}^k)\gamma = \mathbf{y} - F(\mathbf{z}^k) + DF(\mathbf{z}^k)\mathbf{z}^k\,. \tag{5.5.4}$$

Note that $K(\boldsymbol{\phi}^k,\boldsymbol{\phi}^k) \in \mathbb{R}^{M\times M}$, in contrast to $K(\boldsymbol{\phi},\boldsymbol{\phi}) \in \mathbb{R}^{N\times N}$. The dimension is reduced. The computational bottleneck lies in the linear system with the reduced kernel matrix $K(\boldsymbol{\phi}^k,\boldsymbol{\phi}^k)$.

5.5.2 Sparse Cholesky factorization for the reduced kernel matrices

As $K(\boldsymbol{\phi}^k,\boldsymbol{\phi}^k)$ is also a kernel matrix with derivative-type measurements, we hope to use the sparse Cholesky factorization idea to approximate its inverse. The first question, again, is how to order these measurements.

To begin with, we examine the structure of the reduced kernel matrix. Note that as $F(\mathbf{z}) = \mathbf{y}$ encodes the PDE at the collocation points, the linearization of F in (5.5.2) is also equivalent to first linearizing the PDE at the current solution and then applying the kernel method. Thus, $\boldsymbol{\phi}^k$ will typically contain M_Ω interior measurements corresponding to the linearized PDE at the interior points and $M - M_\Omega$ boundary measurements corresponding to the sampled boundary condition. For problems with Dirichlet's boundary condition, which are the main focus of this chapter, the

boundary measurements are of Diracs type. It is worth noting that, in contrast to $K(\boldsymbol{\phi}, \boldsymbol{\phi})$, we no longer have Diracs measurements at every interior point now.

We propose to order the boundary Diracs measurements first, using the maximin ordering on $\partial\Omega$. Then, we order the interior derivative-type measurements using the maximin ordering in Ω, conditioned on $\partial\Omega$. We use numerical experiments to investigate the screening effects under such ordering. Suppose Θ is the reordered version of the reduced kernel matrix $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)$, then similar to Figures 5.1 and 5.2, we show the magnitude of the corresponding Cholesky factor of $\Theta^{-1} = U^\star U^{\star T}$, i.e., we plot $|U^\star_{ij}|$ for $i \leq j$; here j is selected to correspond to some boundary and interior points. The result can be found in Figure 5.5.

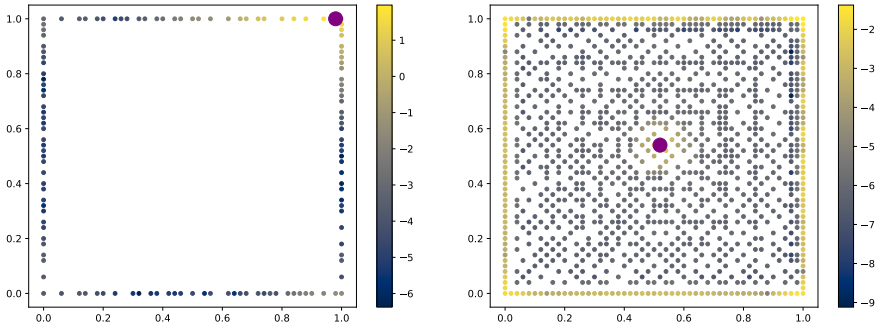


Figure 5.5: Demonstration of screening effects for the reduced kernel matrix. We choose the Matérn kernel with $\nu = 5/2$; the lengthscale parameter is 0.3. The data points are equidistributed in $[0, 1]^2$ with grid size $h = 0.02$. In the left figure, we show a boundary point, and all the points ordered before are marked with a color whose intensity scales with the entry value $|U^\star_{ij}|$. The right figure is obtained in the same manner but for an interior measurement.

From the left figure, we observe a desired screening effect for boundary Diracs measurements. However, in the right figure, we observe that the interior derivative-type measurements still exhibit a strong conditional correlation with boundary points. That means that the correlation with boundary points is not screened thoroughly. This also implies that the presence of the interior Diracs measurements is the key to the sparse Choleksy factors for the previous $K(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\phi}})^{-1}$.

The right of Figure 5.5 demonstrates a negative result: one cannot hope that the Cholesky factor of Θ^{-1} will be as sparse as before. However, algorithmically, we can still apply the sparse Cholesky factorization to the matrix. We present numerical experiments to test the accuracy of such factorization. In the right of Figure 5.4, we show the KL errors of the resulting factorization concerning the sparsity parameter

ρ. Even though the screening effect is not perfect, as we discussed above, we still observe a consistent decay of the KL errors when ρ increases.

In addition, although we cannot theoretically guarantee the factor is as accurate as before, we can use it as a preconditioner to solve linear systems involving $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)$. In practice, we observe that this idea works very well, and nearly constant steps of preconditioned conjugate gradient (pCG) iterations can lead to an accurate solution to (5.5.4). As a demonstration, in Figure 5.6, we show the pCG iteration history when the preconditioning idea is employed. The stopping criterion for pCG is that the relative tolerance is smaller than $2^{-26} \approx 10^{-8}$, which is the default criterion in Julia. From the figures, we can see that pCG usually converges in 10-40 steps, and this fast convergence is insensitive to the numbers of points. When ρ is large, the factor is more accurate, and the preconditioner is better, leading to smaller number of required pCG steps. Among all the cases, the number of pCG steps required to reach the stopping criterion is of the same magnitude, and is not large.

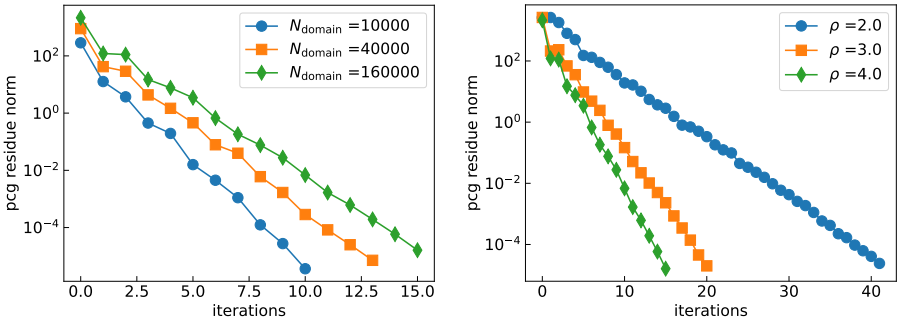


Figure 5.6: Demonstration of the convergence history of the pCG iteration. We choose the Matérn kernel with $\nu = 5/2$; the lengthscale parameter is 0.3. In the left figure, we choose the data points to be equidistributed in $[0,1]^2$ with different grid sizes; $\rho = 4.0$. We show the equation residue norms of the iterates in each pCG iteration. In the right figure, we choose the data points to be equidistributed in $[0,1]^2$ with grid size 0.0025 so that $N_{\text{domain}} = 160000$. We plot the pCG iteration history for $\rho = 2.0, 3.0, 4.0$.

It is worth noting that since $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k) = DF(\mathbf{z}^k)K(\boldsymbol{\phi}, \boldsymbol{\phi})(DF(\mathbf{z}^k))^T$ and we have a provably accurate sparse Cholesky factorization for $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}$, the matrix-vector multiplication for $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)$ in each pCG iteration is efficient.

5.5.3 General case

The description in the last subsection applies directly to general nonlinear PDEs, which correspond to general $\boldsymbol{\phi}$ and F in (5.2.5). We use the maximin ordering on

the boundary, followed by the conditioned maximin ordering in the interior. We denote the ordering by $Q : I \to I$. The lengthscale is defined by

$$l_i = \text{dist}(\mathbf{x}_{Q(i)}, \{\mathbf{x}_{Q(1)}, ..., \mathbf{x}_{Q(i-1)}\} \cup \mathsf{A}), \tag{5.5.5}$$

where for a boundary measurement, $\mathsf{A} = \emptyset$ and for an interior measurement $\mathsf{A} = \partial\Omega$. With the ordering and lengthscales, we create the sparse pattern through (5.3.4) (and aggregate it using the supernodes idea) and apply the KL minimization in Subsection 5.3.1 to obtain an approximate factor for $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)^{-1}$. The general algorithmic procedure is outlined in Algorithm 2. We now denote the sparsity parameter for the reduced kernel matrix by ρ_r.

Algorithm 2 Sparse Cholesky factorization for $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)^{-1}$

1: **Input**: Measurements $\boldsymbol{\phi}^k$, kernel function K, sparsity parameter ρ_r, supernodes parameter λ
2: **Output**: $U_\text{r}^{\rho_\text{r}}, Q_\text{perm}$
3: Reordering and sparsity pattern: we first order the boundary measurements using the maximin ordering. Next, we order the interior measurements using the maximin ordering conditioned on $\partial\Omega$. This process yields a permutation matrix denoted by Q_perm such that $Q_\text{perm}\boldsymbol{\phi}^k = \tilde{\boldsymbol{\phi}}^k$, and lengthscales l for each measurement in $\tilde{\boldsymbol{\phi}}^k$. Under the ordering, we construct the aggregate sparsity pattern $S_{Q,l,\rho_\text{r},\lambda}$ based on the chosen values of ρ_r and λ.
4: KL minimization: solve (5.3.6) with $\Theta = K(\tilde{\boldsymbol{\phi}}^k, \tilde{\boldsymbol{\phi}}^k)$, by (5.3.7), to obtain $U_\text{r}^{\rho_\text{r}}$
5: **return** $U_\text{r}^{\rho_\text{r}}, Q_\text{perm}$

Now, putting all things together, we outline the general algorithmic procedure for solving the PDEs using the second-order Gauss-Newton method, in Algorithm 3.

For the choice of parameters, we usually set t to be between 2 to 10. Setting $\rho = O(\log(N/\epsilon))$ suffices to obtain an ϵ-accurate approximation of $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}$. We do not have a theoretical guarantee for the factorization algorithm applied to the reduced kernel matrix $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)^{-1}$. Still, our experience indicates that setting $\rho_\text{r} = \rho$ or a constant such as $\rho_\text{r} = 3$ works well in practice. We note that a larger ρ_r increases the factorization time while decreasing the necessary pCG steps to solve the linear system, as demonstrated in the right of Figure 5.6. There is a trade-off here in general.

The overall complexity of Algorithm 3 for solving (5.5.2) is $O(N\log^2(N)\rho^d + N\rho^{2d} + Mt\rho_\text{r}^{2d} + T_\text{pCG})$ in time and $O(N\rho^d + M\rho_\text{r}^d)$ in space, where $O(N\log^2(N)\rho^d)$ is the time for generating the ordering and sparsity pattern, $O(N\rho^{2d})$ is for the

Algorithm 3 Sparse Cholesky accelerated Gauss-Newton for solving (5.2.5)

Input: Measurements $\boldsymbol{\phi}$, data functional F, data vector $\mathbf{y}$, kernel function K, number of Gauss-Newton steps t, sparsity parameters ρ, ρ_{r}, supernodes parameter λ
Output: Solution $\mathbf{z}^t$
Factorize $K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} \approx P_{\mathrm{perm}}^T U^\rho U^{\rho T} P_{\mathrm{perm}}$ using Algorithm 1
Set $k = 0$, $\mathbf{z}^k = \mathbf{0}$ or other user-specified initial guess
while $k < t$ **do**
    Form the reduced measurements $\boldsymbol{\phi}^k = DF(\mathbf{z}^k)\boldsymbol{\phi}$
    Factorize $K(\boldsymbol{\phi}^k, \boldsymbol{\phi}^k)^{-1}$ to get $Q_{\mathrm{perm}}^T U_{\mathrm{r}}^{\rho_{\mathrm{r}}} U_{\mathrm{r}}^{\rho_{\mathrm{r}} T} Q_{\mathrm{perm}}$ using Algorithm 2
    Use pCG to solve (5.5.4) with the preconditioner $Q_{\mathrm{perm}}^T U_{\mathrm{r}}^{\rho_{\mathrm{r}}} U_{\mathrm{r}}^{\rho_{\mathrm{r}} T} Q_{\mathrm{perm}}$
    $\mathbf{z}^{k+1} = (P_{\mathrm{perm}}^T U^\rho U^{\rho T} P_{\mathrm{perm}}) \backslash ((DF(\mathbf{z}^k))^T \gamma)$
    $k = k + 1$
end while
return $\mathbf{z}^t$

factorization, and $O(Mt\rho_{\mathrm{r}}^{2d})$ is for the factorizations in all the GN iterations, T_{pCG} is the time that the pCG iterations take.

Based on empirical observations, we have found that T_{pCG} scales nearly linearly with respect to $N\rho^d$. This is because a nearly constant number of pCG iterations are sufficient to obtain an accurate solution, and each pCG iteration takes at most $O(N\rho^d)$ time, as explained in the matrix-vector multiplication mentioned at the end of Subsection 5.5.2. Additionally, it is worth noting that the time required for generating the ordering and sparsity pattern ($O(N \log^2(N)\rho^d)$) is negligible in practice, compared to that for the KL minimization. Furthermore, the ordering and sparsity pattern can be pre-computed once and reused for multiple runs.

5.6 Numerical Experiments

In this section, we use Algorithm 3 to solve nonlinear PDEs. The numerical experiments are conducted on the personal computer MacBook Pro 2.4 GHz Quad-Core Intel Core i5. In all the experiments, the physical data points are equidistributed on a grid; we specify its size in each example. We always set the sparsity parameter for the reduced kernel matrix $\rho_{\mathrm{r}} = \rho$, the sparsity parameter for the original matrix. We adopt the supernodes ideas in all the examples and set the parameter $\lambda = 1.5$.

Our theory guarantees that once the Diracs measurements are ordered first by the maximin ordering, the derivative measurements can be ordered arbitrarily. In practice, for convenience, we order them from lower-order to high-order derivatives, and for the same type of derivatives, we order the corresponding measurements based

on their locations, in the same maximin way as the Diracs measurements.

Our codes are in `https://github.com/yifanc96/PDEs-GP-KoleskySolver`.

5.6.1 Nonlinear elliptic PDEs

Our first example is the nonlinear elliptic equation

$$\begin{cases} -\Delta u + \tau(u) = f & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{5.6.1}$$

with $\tau(u) = u^3$. Here $\Omega = [0,1]^2$. We set

$$u(\mathbf{x}) = \sum_{k=1}^{600} \frac{1}{k^6} \sin(k\pi x_1) \sin(k\pi x_2)$$

as the ground truth and use it to generate the boundary and right hand side data. We set the number of Gauss-Newton iterations to be 3. The initial guess for the iteration is a zero function. The lengthscale of the kernels is set to be 0.3.

We first study the solution error and the CPU time regarding ρ. We choose the number of interior points to be $N_{\text{domain}} = 40000$; or equivalently, the grid size $h = 0.005$. In the left side of Figure 5.7, we observe that a larger ρ leads to a smaller L^2 error of the solution. For the Matérn kernel with $\nu = 5/2, 7/2$, we observe that such accuracy improvement saturates at $\rho = 2$ or 4, while when $\nu = 7/2$. the accuracy keeps improving until $\rho = 10$. This high accuracy for large ν is because the solution u is fairly smooth. Using smoother kernels can lead to better approximation accuracy. On the other hand, smoother kernels usually need a larger ρ to achieve the same level of approximation accuracy, as we have demonstrated in the left of Figure 5.3.

In the right side of Figure 5.7, we show the CPU time required to compute the solution for different kernels and ρ. A larger ρ generally leads to a longer CPU time. But there are some exceptions: for the Matérn kernel with $\nu = 5/2, 7/2$, the CPU time for $\rho = 3$ is longer than that for $\rho = 2$. Again, the reason is that these smoother kernels often require a larger ρ for accurate approximations. When ρ is very small, although the sparse Cholesky factorization is very fast, the pCG iterations could take long since the preconditioner matrix does not approximate the matrix well.

We then study the L^2 errors and CPU time regarding the number of physical points. We fix $\rho = 4.0$. In the left of Figure 5.8, we observe that the accuracy improves when N_{domain} increases. For the smoother Matérn kernels with $\nu = 7/2, 9/2$, they

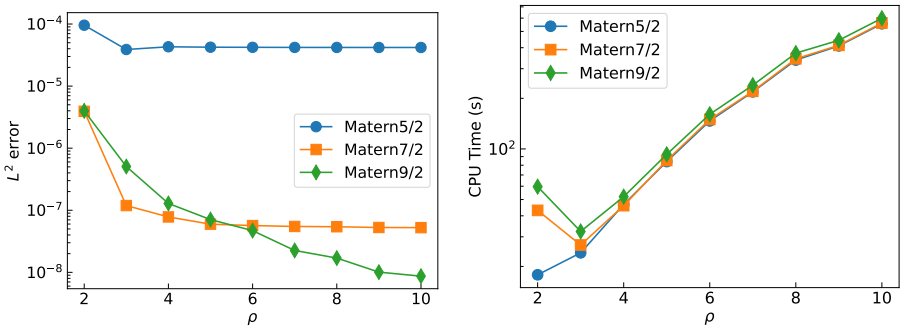


Figure 5.7: Nonlinear elliptic PDE example. The left figure concerns the L^2 errors of the solution, while the right figure concerns the CPU time. Both plots are with regard to ρ. We set $N_{\text{domain}} = 40000$.

will hit an accuracy floor of 10^{-7}. This is because we only have a finite number of Gauss-Newton steps and a finite ρ. In the right of Figure 5.8, a near-linear complexity in time regarding the number of points is demonstrated.

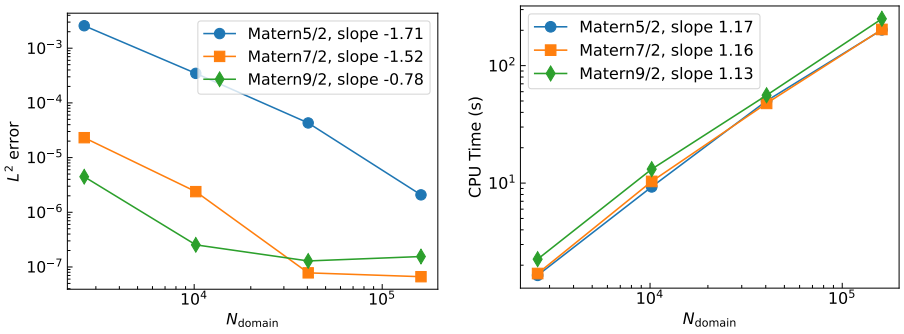


Figure 5.8: Nonlinear elliptic PDE example. The left figure concerns the L^2 errors of the solution, while the right figure concerns the CPU time. Both plots are with regard to the number of physical points in the domain. We set $\rho = 4.0$.

5.6.2 Burgers' equation

Our second example concerns the time-dependent Burgers equation:

$$\begin{aligned} \partial_t u + u\partial_x u - 0.001\partial_x^2 u = 0, \quad \forall (x,t) \in (-1,1) \times (0,1], \\ u(x,0) = -\sin(\pi x), \\ u(-1,t) = u(1,t) = 0. \end{aligned} \tag{5.6.2}$$

Rather than using a spatial-temporal GP as in [43], we first discretize the equation in time and then use a spatial GP to solve the resulting PDE in space. This reduces

the dimensionality of the system and is more efficient. More precisely, we use the Crank–Nicolson scheme with time stepsize Δt to obtain

$$\begin{aligned}\frac{\hat{u}(x,t_{n+1})-\hat{u}(x,t_n)}{\Delta t}+\frac{1}{2}\left(\hat{u}(x,t_{n+1})\partial_x\hat{u}(x,t_{n+1})+\hat{u}(x,t_n)\partial_x\hat{u}(x,t_n)\right)\\ =\frac{0.001}{2}\left(\partial_x^2\hat{u}(x,t_{n+1})+\partial_x^2\hat{u}(x,t_n)\right),\end{aligned} \tag{5.6.3}$$

where $\hat{u}(t_n,x)$ is an approximation of the true solution $u(t_n,x)$ with $t_n=n\Delta t$. When $\hat{u}(\cdot,t_n)$ is known, (5.6.3) is a spatial PDE for the function $\hat{u}(\cdot,t_{n+1})$. We can solve (5.6.3) iteratively starting from $n=0$. We use two steps of Gauss-Newton iterations with the initial guess as the solution at the last time step.

We set $\Delta t=0.02$ and compute the solution at $t=1$. The lengthscale of the kernels is chosen to be 0.02. We set $\rho=4.0$ in the factorization algorithm. In the left of Figure 5.9, we show our numerical solution by using a grid of size $h=0.001$ and the true solution computed by using the Cole-Hopf transformation. We see that they match very well, and the shock is captured. This is possible because we use a grid of small size so that the shock is well resolved. With a very small grid size, we need to deal with many large-size dense kernel matrices, and we use the sparse Cholesky factorization algorithm to handle such a challenge.

In the right of Figure 5.9, we show the CPU time of our algorithm regarding different N_{domain}. We clearly observe a near-linear complexity in time. The total CPU time is less than 10 seconds to handle 50 dense kernel matrices (since $1/\Delta t=50$) of size larger than 10^4 (the dimension of $K(\boldsymbol{\phi},\boldsymbol{\phi})$ is around $3\times N_{\text{domain}}$ since we have three types of measurements) sequentially.

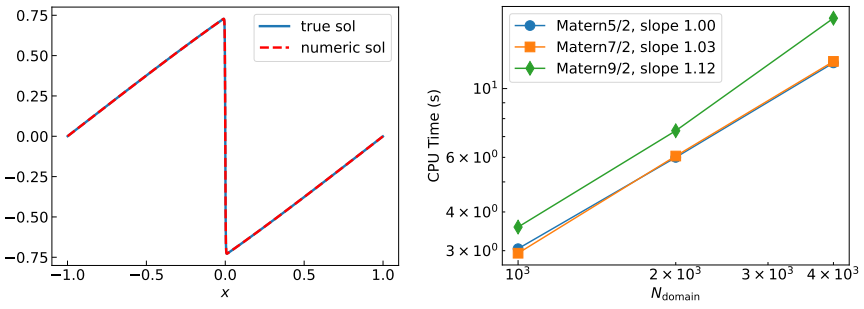


Figure 5.9: Burgers' equation example. The left figure is a demonstration of the numerical solution and true solution at $t=1$. The right figure concerns the CPU time regarding the number of physical points. We set $\rho=4.0$.

We also show the accuracy of our solutions in the following Table 5.1. We observe

N_{domain}	1000	2000	4000
L^2 error	1.729e-4	6.111e-5	7.453e-5
L^∞ error	1.075e-3	2.745e-4	1.075e-4

Table 5.1: Burgers' equation example. The L^2 and L^∞ errors of the computed solution at $t = 1$. We use the Matérn kernel with $\nu = 7/2$. The sparsity parameter $\rho = 4.0$.

high accuracy, $O(10^{-5})$ in the L^2 norm and $O(10^{-4})$ in the L^∞ norm. The L^2 errors do not decrease when we increase the number of points from 2000 to 4000. It is because we use a fixed time stepsize $\Delta t = 0.02$ and a fixed $\rho = 4.0$.

5.6.3 Monge-Ampère equation

Our last example is the Monge-Ampère equation in two dimensional space.

$$\det(D^2 u) = f, \quad \mathbf{x} \in (0,1)^2 . \tag{5.6.4}$$

Here, we choose $u(\mathbf{x}) = \exp\big(0.5((x_1 - 0.5)^2 + (x_2 - 0,5)^2)\big)$ to generate the boundary and right hand side data. To ensure uniqueness of the solution, some convexity assumption is usually needed. Here, to test the wide applicability of our methodology, we directly implement Algorithm 3. We adopt 3 steps of Gauss-Newton iterations with the initial guess $u(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$. We choose the Matérn kernel with $\nu = 5/2$. The lengthscale of the kernel is set to be 0.3.

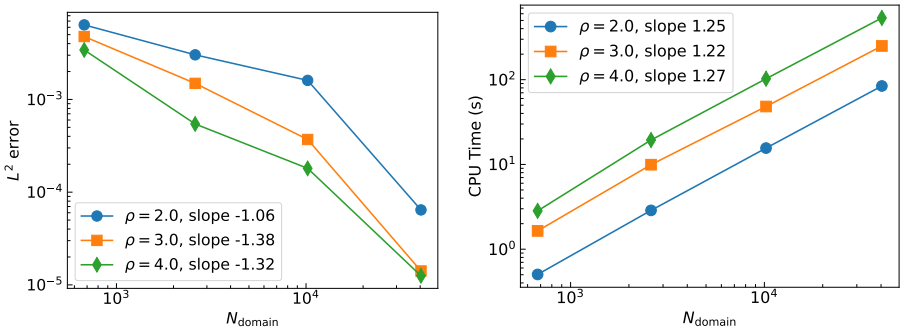


Figure 5.10: The Monge-Ampère equation example. The left figure concerns the L^2 errors, while the right figure concerns the CPU time. Both are with respect to the number of physical points in space, and in both figures, we consider $\rho = 2.0, 3.0, 4.0$. We choose the Matérn kernel with $\nu = 5/2$ in this example.

In Figure 5.10, we present the L^2 errors of the solution and the CPU time with respect to N_{domain}. Once again, we observe a nearly linear complexity in time. However, since $\det\big(D^2 u\big)$ involves several partial derivatives of the function, we

need to differentiate our kernels accordingly; we use auto-differentiation in Julia for convenience, which is slightly slower than the hand-coded derivatives used in our previous numerical examples. Consequently, the total CPU time is longer compared to the earlier examples, although the scaling regarding N_{domain} remains similar.

As N_{domain} increases, the L^2 solution errors decrease for $\rho = 2.0, 3.0, 4.0$. This indicates that our kernel method is convergent for such a fully nonlinear PDE. However, since we do not incorporate singularity into the solution, this example may not correspond to the most challenging setting. Nonetheless, the success of this simple methodology combined with systematic fast solvers demonstrates its potential for promising automation and broader applications in solving nonlinear PDEs.

5.7 Conclusions

We have investigated a sparse Cholesky factorization algorithm that enables scaling up the GP method for solving nonlinear PDEs. Our algorithm relies on a novel ordering of the Diracs and derivative-type measurements that arise in the GP-PDE methodology. With this ordering, the Cholesky factor of the inverse kernel matrix becomes approximately sparse, and we can use efficient KL minimization, equivalent to Vecchia approximation, to compute the sparse factors. We have provided rigorous analysis of the approximation accuracy by showing the exponential decay of the conditional covariance of GPs and the Cholesky factors of the inverse kernel matrix, for a wide class of kernel functions and derivative measurements.

When using second-order Gauss-Newton methods to solve the nonlinear PDEs, a reduced kernel matrix arises, in which many interior Dirac measurements are absent. In such cases, the decay is weakened, and the accuracy of the factorization deteriorates. To compensate for this loss of accuracy, we use pCG iterations with this approximate factor as a preconditioner. In our numerical experiments, our algorithm achieves high accuracy, and the computation time scales near-linearly with the number of points. This justifies the potential of GPs for solving general PDEs with automation, efficiency, and accuracy. We anticipate extending our algorithms to solving inverse problems and our theories to more kernel functions and measurement functionals in the future.

Chapter 6

CONSISTENCY OF HIERARCHICAL LEARNING FOR GAUSSIAN PROCESSES

In this chapter, we study and analyze the use of hiearchical learning for Gaussian processes to improve the adaptivity of the algorithm. The exposition is based on our work [51] published in *Mathematics of Computation*, 90(332):2527–2578, 2021.

6.1 Introduction

6.1.1 Background and Context

Gaussian process regression (GPR) is important in its own right, and as a prototype for more complex inverse problems in which there is a possibly indirect, nonlinear set of observations. An important reason for the success of GPR in applications is its ability to learn hyperparameters, entering through a hierarchical prior, from data. Learning of these hyperparameters is typically achieved through fully Bayesian (sampling) or empirical Bayesian (optimization) methods. However, new approaches suggested in the machine learning literature, particularly the kernel flow method [206], rely on approximation theoretic criteria that can be traced back to the classical idea of cross-validation for model selection. The primary goal of this work is to study and compare these two approaches. Special attention will be paid to their large data consistency, implicit bias, and robustness to model misspecification.

6.1.2 Gaussian Process Regression

We start with a brief introduction to GPR; for simplicity, we focus on the noise-free scenario. The target is to recover a function $u^\dagger : D \mapsto \mathbb{R}$ from pointwise data $y_i = u^\dagger(x_i)$ for $1 \le i \le N$, where $x_i \in D \subset \mathbb{R}^d$ and D is a compact domain. This problem often appears in fields such as supervised learning in machine learning, non-parameteric regression in statistics, and interpolation in numerical analysis.

The GPR solution to this problem is as follows. Given a family of positive definite covariance/kernel functions $K_\theta : D \times D \to \mathbb{R}$ where $\theta \in \Theta$ is a hyperparameter, GPR approximates $u^\dagger$ with the conditional expectation

$$u(\cdot, \theta, \mathcal{X}) := \mathbb{E}\left[\xi(\cdot,\theta) \mid \xi(\mathcal{X},\theta) = u^\dagger(\mathcal{X})\right] = K_\theta(\cdot,\mathcal{X})[K_\theta(\mathcal{X},\mathcal{X})]^{-1}u^\dagger(\mathcal{X}), \quad (6.1.1)$$

where $\xi(\cdot,\theta) \sim \mathcal{GP}(0, K_\theta)$ is a centered Gaussian process[1](GP) with covariance function K_θ. We have used the following compressed notation:

$$\mathcal{X} := (x_1, \ldots, x_N)^\intercal \quad \text{and} \quad u^\dagger(\mathcal{X}) := (u^\dagger(x_1), \ldots, u^\dagger(x_N))^\intercal .$$

Moreover, $K_\theta(\mathcal{X},\mathcal{X})$ denotes the $N \times N$ dimensional Gram matrix with $(i,j)^{\text{th}}$ entry $K_\theta(x_i, x_j)$, and $K_\theta(\cdot,\mathcal{X})$ is a function mapping D to $\mathbb{R}^N$ with i^{th} component $K_\theta(\cdot, x_i) : D \mapsto \mathbb{R}$.

Normally, every $\theta \in \Theta$ produces a solution $u(\cdot,\theta,\mathcal{X})$ that agrees with $u^\dagger$ on $\mathcal{X}$. Nevertheless, different choices may yield distinct out-of-sample errors, known as generalization errors in the machine learning context. Therefore, it is of paramount importance to learn a good hierarchical parameter θ adaptively from data.

6.1.3 Two Approaches

In this chapter, we study two approaches to the question posed above, both based on selecting θ as the optimizer of a variational problem.

6.1.3.1 Empirical Bayes Approach

The empirical Bayes (EB) approach addresses the question by proposing a statistical model. It formulates a prior distribution on the pair (ξ,θ) by assuming that θ is sampled from a prior distribution and ξ is then sampled from the conditional distribution of $\xi|\theta$; then, it finds the posterior distribution of the pair (ξ,θ) conditioned on $\xi(\mathcal{X}) = u^\dagger(\mathcal{X})$, and selects the parameter θ that maximizes the marginal probability of θ under this posterior. For simplicity, we work with uninformative priors, which lead to the following objective function:

$$\mathrm{L}^{\mathrm{EB}}(\theta,\mathcal{X},u^\dagger) = u^\dagger(\mathcal{X})^\intercal [K_\theta(\mathcal{X},\mathcal{X})]^{-1} u^\dagger(\mathcal{X}) + \log\det K_\theta(\mathcal{X},\mathcal{X}) . \tag{6.1.2}$$

This is also twice the negative marginal log likelihood of θ given the data $u^\dagger(\mathcal{X})$. Then, EB will choose θ by minimizing this objective function, namely

$$\theta^{\mathrm{EB}}(\mathcal{X},u^\dagger) := \operatorname*{arg\,min}_{\theta\in\Theta} \mathrm{L}^{\mathrm{EB}}(\theta,\mathcal{X},u^\dagger) . \tag{6.1.3}$$

[1]Recall that the covariance function K_θ of a Gaussian process $\mathcal{GP}(0,K_\theta)$ is the kernel of the integral operator representation of C_θ in the covariance operator notation $\mathcal{N}(0,C_\theta)$. Connections between these perspectives are reviewed in Subsection 6.2.1. We will use the covariance operator notation more frequently later in this chapter.

6.1.3.2 Approximation Theoretic Approach

Approximation theoretic considerations, on the other hand, provide a different answer without proposing statistical models. This methodology proceeds by asking for an ideal θ that minimizes $\mathsf{d}(u^\dagger, u(\cdot,\theta,\mathcal{X}))$ for some cost function d. Though in practice $u^\dagger$ is not available, there are ideas in cross-validation that split $\mathcal{X}$ into training data and validation data, and use the approximation error in validation data to estimate the exact error. Inspired by this idea, we could turn to optimize the following objective function:

$$\mathsf{d}(u(\cdot,\theta,\mathcal{X}), u(\cdot,\theta,\pi\mathcal{X}))\,, \tag{6.1.4}$$

where we write $\pi\mathcal{X}$ for a subset of $\mathcal{X}$ obtained by subsampling a proportion, say one-half, of $\mathcal{X}$.

In this work, we focus on a particular choice of d that originates from the Kernel Flow (KF) approach [206]. To describe it, we denote by $(\mathcal{H}_\theta, \|\cdot\|_{K_\theta})$ the associated *Reproducing Kernel Hilbert Space* (RKHS) for the kernel K_θ; note that $\|K_\theta(\cdot,x)\|^2_{K_\theta} = K_\theta(x,x)$. The objective function in KF is chosen as

$$\mathsf{L}^{\mathrm{KF}}(\theta,\mathcal{X},\pi\mathcal{X},u^\dagger) := \frac{\|u(\cdot,\theta,\mathcal{X}) - u(\cdot,\theta,\pi\mathcal{X})\|^2_{K_\theta}}{\|u(\cdot,\theta,\mathcal{X})\|^2_{K_\theta}}\,. \tag{6.1.5}$$

This measures the discrepancy in the RKHS norm between the GPR solution using the whole data $\mathcal{X}$ and using a subset of the data $\pi\mathcal{X}$, normalized by the RKHS norm of the former.

Remark 6.1.1. *As explained above, we understand the numerator as an estimation of the error $\|u^\dagger - u(\cdot,\theta,\mathcal{X})\|^2_{K_\theta}$. Such error estimate, based on comparing solutions obtained via different data resolutions, is a widely used idea in numerical analysis.*

Based on Garlerkin orthogonality (see [206]), the objective function admits a finite dimensional representation formula that is convenient for numerical computation:

$$\mathsf{L}^{\mathrm{KF}}(\theta,\mathcal{X},\pi\mathcal{X},u^\dagger) = 1 - \frac{u^\dagger(\pi\mathcal{X})^\mathsf{T}[K_\theta(\pi\mathcal{X},\pi\mathcal{X})]^{-1}u^\dagger(\pi\mathcal{X})}{u^\dagger(\mathcal{X})^\mathsf{T}[K_\theta(\mathcal{X},\mathcal{X})]^{-1}u^\dagger(\mathcal{X})}\,. \tag{6.1.6}$$

Then, the KF estimator is defined as

$$\theta^{\mathrm{KF}}(\mathcal{X},\pi\mathcal{X},u^\dagger) := \operatorname*{arg\,min}_{\theta\in\Theta} \mathsf{L}^{\mathrm{KF}}(\theta,\mathcal{X},\pi\mathcal{X},u^\dagger)\,. \tag{6.1.7}$$

Remark 6.1.2. *The existence of the finite-sample formula* (6.1.6) *is attributed to the choice of the RKHS norm in comparing solutions. It is essentially a consequence of the standard representer theorem. Additional motivations for using the RKHS norm will be reviewed in Subsection 6.1.5.2.*

6.1.3.3 Guiding Observations and Goals

The EB and KF algorithms estimate the parameter θ from the observed data, the number of which can vary considerably. Thus, a basic question to ask is whether the estimators attain meaningful limits as data accumulate:

(1) Consistency: how do θ^{EB} and θ^{KF} behave in the large data limit, i.e., as the number of data N goes to infinity?

Meanwhile, since we have two estimators, it is natural to compare their performance. Indeed, we observe that EB and KF have distinct objectives: EB seeks to estimate the most likely parameters of the distribution assumed to generate the data, while KF chooses parameters to minimize an estimate of the approximation error in a parameter-dependent RKHS norm, targeting at the approximation efficiency of the underlying function. Moreover, EB is always probabilistic, while KF need not be.

These differences motivate the implicit bias question that has been popular in the machine learning community, and the model misspecification question that is common in mathematical modeling:

(2) Implicit bias: what are the selection bias of EB and KF, or, how should the obtained estimators θ^{EB} and θ^{KF} be interpreted in practice?

(3) Model misspecification: how do θ^{EB} and θ^{KF} behave when there is a mismatch between the data-generating mechanism and the model used to regress the data?

The precise goal of this work is to address these questions for certain concrete models, either theoretically or experimentally.

6.1.4 Our Contributions

Our contributions in this chapter are twofold and explained in the following two subsections.

6.1.4.1 Consistency and Implicit Bias

The first part of this work is devoted to the questions of consistency and implicit bias. We study a Matérn-like model on the torus, in which $u^\dagger$ is a sample drawn from the Matérn-like Gaussian process, with three parameters $\theta = (\sigma, \tau, s)$ that quantify the

amplitude, inverse lengthscale and regularity of the process. The detailed definition is in Subsection 6.2.1.

Our main analysis concerns learning the regularity parameter s using EB and KF. When the sampled points X are equidistributed, we achieve the following contributions:

- Consistency: we prove that the EB estimator converges to s in the large data limit, while the KF estimator converges to $\frac{s-d/2}{2}$, so that s is also determined. Their variances are also computed and compared.
- Implicit bias: we characterize the selection bias of EB and KF algorithms, in terms of the L^2 error between $u^\dagger$ and the GPR solution using learned parameters — this is the so-called generalization error. It is found that EB selects the parameter that achieves the minimal L^2 error in expectation, while KF selects the minimal parameter that suffices for the fastest rate of convergence of the L^2 error to 0 as the data density increases.

We can interpret these contributions from two perspectives. From the machine learning side, we are able to show that KF, as a machine learning method, has a well-defined large data limit for the Matérn-like model. Furthermore we can characterize clearly its implicit bias in terms of L^2 generalization errors. Thus, this work leads to a *first* theory for the KF learning algorithm.

From the spatial statistics side, our analysis contributes to a novel consistency theory for estimating the regularity parameter of Matérn-like fields in general dimensions. Such results are scarce in the spatial statistics literature; the techniques we use to prove consistency may be of independent interest and applicable beyond the setting considered here.

We also include numerical studies concerning the learning of the amplitude parameter σ and the inverse lengthscale parameter τ; these experiments contribute to a more complete picture of GPR using the Matérn-like field with hierarchical parameters. Moreover, we provide numerical experiments for several other well-specified models beyond the Matérn-like model, thus further extending the scope of discussions.

6.1.4.2 Model Misspecification

The second part of this work considers model misspecification: the data generating model for $u^\dagger$ and the model K_θ used for regression do not match. We adopt the following setting:

- We model the truth $u^\dagger$ either as a GP, using a variety of covariance functions, or as a deterministic function which solves a PDE.
- The kernel K_θ is chosen to be Green's function of various differential operators, where θ encodes information beyond the amplitude, lengthscale, and regularity of the field. For example we choose θ to be the location of a discontinuity within a conductivity field.

In this setting we observe distinct behavior distinguishing EB and KF. This raises the discussion of how to choose which algorithm to use when solving practical problems where misspecification is to be expected. Our numerical study explores several misspecification possibilities, showing that KF could be competitive with EB in certain scenarios.

6.1.5 Literature Review

In this subsection, we review the related literature. Several fields are of relevance, so we label them to help organize the review.

6.1.5.1 Regression and Inverse Problems

Regression is a form of inverse problem [63], and if formulated in a Bayesian fashion, it falls within the scope of Bayesian nonparametric estimation [98, 126]. In the paper [145] a simple class of linear inverse problems was studied from the perspective of posterior consistency, and it was demonstrated that the rate of posterior convergence depends sensitively on the relationship between regularity of the true function being sought, and the regularity of draws from the prior. This motivates the need for hierarchical procedures that adapt, on the basis of the data, the regularity of draws from the prior. In [144] the work in [145] was extended to cover the data-adapted learning of the regularity parameter in the prior; as the authors note: theoretical work "that supports the preference for empirical or hierarchical Bayes methods does not exist at the present time, however. It has until now been unknown whether these approaches can indeed robustify a procedure against prior mismatch.

In this paper, we answer this question in the affirmative." This analysis, however, requires simultaneous diagonalization of a self-adjoint operator formed from the forward model and the covariance operator, for all values of the hyper-parameter. Consistency is studied without this assumption in [288], and extended to the study of emulation within Bayesian inversion in [262] and to empirical Bayesian procedures in [267]. The papers [144] and [267] also use the EB loss function (6.1.2). In [71] estimation of hyper-parameters in Gaussian priors is discussed in the context of MAP estimators.

6.1.5.2 Kernel Flow and Cross-validation

The KF loss function in (6.1.6) was originally derived in [206] and motivated from the perspective of optimal recovery theory. It can be interpreted, from a numerical homogenization perspective [204], as the relative energy contained in the fine scales (in the unresolved part) of $u^\dagger$. In the paper [206], the proposed loss function to be optimized (via SGD) has the form

$$\mathbb{E}_{\pi_1}\mathbb{E}_{\pi_2}\mathsf{L}^{\mathrm{KF}}(\theta, \pi_1\mathcal{X}, \pi_2\pi_1\mathcal{X}, u^\dagger), \tag{6.1.8}$$

where $\pi_1\mathcal{X}$ is a subsampling of $\mathcal{X}$, and $\pi_2\pi_1\mathcal{X}$ is a further subsampling of $\pi_1\mathcal{X}$. This choice reduces the dimension of the kernel matrix and enables fast computation per iteration. Although the KF loss appears to be new, it can be seen as a variant of cross-validation (CV), which is a commonly used model selection/parameter estimation criteria [4, 97, 146]. A theoretical understanding of the consistency of CV "is very much of interest" [296] since its convergence rate can be shown to be asymptotically minimax [260] or near minimax optimal [274, 273] while having a lower computational complexity [300] than MLE (maximum likelihood estimation). The consistency of parameter estimation for the Ornstein-Uhlenbeck process has been studied in [297] for MLE, and [21] for CV.

In the setting of hyperparameters estimation of GPs, comparing MLE with CV can be traced back to Wahba [281] and Stein [257] who compared variants of these procedures[2] for choosing the smoothing parameter of a smoothing spline; they observed that while MLE is optimal when the model is well-specified, CV may perform better (than MLE) under misspecification (see also [20] for theoretical analysis and [286] for a practical example involving real data) and has a comparable rate of convergence when the model is correct (Stein [257] observed that "both estimates are

[2] modified maximum likelihood estimation and generalized cross validation.

asymptotically normal with the CV estimate having twice the asymptotic variance of the MLE estimate" and suggested that "The penalty for using CV instead of MLE when the stochastic model is correct is greater for higher-order smoothing splines, both in terms of the efficiency in estimating the smoothing parameter and the impact on subsequent predictions"). We also refer to [147] for a detailed numerical comparison between MLE and CV for estimating spline smoothing parameters. As observed in [242], these comparisons "are relevant for both numerical analysts and statisticians" since kernel interpolation can be interpreted as both approximating a deterministic unknown function from quadrature points or as estimating a sample from a Gaussian process from pointwise measurements.

6.1.5.3 Machine Learning and Kernel Learning

Kernel methods and GPs have long been used in machine learning [127, 228]. Learning a good kernel for a given task is very important in practice. Many works have tried to learn a kernel from data based on different criteria; for example, in [8], the kernel is modified to make the model have a large margin in classification, and in [56], the kernel is selected to have a small local Rademacher complexity. EB and KF loss functions have also been used in [228, 292, 206].

The recent discovery of the neural tangent kernel regime for overparameterized models [134] and the identification [202] of warping kernels [233, 213, 244, 206] as the infinite depth limit of residual neural networks [118] also suggest that a theoretical understanding of kernel selections may lead to important insights for neural network based machine learning. This line of work suggests that it may be fruitful to consider machine learning directly as the problem of selecting an underlying kernel (by minimizing nonlinear functionals of the empirical distribution such as (6.1.2) or (6.1.6)) and learning based on this kernel; in this perspective one has hierarchical GPR with kernel itself as the hyperparameter. This may be more effective than simply fitting the data by minimizing a generalized moment, i.e., a linear functional, of the empirical distribution, which is popularly used in empirical risk minimization. Numerical experiments presented in [298] and [114], based on the KF methodology in [206], provide evidence that (1) this point of view could improve test errors, generalization gaps, and robustness to distribution shifts in the training of of ANNs, and (2) kernel methods can be a simple and effective approach for learning dynamical systems and surrogate models, with the underlying kernel also learned from data (using KF and its variants). This further motivates the desire

to understand the KF-based estimation of θ.

6.1.6 Organization

The rest of this chapter is organized as follows. Section 6.2 is devoted to learning the regularity parameter of the Matérn-like model, where the large data consistency is proved and implicit bias is characterized. Most of the detailed proofs are deferred to Section C.1, and concise intuitive ideas are presented in Section 6.2 for the sake of readability. Section 6.3 considers other well-specified models, including the learning of the lengthscale and amplitude parameters in the Matérn-like model, or beyond the Matérn-like model. Experiments are provided concerning consistency and variance of these EB and KF estimators. Section 6.4 covers discussions on model misspecification through numerical studies. The purpose of the numerical experiments is twofold: (i) to demonstrate the extent to which the ideas learned through the analysis of consistency, which focuses primarily on the regularity parameter, extends to other parameters; (ii) to compare the performance of the EB and KF estimators quantitatively, since use of the latter is somewhat new in this area and its potential pros and cons need to be evaluated. Finally, we conclude this chapter in Section 6.5.

6.2 Regularity Parameter Learning for the Matérn-like Model

In this section, we study a Matérn-like model on the torus. We start with definitions of this model in Subsection 6.2.1, followed with definitions of EB and KF estimators in this context in Subsection 6.2.2. Then, in Subsection 6.2.3, we present our theory for the consistency of EB and KF estimators in learning the regularity parameter, with experiments included to demonstrate the correctness and implications of the theory. In particular, the implicit bias of these two estimators is explained. We outline the sketch of proofs for the theoretical result in Subsections 6.2.4, 6.2.5 and 6.2.6, and summarize several observations in Subsection 6.2.7. Subsection 6.2.8 provides additional experiments discussing the variance of these estimators.

6.2.1 The Matérn-like Model

We follow the general set-up in Subsections 6.1.2 and 6.1.3, where we have mentioned all the abstract ingredients such as the physical domain D, the truth $u^\dagger$, the kernel K_θ, and the data location X. In the current and next subsections, we will specify the exact meaning of these terms for a Matérn-like model on the torus. We will also make remarks to explain its connection to the standard Whittle-Matérn

process in the whole domain; see Remark 6.2.2.

6.2.1.1 The Physical Domain

We set D to be $\mathbb{T}^d = [0,1]^d_{\text{per}}$, the d dimensional unit torus; this will be the domain that we use for all our analysis. We need to introduce some mathematical concepts related to functions defined on this torus $\mathbb{T}^d$. First, the space of square integrable functions on $\mathbb{T}^d$ with mean 0 is denoted by

$$\dot{L}^2(\mathbb{T}^d) := \left\{ v : \mathbb{T}^d \to \mathbb{R} : \int_{\mathbb{T}^d} |v(x)|^2 \, \mathrm{d}x < \infty, \int_{\mathbb{T}^d} v(x) \, \mathrm{d}x = 0 \right\}. \tag{6.2.1}$$

The L^2 inner product and norm are denoted by $[\cdot, \cdot]$ and $\| \cdot \|_0$ respectively.

In order both to define covariance operators and Sobolev spaces it is convenient to introduce the Laplacian operator. Let $-\Delta$ be the negative Laplacian equipped with periodic boundary conditions on $\mathbb{T}^d$ and restricted to functions with zero mean. This operator has orthonormal eigenfunctions $\phi_m(x) = e^{2\pi i \langle m, x \rangle}$ with corresponding eigenvalues $\lambda_m = 4\pi^2 |m|^2$, for every $m \in \mathbb{Z}^d \backslash \{0\}$, where $\mathbb{Z}^d$ denotes the d-fold tensor product of $\mathbb{Z}$, the set of non-negative integers. Here, i is the imaginary number, and $\langle m, x \rangle$ denotes the Euclidean inner product between $m, x \in \mathbb{R}^d$.

Now, we can write functions in $\dot{L}^2(\mathbb{T}^d)$ as Fourier series:

$$v(x) = \sum_{m \in \mathbb{Z}^d} \hat{v}(m) e^{2\pi i \langle m, x \rangle}, \tag{6.2.2}$$

where $\hat{v} : \mathbb{Z}^d \to \mathbb{R}$ is the Fourier coefficient that satisfies $\hat{v}(0) = 0$ and $\hat{v}(m) = [v, \phi_m]$ for $m \in \mathbb{Z}^d \backslash \{0\}$. This representation can be used to define useful Sobolev-like spaces. For every $t > 0$, the Sobolev-like space $\dot{H}^t(\mathbb{T}^d) \subset \dot{L}^2(\mathbb{T}^d)$ consists of functions with bounded $\| \cdot \|_t$ norm:

$$\|v\|_t^2 := \sum_{m \in \mathbb{Z}^d} (4\pi^2 |m|^2)^t |\hat{v}(m)|^2 < \infty. \tag{6.2.3}$$

We note that $\dot{H}^0(\mathbb{T}^d) = \dot{L}^2(\mathbb{T}^d)$. For $t < 0$, the space $\dot{H}^t(\mathbb{T}^d)$ is defined through duality. The Hilbert scale of function spaces defined through varying t serves as the basic ingredient to model the regularity of a function on $\mathbb{T}^d$.

6.2.1.2 The Matérn-like Kernel and Process

The Matérn-like covariance operator on the torus is defined by

$$C_\theta = \sigma^2 (-\Delta + \tau^2 I)^{-s}, \tag{6.2.4}$$

where the parameter $\theta = (\sigma, \tau, s)$. The roles of the three parameters are reviewed in Remark 6.2.2. The orthonormal eigenfunctions of this operator are $\phi_m(x) = e^{2\pi i \langle m, x \rangle}$ with corresponding eigenvalues $\sigma^2(4\pi^2|m|^2 + \tau^2)^{-s}$, for $m \in \mathbb{Z}^d \backslash \{0\}$.

The Matérn-like kernel function K_θ is related to the operator C_θ via

$$K_\theta(x, y) = [\delta(\cdot - x), C_\theta \delta(\cdot - y)], \tag{6.2.5}$$

where $\delta(\cdot - x)$ is the Dirac function centered at x. Equivalently, K_θ can be understood as the Green function of the differential operator C_θ^{-1}. Note that by Sobolev's emdedding theorem, $s > d/2$ is required to make $K_\theta(x, y)$ pointwise well-defined (See Section 7.1.3 and Lemma 7.2 in [63]): $K_\theta(\cdot, y)$ then lies in the space of continuous functions for any $y \in \mathbb{T}^d$.

Remark 6.2.1. *We also have the Mercer decomposition of the kernel function:*

$$K_\theta(x, y) = \sum_{m \in \mathbb{Z}^d \backslash \{0\}} \sigma^2(4\pi^2|m|^2 + \tau^2)^{-s} \phi_m(x) \phi_m^*(y), \tag{6.2.6}$$

where ϕ_m^* is the complex conjugate of ϕ_m.

Given these function spaces and operators, we can define the Matérn-like process using the Gaussian measure notation:

$$\xi \sim \mathcal{N}\left(0, \sigma^2(-\Delta + \tau^2 I)^{-s}\right). \tag{6.2.7}$$

This covariance operator viewpoint could be understood as follows: for any $f \in \dot{L}^2(\mathbb{T}^d)$, the quantity $[f, \xi]$ is a Gaussian random variable with mean 0 and variance $[f, \sigma^2(-\Delta + \tau^2 I)^{-s} f]$. We note that (6.2.7) is equivalent to the GP notation $\xi \sim \mathcal{GP}(0, K_\theta)$. For more details on how to define Gaussian measures using operators we refer to [30, 204]. A sample from this process can be realized by the Karhunen–Loève expansion

$$\xi(x) = \sum_{m \in \mathbb{Z}^d \backslash \{0\}} \sigma(4\pi^2|m|^2 + \tau^2)^{-s/2} \phi_m(x) \xi_m, \tag{6.2.8}$$

where ξ_m ($m \in \mathbb{Z}^d \backslash \{0\}$) are i.i.d. standard normal random variables; we have $\mathbb{E}\, \xi(x)\xi(y) = K_\theta(x, y)$. Numerically, we can draw a sample by truncating this series and restricting to a grid of values on the torus. Alternatively it is possible to discretize the differential operator C_θ^{-1} on a grid first, and then compute the discrete eigenfunctions to draw a sample. Such an idea is useful when the eigenvalues and

eigenfunctions of C_θ^{-1} are not analytically known a priori. Indeed, when the operator is discretized into a matrix, the infinite dimensional Gaussian measure becomes a finite dimensional one with the covariance matrix being the discretization of C_θ. Drawing samples is then straightforward. In this section, however, we work on the torus and so the eigenvalues and eigenfunctions are known explicitly and the truncated Karhunen–Loève expansion could be employed.

Remark 6.2.2. *The three parameters σ, τ and s quantify the amplitude, inverse lengthscale, and regularity of the process, respectively. This setting is similar to that of the standard Matérn process [258, 109], defined on the whole space $\mathbb{R}^d$, whose kernel function and associated covariance operator are both characterized by three parameters; see [166] for links to the solution of stochastic PDEs, an approach attributable to Whittle [289, 109]. The Matérn kernel function is*

$$K_{\sigma,l,\nu}(x,y) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{|x-y|}{l} \right)^\nu B_\nu \left(\frac{|x-y|}{l} \right),$$

for $x, y \in \mathbb{R}^d$, where B_ν is the modified Bessel function of the second kind of order ν. On $\mathbb{R}^d$, this kernel function corresponds to the covariance operator

$$C_{\sigma,l,\nu} = \frac{\sigma^2 l^d \Gamma(\nu + d/2)(4\pi)^{d/2}}{\Gamma(\nu)} (I - l^2\Delta)^{-\nu - d/2}.$$

From this formula, the connection between the Matérn covariance operator in $\mathbb{R}^d$ and the Matérn-like kernel operator (6.2.4) *on $\mathbb{T}^d$ becomes apparent. We restrict our analysis to the torus to exploit powerful Fourier series techniques. We will also comment on other boundary conditions in Subsection 6.2.7. For related results regarding the Matérn process in $\mathbb{R}^d$ or other bounded domains, we recommend the book [258]. We note that [258, Sec. 6.7] also considers a periodic version of the Matérn model and discusses (via the Fisher information matrix) the fixed domain asymptotics of the maximum likelihood estimate of the three parameters. By using the Mercer decomposition* (6.2.6)*, the periodic case there is mathematically equivalent to the Matérn-like model on the torus that is considered in this chapter. In the next subsection, we prove the consistency of estimators for the regularity parameter, providing a rigorous theory for this periodic model. It would be interesting, in future work, to combine this consistency with the properties of the Fisher information matrix established in [258, Sec. 6.7] to obtain Bernstein-von-Mises type theorems characterizing asymptotic normality of the estimator.*

6.2.2 Regularity Parameter Learning

With the Matérn-like kernel and process defined, we move to discuss the parameter learning problem in this subsection. We fix $\sigma = 1$ and $\tau = 0$ in the Matérn-like model and focus on the regularity parameter only. To proceed, we need to make precise the ground truth $u^\dagger$, the kernel, and the data location $\mathcal{X}$, of the learning problem.

6.2.2.1 The Ground Truth

Our theoretical results regarding the consistency of EB and KF estimators will be based on the assumption that $u^\dagger$ is drawn from the GP $\mathcal{N}(0, (-\Delta)^{-s})$ for some $s > d/2$.

Remark 6.2.3. *We note some regularity properties of this GP here. The Cameron-Martin space for $\xi \sim \mathcal{N}(0, (-\Delta)^{-s})$ is $\dot{H}^s(\mathbb{T}^d)$ (for readers not familiar with the Cameron-Martin space, see Theorem 7.33 in [63]). However, ξ is not an element of this space, almost surely. Indeed, it holds that ξ belongs to $\dot{H}^{s-d/2-\eta}(\mathbb{T}^d)$ for any $\eta > 0$ almost surely (and to Hölder spaces with the same number of fractional derivatives; see Theorem 2.12 in [63]). Furthermore, since the Laplacian operator is homogeneous and thus the covariance operator is stationary in space, the regularity of the path is spatially homogeneous (the measure is space translation-invariant). Here, we refer, for this phenomenon, to ξ (as a function) having* homogeneous critical regularity *$s - d/2$ across $\mathbb{T}^d$. If we drop the term "homogeneous", we mean the property holds without the requirement of spatial homogeneity. Such behavior may occur for functions with spatial singularities.*

Remark 6.2.4. *We always require $s > d/2$, which ensures the continuity of the sample path of ξ almost surely and guarantees that $\dot{H}^s(\mathbb{T})$ is a RKHS, according to discussions in Remark 6.2.3. Thus, the pointwise value of ξ makes sense.*

6.2.2.2 The Equidistributed Data

We observe equidistributed pointwise values of $u^\dagger$ over the torus, i.e., the data lie on a lattice. To describe the data locations we introduce a level parameter $q \in \mathbb{N}$ such that, for a given q, we have the data locations $\mathcal{X}_q := \{x_j : j \in J_q\}$, where $x_j = (j_1, j_2, ..., j_d) \cdot 2^{-q}$ and $J_q := \{(j_1, j_2, ..., j_d) \in \mathbb{N}^d : 0 \le j_k \le 2^q - 1, \forall\ 1 \le k \le d\}$. We also use the simplified notation $x_j = j2^{-q}$ throughout the chapter.

6.2.2.3 The EB and KF Estimators

We follow the definitions in Subsection 6.1.3. Here, the kernel function for the regularity learning problem will be

$$K_\theta(x,y) = [\delta(\cdot - x), (-\Delta)^{-t}\delta(\cdot - y)]\,,$$

where the parameter $\theta = \{t\}$. Similar to Remark 6.2.1, it has the following Mercer decomposition

$$K_\theta(x,y) = \sum_{m\in\mathbb{Z}^d\setminus\{0\}} (4\pi^2|m|^2)^{-s}\phi_m(x)\phi_m^*(y)\,. \tag{6.2.9}$$

Numerically, we can compute it by truncating this infinite series. Fast Fourier Transform could be applied to speed up computation of the kernel matrix.

We adapt several notations from Subsection 6.1.3 to this specific problem, by writing t instead of θ, and q instead of $\mathcal{X}_q$, and $K(t,q)$ instead of $K_\theta(\mathcal{X}_q,\mathcal{X}_q)$. These simplified notations make the analysis cleaner to present. Under such convention, the EB estimator for the regularity parameter is

$$s^{\mathrm{EB}}(q,u^\dagger) = \operatorname*{arg\,min}_{t\in[d/2+\delta,1/\delta]} \mathsf{L}^{\mathrm{EB}}(t,q,u^\dagger),\ \mathsf{L}^{\mathrm{EB}}(t,q,u^\dagger) := \|u(\cdot,t,q)\|_t^2 + \log\det K(t,q)\,. \tag{6.2.10}$$

Here, $u(\cdot,t,q)$ is the GPR solution using the kernel function K_t and the observational data of $u^\dagger$ at $\mathcal{X}_q$.

Remark 6.2.5. *The formula* (6.2.10) *is the continuous formulation of the EB loss function, which is more convenient for theoretical analysis of consistency. The finite-sample formula* (6.1.2) *is more useful in numerical computation, and it can be derived from* (6.2.10) *by using the representer theorem.*

Remark 6.2.6. *As in Remark 6.2.4, we require the regularity parameter $t > d/2$. Here, furthermore, we introduce a number $\delta > 0$ and select the domain of the parameter to be $t \in [d/2+\delta, 1/\delta]$; δ can be any arbitrary positive number, and this compactification of the parameter domain will simplify the subsequent analysis. The reader should not confuse real number δ with Dirac delta function δ.*

For the KF loss function, we fix the subsampling operator to be equidistributed subsampling so that $\pi\mathcal{X}_q = \mathcal{X}_{q-1}$; for this choice, we can omit the dependence of the

estimator on the subsampling operator π in the notation and write:

$$s^{\mathrm{KF}}(q, u^\dagger) = \underset{t \in [d/2+\delta, 1/\delta]}{\arg\min} \, \mathsf{L}^{\mathrm{KF}}(t, q, u^\dagger), \; \mathsf{L}^{\mathrm{KF}}(t, q, u^\dagger) := \frac{\|u(\cdot, t, q) - u(\cdot, t, q-1)\|_t^2}{\|u(\cdot, t, q)\|_t^2} \,. \tag{6.2.11}$$

6.2.3 Consistency and Implicit Bias

In this subsection, we present our theory of consistency and characterize the implicit bias via numerical experiments. The sketch of proofs is given in the next subsections.

6.2.3.1 Main Theorem

We have the following theorem regarding the consistency of the two statistical estimators in the large data limit:

Theorem 6.2.7. *Fix $\delta > 0$. Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0, (-\Delta)^{-s})$. If $s \in [d/2 + \delta, 1/\delta]$ then, for the Empirical Bayesian estimator,*

$$\lim_{q \to \infty} s^{\mathrm{EB}}(q, u^\dagger) = s \,;$$

if $\frac{s-d/2}{2} \in [d/2 + \delta, 1/\delta]$ then for the Kernel Flow estimator,

$$\lim_{q \to \infty} s^{\mathrm{KF}}(q, u^\dagger) = \frac{s - d/2}{2} \,.$$

In both cases the convergence is in probability with respect to randomly chosen $u^\dagger$.

Remark 6.2.8. *Strictly speaking this theorem shows that EB consistently estimates the regularity parameter, whilst KF does not. However we make two observations about this. Firstly, the true value of s can be recovered from the KF estimator by a simple linear transformation. And, secondly, the value selected by KF is optimal with respect to minimizing a specific measure of generalization error (as we will show in the discussion of implicit bias in Subsection 6.2.3.3), and is of clear interest from this perspective.*

Remark 6.2.9. *The use of δ in the proof (and hence statement) of this theorem helps by compactifying the parameter space. In practice, numerics demonstrate that it is not intrinsic to the problem. We leave for future work the problem of a more refined theorem, and proof, which does not rely on it.*

Remark 6.2.10. *For economy of notation we will drop explicit reference to the dependence of the loss functions and the estimators on $u^\dagger$ in what follows; we will simply write $\mathsf{L}^{\mathrm{EB}}(t,q)$, $\mathsf{L}^{\mathrm{KF}}(t,q)$, $s^{\mathrm{EB}}(q)$, $s^{\mathrm{KF}}(q)$.*

The remainder of this subsection is devoted to numerical experiments illustrating the theory, discussion of the implications of the theory (i.e. implicit bias), and an overview of the proof techniques we adopt.

6.2.3.2 Numerical Illustration of Theory

We present a numerical example to demonstrate the main theorem, and its consequences for regression. Consider the one dimensional case, i.e., $d = 1$. We set the ground truth $s = 2.5$ and so $\frac{s-d/2}{2} = 1$. The domain is discretized with $N = 2^{10}$ equidistributed grid points. For our first set of experiments we fix the resolution level of the data points to be $q = 9$, i.e., we have 2^9 equidistributed observations of the unknown function $u^\dagger$. In what follows the Laplacian is as defined in Subsection 6.2.1.2. Given a sample of $u^\dagger$ from $\mathcal{N}(0,(-\Delta)^{-s})$, we form the loss function for the EB and the KF estimators. We draw this sample using the formula (6.2.8) with $\sigma = 1$ and $\tau = 0$; we truncate the series to the grid resolution. A single realization of these loss functions is then shown in Figure 6.1.

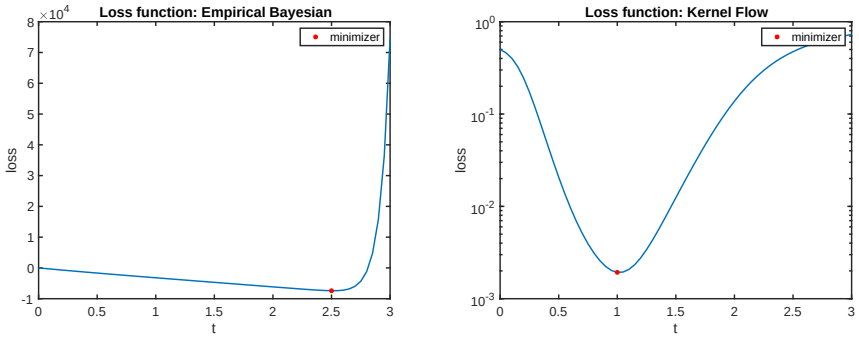


Figure 6.1: Left: EB loss; right: KF loss

We observe that the minimizer of the EB loss function is very close to $t = 2.5$, while the minimizer of the KF loss function is very close to $t = 1$, matching the predictions of Theorem 6.2.7. Furthermore, the loss functions exhibit some interesting features. Specifically, the EB loss function behaves as a linear function of t, for t less than s, and then blows up rapidly when t exceeds s. The KF loss function is more symmetric with respect to the minimizer $t = \frac{s-d/2}{2}$ in the logarithmic scale. We will make remarks that explain these observations in our theoretical analysis.

6.2.3.3 Implicit Bias

We present here a second set of numerical experiments looking at the effect of the parameter value s selected by EB and KF on the approximation of the function $u^\dagger$, which is (typically) the primary goal of hierarchical parameter estimation. The experimental set-up is the same, but now we vary the resolution of the data points $q = 3, 4, ..., 9$. We focus on the L^2 error between $u^\dagger$ and the GPR solution using learned parameters, i.e.,

$$\|u^\dagger(\cdot) - u(\cdot, t, q)\|_0^2 .$$

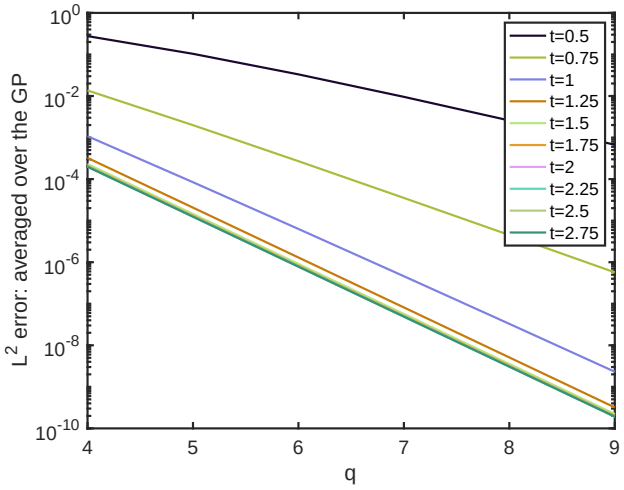


Figure 6.2: L^2 error: averaged over the GP.

We start, in Figure 6.2, by considering the error as a function of q, for different t. As we increase t, the regularity of the GP used for regression increases. In order to illustrate clear trends, the L^2 error is averaged over the random draw of $u^\dagger \sim \mathcal{N}(0, (-\Delta)^{-s})$, so the effective error is $\mathbb{E}_{u^\dagger}\|u^\dagger(\cdot) - u(\cdot, t, q)\|_0^2$. From the figure, we can see that when t increases from 0.5 to 1, the convergence rate of the L^2 approximation error increases. Then, if we increase t further from 1 to 3, the slope of the convergence curve remains nearly the same. This demonstrates the fact that $1 = \frac{s-d/2}{2}$ is the minimal t that suffices to achieve the fastest rate of L^2 error convergence. We have observed that this phenomenon is very stable with respect to the specific random draw: the general shape of the curves seen in Figure 6.2 is still observed when one specific draw of the true random process is used, although the resulting figure contains fluctuations and is not as clear as the average case that we show.

On the other hand, we can compute $\mathbb{E}_{u^\dagger}\|u^\dagger(\cdot) - u(\cdot, t, q)\|_0^2$ for $q = 9$ as a function of t; see Figure 6.3. The optimality of the value $s = 2.5$ is clear. However, unlike the experiments in Figure 6.2, this result is not stable with respect to the random instance of the GP: the minimizer of the L^2 error fluctuates wildly in our experiments.

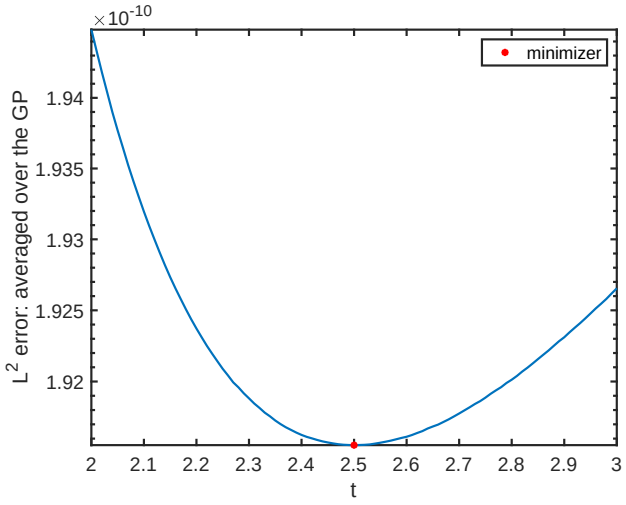


Figure 6.3: L^2 error: averaged over the GP, for $q = 9$

In summary, the second set of numerical experiments indicates the following implications for the regression accuracy of the EB and KF approaches to hierarchical parameter estimation. The KF estimator selects the minimal t that suffices to achieve the fastest rate of approximation error in the L^2 norm for a given fixed truth; in contrast, the EB estimator converges to the t that achieves the minimal L^2 error, averaged over the draw $u^\dagger \in \mathcal{N}(0, (-\Delta)^{-s})$. Note that KF is based on purely approximation theoretic considerations whilst EB is founded on statistical considerations—they attain very different implicit bias in selecting parameters.

6.2.3.4 Further Discussion of The Theory

We provide some further discussions of the implications of Theorem 6.2.7 in this subsection. The theory shows that the EB estimator recovers the ground truth parameter s of the statistical model. This is in line with expectations since the methodology is designed to recover the most likely value of s, given the data, and since the Gaussian measures occurring for different s are mutually singular. In the literature, such consistency results are primarily for observational data in the Fourier domain; thus, the observation operator commutes with the prior. Here, our data model is in the physical domain, which leads to the need for considerably more

sophisticated analysis, due to the noncommutativity of the observation operator and the prior operator, and yet is a much more practically useful setting, justifying the investment in the somewhat involved analysis. Our proof provides a novel sharp upper and lower bound on the terms $\|u(\cdot,t,q)\|_t^2$ and $\log\det K(t,q)$, based on techniques in approximation theory and the multiresolution analysis developed in [204]. Our techniques may have broader applications in analyzing the observational model in the physical domain.

Another interesting phenomenon shown in Theorem 6.2.7 is that the KF estimator, first proposed in [206] as a method to learn kernels for machine learning tasks, achieves a rather different consistency behavior, with the large data limit being $\frac{s-d/2}{2}$. This fact has the following consequence: if the ground truth function $u^\dagger$ has homogeneous critical regularity $s-d/2$, then the KF estimator will converge to half the critical regularity in the large data limit.

To understand the mechanism behind this effect, we observe that the KF loss is a surrogate for the (relative) $\|\cdot\|_t$-norm approximation error between $u^\dagger$ and $u(\cdot,t,q)$. Furthermore, approximation theory implies that the GP regressor $u(\cdot,t,q)$ is also the optimal $\|\cdot\|_t$-norm approximant of $u^\dagger$ in the linear span of the basis functions $\{(-\Delta)^{-t}\delta(x-x_j)\}_{j\in J_q}$. Under this perspective, we see the KF loss incorporates two competing factors in the approximation: increasing t improves the approximation error by increasing the regularity of the basis functions while worsening the measurement of that approximation error by using a stronger norm. The balance between these two competing factors is achieved when t is half the critical regularity, which is the parameter that KF eventually picks. Our proof provides a detailed demonstration of this phenomenon.

In short, EB learns hierarchically based on statistical principles, whilst KF learns based on approximation theoretic ones. The consistency results presented here provide evidence that the interplay between statistical estimation and numerical approximation can be very useful for parameter estimation and kernel learning in general, thus suggesting new ways of thinking hierarchically. This perspective is one of the main messages that we convey in this work.

6.2.3.5 Proof Strategy

The following Subsections 6.2.4, 6.2.5, and 6.2.6 are devoted to proving the above Theorem 6.2.7. For the sake of understanding, we provide a high-level view of our proof strategies in this subsection. Fourier analysis plays an important role in the

proof. It allows us to analyze the approximation error in a very precise way under this equidistributed design setting.

In our proof, we begin by establishing tight bounds on the terms that appear in the objective functions, i.e., $\|u(\cdot,t,q)\|_t^2$, $\log\det K(t,q)$ and $\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2$, using the toolkit we develop in Subsection 6.2.4. The norms $\|u(\cdot,t,q)\|_t^2$ and $\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2$ are expressed as random (as a function of $u^\dagger$) series and we carefully analyze the dependencies of the random variables to establish the convergence in probability. For $\log\det K(t,q)$, we employ the multiresolution approach introduced in [204] to establish a tight estimate of the spectrum of the Gram matrix from below and above. Given these estimates, we provide an intuitive understanding of how the loss functions behave and how the minimizers converge in Subsections 6.2.5, 6.2.6. In the rigorous treatment, the sharp bounds on the different components of the objective functions will be combined with the uniform convergence result of random series in [277] to obtain the convergence of minimizers.

6.2.3.6 Notations

In many parts of the analysis, we need to develop tight estimates on the terms appearing in the loss functions. Some useful notation for comparing different terms are introduced here. We write $A \simeq B$ if there exists a constant C independent of q,t such that

$$\frac{1}{C}B \le A \le CB\,.$$

The constant may depend on the dimension d and on δ. Correspondingly, if we use $A \gtrsim B$ or $A \lesssim B$, then only one side of the above inequality holds.

Fourier analysis plays a critical role in the analysis. We always use $u^\dagger$ for the ground truth function, while we omit the $\dagger$ symbol for ease of notation when discussing its Fourier transform, and write $\hat{u}$; we will also use $\hat{u}$, with more arguments, to denote the Fourier transform of the Gaussian process mean; see the discussion following Theorem 6.2.13. In the Fourier domain, we let $B_q := \{m \in \mathbb{Z} : -2^{q-1} \le m \le 2^{q-1}-1\}$ and $B_q^d = B_q \otimes B_q \otimes \cdots \otimes B_q$ be the tensor product of d multiples of B_q. We have that B_q^d is a box concentrating around the origin, so only the low-frequency part of the Fourier coefficients are considered.

6.2.4 Toolkit: Fourier Series Characterization

In this subsection, we prepare the necessary tools that are used to prove the main theorem of this chapter.

We start by establishing a Fourier series characterization for $u(\cdot, t, q)$. This is a key ingredient in expressing the terms in the loss functions as random series. Our approach, using Fourier series, is motivated by the papers [65, 231], where the approximation power of shift-invariant subspaces of $L^2(\mathbb{R}^d)$ is studied; in our case we use related ideas in the $\dot{L}^2(\mathbb{T}^d)$ setting.

To find the representation of the term $u(\cdot, t, q)$, we invoke its definition, i.e. $u(\cdot, t, q)$ is obtained by GP regression with the q-level data and the covariance function $(-\Delta)^{-t}$. We use the representer theorem from GPR. Concretely, let the set of basis functions be

$$\mathcal{F}_{t,q} = \operatorname{span}_{j \in J_q} \{(-\Delta)^{-t} \delta(\cdot - x_j)\},$$

then, $u(\cdot, t, q)$ is the best approximation in $\mathcal{F}_{t,q}$ to the true function under the $\|\cdot\|_t$ norm. Let us define

$$\hat{\mathcal{F}}_{t,q} := \{g : \mathbb{Z}^d \to \mathbb{C}, \text{there exists an } f \in \mathcal{F}_{t,q} \text{ such that } g = \hat{f}\},$$

the Fourier coefficients of functions in $\mathcal{F}_{t,q}$. A quick observation is that for every $g \in \hat{\mathcal{F}}_{t,q}$, we must have $g(0) = 0$ because of the mean zero property of $f \in \mathcal{F}_{t,q}$. The following proposition gives a complete characterization of the basis functions in $\hat{\mathcal{F}}_{t,q}$, for $t > d/2$.

Proposition 6.2.11. *For any $g \in \hat{\mathcal{F}}_{t,q}$, there exists a 2^q-periodic function p on $\mathbb{Z}^d$, such that*

$$g(m) = \begin{cases} |m|^{-2t} p(m), & m \neq 0 \\ 0, & m = 0\,. \end{cases}$$

The proof is in Subsection C.1.1. Next, we define a 2^q-periodization operator, which will be used to compute the representation of $\hat{u}(m, t, q)$.

Definition 6.2.12. *The operator T_q is defined as a mapping from the space of functions on $\mathbb{Z}^d$ to itself, such that*

$$(T_q g)(m) := \sum_{\beta \in \mathbb{Z}^d} g(m + 2^q \beta), \quad m \in \mathbb{Z}^d\,,$$

whenever the right hand side series converges for the function $g : \mathbb{Z}^d \to \mathbb{R}$. We also define

$$M_q^t(m) := \begin{cases} \sum_{\beta \in \mathbb{Z}^d \setminus \{0\}} |2^q \beta|^{-2t}, & \text{if } m = j \cdot 2^q \text{ for some } j \in \mathbb{Z}^d \\ \sum_{\beta \in \mathbb{Z}^d} |m + 2^q \beta|^{-2t}, & \text{else}\,. \end{cases} \tag{6.2.12}$$

Both $T_q g$ and M_q^t are 2^q-periodic functions on $\mathbb{Z}^d$. Based on this definition, Theorem 6.2.13 presents the explicit form of the Fourier transform of $u(\cdot, t, q)$; the proof is in Subsection C.1.2. The proof relies on the Galerkin orthogonality property of $u(\cdot, t, q)$ due to its being the optimal approximate solution.

Theorem 6.2.13. *Let $\hat{u}(\cdot, t, q)$ be the Fourier coefficients of $u(\cdot, t, q)$, then for $m \in \mathbb{Z}^d$, we have*

$$\hat{u}(m, t, q) = \begin{cases} 0, & \text{if } m = 0 \\ |m|^{-2t} \frac{(T_q \hat{u})(m)}{M_q^t(m)}, & \text{else} \end{cases}$$

where $\hat{u}$ denotes the Fourier coefficients of $u^\dagger$.

This above representation is very useful for analyzing the terms $\|u(\cdot, t, q)\|_t^2$ and $\|u(\cdot, t, q) - u(\cdot, t, q-1)\|_t^2$. As well as studying the Fourier coefficients of $u(\cdot, t, q)$, which we denote by $\hat{u}(\cdot, t, q)$, we will also need to study the Fourier coefficients of $u^\dagger(\cdot)$ which, for ease of notation we will denote by $\hat{u}(\cdot)$, henceforth, omitting the $\dagger$ symbol. It is thus important to look at the number of arguments of $\hat{u}$ to determine which object it is the Fourier transform of. Note also that $u(\cdot, t, q)$ is determined by $u^\dagger$; hence if $u^\dagger$ is random, so is $u(\cdot, t, q)$.

We will use the above Fourier analysis toolkit to study the consistency of EB and KF in the following two subsections.

6.2.5 Proof for the Empirical Bayesian Estimator

In this subsection, we prove the consistency of the EB estimator. As explained before, our roadmap is to give a tight estimate of the loss functions first and then analyze the minimizers. For the norm term $\|u(\cdot, t, q)\|_t^2$, we invoke Theorem 6.2.13, based on which this term is expressed as a random series:

Proposition 6.2.14. *The $\dot{H}^t(\mathbb{T}^d)$ norm of $u(\cdot, t, q)$ has the representation*

$$\|u(\cdot, t, q)\|_t^2 = (4\pi^2)^t \sum_{m \in B_q^d} \frac{|T_q \hat{u}(m)|^2}{M_q^t(m)} .$$

Moreover, suppose $u^\dagger \sim \mathcal{N}(0, (-\Delta)^{-s})$ for $s > \frac{d}{2}$, then

$$\|u(\cdot, t, q)\|_t^2 = (4\pi^2)^{t-s} \sum_{m \in B_q^d} \frac{M_q^s(m)}{M_q^t(m)} \xi_m^2 ,$$

where $\{\xi_m\}_{m \in B_q^d}$ are independent unit scalar Gaussian random variables.

Proof. Using Theorem 6.2.13, we get

$$\begin{aligned}\|u(\cdot,t,q)\|_t^2 =& \sum_{m\in\mathbb{Z}^d\setminus\{0\}} (4\pi^2)^t |m|^{2t} |\hat{u}(m,t,q)|^2 \\ =&(4\pi^2)^t \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t} \frac{|T_q\hat{u}(m)|^2}{|M_q^t(m)|^2} \\ =&(4\pi^2)^t \sum_{m\in B_q^d} M_q^t(m) \frac{|T_q\hat{u}(m)|^2}{|M_q^t(m)|^2} \\ =&(4\pi^2)^t \sum_{m\in B_q^d} \frac{|T_q\hat{u}(m)|^2}{M_q^t(m)},\end{aligned}$$

where in the third equality, we use the periodicity of the function $\frac{|T_q\hat{u}(m)|^2}{|M_q^t(m)|^2}$.

If we further assume $u^\dagger \sim \mathcal{N}(0,(-\Delta)^{-s})$, then $\hat{u}(m) \sim \mathcal{N}(0,(4\pi^2)^{-s}|m|^{-2s})$. For different m, these Gaussian random variables are independent. Thus, for different $m \in B_q^d$, we have $T_q\hat{u}(m) \sim \mathcal{N}(0,(4\pi^2)^{-s}M_q^s(m))$, and they are independent. So we can write

$$\sum_{m\in B_q^d} \frac{|T_q\hat{u}(m)|^2}{M_q^t(m)} = (4\pi^2)^{-s} \sum_{m\in B_q^d} \frac{M_q^s(m)}{M_q^t(m)} \xi_m^2,$$

where $\{\xi_m\}_{m\in B_q^d}$ are independent unit scalar Gaussian random variables. □

The independence of the random variables established in the preceding representation is crucial for the analysis. The terms $M_q^s(m), M_q^t(m)$ appear in the preceding; to analyze them we present a useful lemma below. The proof is in Subsection C.1.3.

Lemma 6.2.15. *For $t \in [d/2+\delta, 1/\delta]$ and $q \geq 0$, we have*

$$M_q^t(m) \simeq \begin{cases} 2^{-2qt}, \text{ if } m = 0 \\ |m|^{-2t}, \text{ if } m \in B_q^d\setminus\{0\} \end{cases}$$

Moreover, for $m \in B_q^d\setminus\{0\}$, we have $M_q^t(m) - |m|^{-2t} \simeq 2^{-2qt}$.

Now, we are ready to get the estimates of the loss function. The following proposition shows an upper and lower bound on the norm term.

Proposition 6.2.16 (Bound on the norm term)**.** *Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0,(-\Delta)^{-s})$ for $d/2+\delta \leq s \leq 1/\delta$, then*

$$\|u(\cdot,t,q)\|_t^2 \simeq 2^{-q(2s-2t)}\xi_0^2 + \sum_{m\in B_q^d\setminus\{0\}} |m|^{2t-2s}\xi_m^2,$$

where $\{\xi_m\}_{m\in B_q^d}$ *are independent unit scalar Gaussian random variables.*

Proof. According to Lemma 6.2.15, for $m \in B_q^d\backslash\{0\}$, we have $M_q^t(m) \simeq |m|^{-2t}$; for $m = 0$, we have $M_q^t(m) \simeq 2^{-2tq}$. Thus,

$$\begin{aligned}\|u(\cdot,t,q)\|_t^2 &= (4\pi^2)^{t-s}\sum_{m\in B_q^d}\frac{M_q^s(m)}{M_q^t(m)}\xi_m^2\\ &= (4\pi^2)^{t-s}\left(\sum_{m\in B_q^d\backslash\{0\}}\frac{M_q^s(m)}{M_q^t(m)}\xi_m^2+\frac{M_q^s(0)}{M_q^t(0)}\xi_0^2\right)\\ &\simeq 2^{-q(2s-2t)}\xi_0^2+\sum_{m\in B_q^d\backslash\{0\}}|m|^{2t-2s}\xi_m^2\,.\end{aligned}$$

This completes the proof. □

Proposition 6.2.16 states that the behavior of the norm term is nothing but a weighted sum of squares of independent Gaussian random variables, which is amenable to analysis. With this in mind, we state a lemma useful in the analysis of such random series, with proof deferred to Subsection C.1.4.

Lemma 6.2.17. *Suppose* $\{\xi_m\}_{m\in\mathbb{Z}^d}$ *are independent unit Gaussian random variables.*

- *For* $r > 0$*, define the random series*

$$\alpha(r,q) = 2^{-qr}\sum_{m\in B_q^d\backslash\{0\}}|m|^{r-d}\xi_m^2\,.$$

  Fix $\epsilon > 0$*, then there exists a function* $\gamma(r) > 0$ *such that* $\lim_{q\to\infty}\alpha(r,q) = \gamma(r) > 0$ *uniformly for* $r \in [\epsilon, 1/\epsilon]$*, where the convergence is in probability.*

- *For* $r = 0$*, define*

$$\alpha(0,q) = \frac{1}{q}\sum_{m\in B_q^d\backslash\{0\}}|m|^{-d}\xi_m^2\,,$$

  then there exists $\gamma(0) \in (0,\infty)$ *such that* $\lim_{q\to\infty}\alpha(0,q) = \gamma(0)$ *in probability.*

We then move to the second term in the loss function, i.e., the log determinant term. It is deterministic and to study it we need a way of analyzing the spectrum of the Gram matrix. The following Proposition 6.2.18 gives upper and lower bounds on

this term. The proof is in Subsection C.1.5 and is motivated by analysis developed in the paper [204]. The idea is to use the Schur complement of the Gram matrix and rely on the variational characterization of the Schur complement to get a tight control on the spectrum. This technique is quite general and has been used in [204] to characterize the spectrum of heterogeneous Laplacian operators; here we adapt it to fractional operators. On the other hand, for the homogeneous fractional Laplacian operators in this chapter, it is also possible to calculate an explicit formula for the spectrum of $K(t,q)$, as has been used in Section 6.7 of [258]. We describe this simple proof in Subsection C.1.5 but retain the proof employing the more general methodology as it may be useful for other problems.

Proposition 6.2.18 (Bound on the log det term)**.** *For $d/2+\delta \le t \le 1/\delta$, we have*

$$(2t-d)g_1(q) - Cg_2(q) + K(t,0) \le \log\det K(t,q) \le (2t-d)g_1(q) + Cg_2(q) + K(t,0),$$

where $g_1(q) = \sum_{k=1}^{q}(2^{kd} - 2^{(k-1)d})(-k\log 2)$ and $g_2(q) = (2^{qd}-1)(2t-d)$. The constant C is independent of t, q. Moreover, $g_1(q) \simeq -q2^{qd}$.

With the loss function analyzed by the above results, the consistency of the EB estimator is readily stated as follows.

Theorem 6.2.19 (Consistency of Empirical Bayesian estimator)**.** *Fix $\delta > 0$. Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0,(-\Delta)^{-s})$. If $s \in [d/2+\delta, 1/\delta]$ then*

$$\lim_{q\to\infty} s^{\mathrm{EB}}(q) = s \quad \text{in probability}.$$

The detailed proof is in Subsection C.1.6. We can understand the theorem intuitively by using the established results above. Recall there are two terms in the loss function: (1) the norm term $\|u(\cdot,t,q)\|_t^2$; (2) the log det term. For the norm term, from Proposition 6.2.16 and Lemma 6.2.17, its behavior for $q \to \infty$ is roughly

- Growing like $2^{q(2t-2s+d)}$ if $t > s - d/2$;
- Growing like q if $t = s - d/2$;
- Remaining bounded if $t < s - d/2$.

The log det term decreases like $-(2t-d)q2^{qd}$ according to Proposition 6.2.18. Noticing that the EB loss function has the form

$$\mathsf{L}^{\mathrm{EB}}(t,q) = \|u(\cdot,t,q)\|_t^2 + \log\det K(t,q),$$

we arrive at the following intuitive observations:

- When $t < s$, the dominant behavior of $\mathsf{L}^{\mathrm{EB}}(t,q)$ is controlled by the log determinant term, since the growth rate of the norm term $2^{q(2t-2s+d)} = o(q2^{qd})$. As a consequence, $\mathsf{L}^{\mathrm{EB}}(t,q)$ exhibits the overall behavior $-(2t-d)q2^{qd}$. Therefore, the loss function decreases linearly with t in this regime. This is consistent with what is observed in Figure 6.1.

- When $t \geq s$, the increasing speed of the norm term beats the decreasing rate of the log det term, so the norm term dominates the behavior of $\mathsf{L}^{\mathrm{EB}}(t,q)$. Overall, it is like $2^{q(2t-2s+d)}$, which increases exponentially with t; again this is consistent with what is observed in Figure 6.1.

According to the above observations, the minimizer of $\mathsf{L}^{\mathrm{EB}}(t,q)$ will converge to s. To make the intuition leading to this conclusion rigorous, we need to use techniques of uniform convergence for random series. For details we refer to Subsection C.1.6.

6.2.6 Proof for the Kernel Flow Estimator

In this subsection, we establish the consistency of the KF estimator. As before, we start by estimating the growth behavior of terms that appear in the loss function. We begin with the interaction term $\|u(\cdot,t,q) - u(\cdot,t,q-1)\|_t^2$. Similar to the analysis of the norm term in the preceding subsection, we represent it by using Fourier series.

Proposition 6.2.20. *The $\dot{H}^t(\mathbb{T}^d)$ norm of $u(\cdot,t,q)-u(\cdot,t,q-1)$ has the representation*

$$\|u(\cdot,t,q) - u(\cdot,t,q-1)\|_t^2 = (4\pi^2)^t \sum_{m\in B_q^d} M_q^t(m) \left(\frac{T_q\hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \right)^2 . \quad (6.2.13)$$

Proof. By Theorem 6.2.13, we have

$$\hat{u}(m,t,q) - \hat{u}(m,t,q-1) = \begin{cases} 0, & \text{if } m = 0 \\ |m|^{-2t} \left(\frac{T_q\hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \right), & \text{else} . \end{cases}$$

Thus,

$$\begin{aligned}\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2 =&(4\pi^2)^t \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{2t}|\hat{u}(m,t,q)-\hat{u}(m,t,q-1)|^2\\ =&(4\pi^2)^t \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t}\left(\frac{T_q\hat{u}(m)}{M_q^t(m)}-\frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)}\right)^2\\ =&(4\pi^2)^t \sum_{m\in B_q^d} M_q^t(m)\left(\frac{T_q\hat{u}(m)}{M_q^t(m)}-\frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)}\right)^2.\end{aligned}$$

□

By carefully studying the correlation between the random variables appearing in the preceding proposition, we obtain lower and upper bounds in the following two propositions; proofs can be found in Subsections C.1.7 and C.1.8.

Proposition 6.2.21 (Lower bound on the interaction term)**.** *Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0,(-\Delta)^{-s})$ for $d/2+\delta\le s\le 1/\delta$, then*

$$\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2 \gtrsim \sum_{m\in B_{q-1}^d\setminus\{0\}} 2^{-2tq}|m|^{4t-2s}\xi_m^2,$$

where $\{\xi_m\}_{m\in B_{q-1}^d\setminus\{0\}}$ are independent unit scalar Gaussian random variables.

The upper bound has a more complex form. We introduce the notation $\mathbb{Z}_2^d=\{0,1\}^d$ comprising d dimensional vectors with each component being in $\{0,1\}$. In the following proposition, we also use the convention that $|m|^\alpha=0$ for $m=0$ and any $\alpha\in\mathbb{R}$ to make the notation more compact.

Proposition 6.2.22 (Upper bound on the interaction term)**.** *Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0,(-\Delta)^{-s})$ for $d/2+\delta\le s\le 1/\delta$, then*

$$\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2 \lesssim \sum_{k\in\mathbb{Z}_2^d}\sum_{m\in B_{q-1}^d} (2^{-q(2s-2t)}+2^{-2tq}|m|^{4t-2s})\xi_{k,m}^2,$$

where for a fixed $k\in\mathbb{Z}_2^d$, $\{\xi_{k,m}\}_{m\in B_{q-1}^d}$ are independent unit scalar Gaussian random variables.

We remark that in the upper bound, the random variables for different k may exhibit correlation. However, since the term $\sum_{m\in B_{q-1}^d}(2^{-q(2s-2t)}+2^{-2tq}|m|^{4t-2s})\xi_{k,m}^2$ has the same form for each k, and the number of different k is finite, it suffices to analyze the random series for a single k, in which we have the independence of random variables. The theorem is stated below.

Theorem 6.2.23 (Consistency of the Kernel Flow estimator)**.** *Fix $\delta > 0$. Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0, (-\Delta)^{-s})$. If $\frac{s-d/2}{2} \in [d/2+\delta, 1/\delta]$ then for the Kernel Flow estimator,*

$$\lim_{q\to\infty} s^{\mathrm{KF}}(q) = \frac{s-d/2}{2} \quad \text{in probability}\,.$$

The idea behind the proof of the theorem is to combine Propositions 6.2.21, 6.2.22 and Lemma 6.2.17. Together they imply the growth behavior of the loss function

$$\mathsf{L}^{\mathrm{KF}}(t,q) = \frac{\|u(\cdot,t,q) - u(\cdot,t,q-1)\|_t^2}{\|u(\cdot,t,q)\|_t^2}$$

as follows:

- When $t < \frac{s-d/2}{2}$, the numerator decays like 2^{-2tq} since $4t - 2s < -d$, in which case the summation $\sum_{m\in B_q^d\setminus\{0\}} |m|^{4t-2s}\xi_m^2$ remains bounded. The denominator remains bounded. So the overall behavior is 2^{-2tq}.
- When $\frac{s-d/2}{2} < t < s - d/2$, the numerator decays like $2^{-2tq} \times 2^{q(4t-2s+d)} = 2^{q(2t-2s+d)}$ according to Lemma 6.2.17. The denominator remains bounded, The overall behavior is $2^{q(2t-2s+d)}$.
- When $t > s - d/2$, the numerator behaves like $2^{q(2t-2s+d)}$, while the denominator behaves like $2^{q(2t-2s+d)}$. The overall behavior is of order 1.

These observations are consistent with what is observed in Figure 6.1. Based on them we deduce that the minimizer converges to $\frac{s-d/2}{2}$. The loss function exhibits symmetric behavior with respect to $\frac{s-d/2}{2}$ for $t \in (d/2, s-d)$. The detailed rigorous treatment is presented in Subsection C.1.9.

6.2.7 Discussions

In the preceding three subsections, we have presented the consistency theory, its implication for implicit bias, as well as the tools and strategies underlying our proofs. This subsection adds to several discussions on the theory and proofs.

First, our theory applies to the torus domain. One may wonder whether these techniques can be applied to boundary conditions beyond the periodic ones. The main tool used in the proofs is Fourier's series (based on the eigenfunctions of the Laplacian operator). These are used to characterize the norm term and determinant term. We expect these techniques to generalize to other problems, such as the

box with Dirichlet or Neumann boundary conditions in which the Fourier sine or cosine series are natural; the detailed analysis is left as future work. However, we need to point out that the limitation of this proof idea is that it requires a clear analytic understanding of the spectral properties of the kernel operator, i.e., its eigenfunctions. In Subsection 6.3.2.1, we present numerical experiments beyond this setting, which involves more challenging Laplacians with discontinuous coefficients that can model more complicated heterogeneous random fields.

Second, this section considers the regularity parameter only. In spatial statistics literature, consistency results on this parameter (for general Matérn type model) are very scarce and difficult. Here, we obtain a proof for the torus model, which is the main technical contribution of this work. We will discuss the learning of other parameters in the next section, to make the story of the Matérn-like model on the torus more complete.

Finally, as we get two algorithms that can "consistently" learn the information of the regularity parameter when the number of data is large, a natural question is when to choose which. To answer this question, we presents numerical study of the variances of both estimators for the Matérn-like model in the next subsection.

6.2.8 Variance of Regularity Parameter Estimation

In this subsection, we compare the variance of the two estimators for recovering the regularity parameter s. We return to the experimental set-up in Subsection 6.2.3.2. We form the EB and KF estimators for 50 instances of different draws of the GP, normalized by the limiting optimum values s and $\frac{s-d/2}{2}$ respectively. The statistics of the two estimators are summarized in the histogram (see Figure 6.4). Clearly, EB

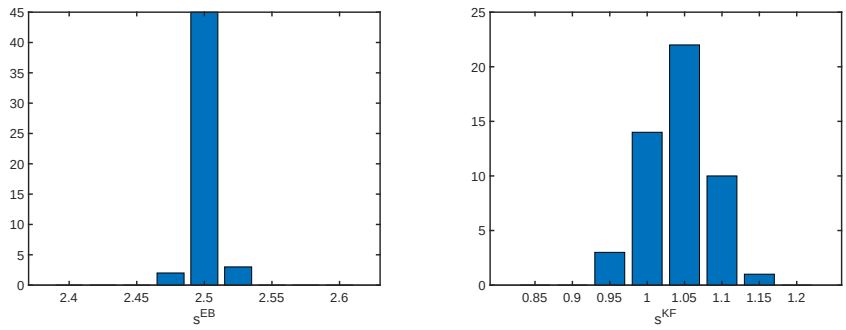


Figure 6.4: Histogram of the regularity estimators for the Matérn-like process. Left: EB; right: KF.

exhibits smaller variance than KF. We compute the estimated variance using the 50

instances. Finally we get

$$\frac{\mathrm{Var}(s^{\mathrm{EB}})}{s^2} \approx 1.44 \times 10^{-5} \quad \text{and} \quad \frac{\mathrm{Var}(s^{\mathrm{KF}})}{((s-d/2)/2)^2} \approx 3.6 \times 10^{-3}\,.$$

Since the variance of EB is smaller, if our target is to estimate s for the exact GP model, then this suggests that the EB method is preferable.

6.3 More Well-specified Examples

The setting in Section 6.2 concerns regularity parameter of the Matérn-like model only. This section aims to extend this discussion to a wider range of settings by means of numerical experiments. First, we study the learning of lengthscale and amplitude parameters in the Matérn-like model in Subsection 6.3.1; these experiments lead to a more complete story for the Matérn-like model on the torus. Then, in Subsection 6.3.2, we consider other well-specified models, extending beyond the Matérn-like process example. In Subsection 6.3.3, we also discuss some computational aspects of the EB and KF approaches.

6.3.1 Recovery of Amplitude and Lengthscale

We start with the learning of amplitude and lengthscale parameters in the Matérn-like model, via either EB or KF method.

In spatial statistics, an important general principle in looking at the recovery of hyperparameters via EB is to determine whether or not the family of measures are mutually singular with respect to changes in the parameter to be estimated; learning parameters which give rise to mutually singular families is usually easy, since different almost sure properties can often be used to distinguish measures and this can be achieved without an abundance of data; in contrast those parameters that do not give rise to mutually singular measures typically require an abundance of realizations to be accurately learned. We illustrate this issue in the context of estimating one parameter by EB, the changing of which leads to mutually singular measures, and estimating two parameters by EB, changing one of which leads to mutual singularity, and the other to equivalence, for the Matérn-like process. We also study analogous questions about identifiability for the KF method. In all cases we work with loss functions that are natural generalizations of (6.2.10), (6.2.11).

6.3.1.1 Recovery of σ

A first observation is that the KF loss function is invariant under change of σ, so it cannot recover this parameter. We also note that measures are mutually

singular with respect to changes in σ, and so we do expect to be able to recover σ by EB. For the EB estimator, we design the experiment as follows. We study whether the EB method can recover σ while s, τ are fixed. In detail, we consider a problem with domain the one dimensional torus $\mathbb{T}^1$. The Matérn-like kernel has regularity $s = 2.5$, amplitude $\sigma = 1$ and lengthscale $\tau = 0$. We assume the values of s, τ are known, but not σ. We want to recover σ by seeing a single discretized realization $u^\dagger \sim \mathcal{N}(0, \sigma^2(-\Delta + \tau^2 I)^{-s})$. The domain $\mathbb{T}^1$ is discretized into $N = 2^{10}$ equidistributed grid points. The data we observe is the values of $u^\dagger$ in 2^9 equidistributed points. We build the EB loss function (see equation (6.3.1)) and plot the figure for a single instance; see Figure 6.5. We introduce ς as the variable

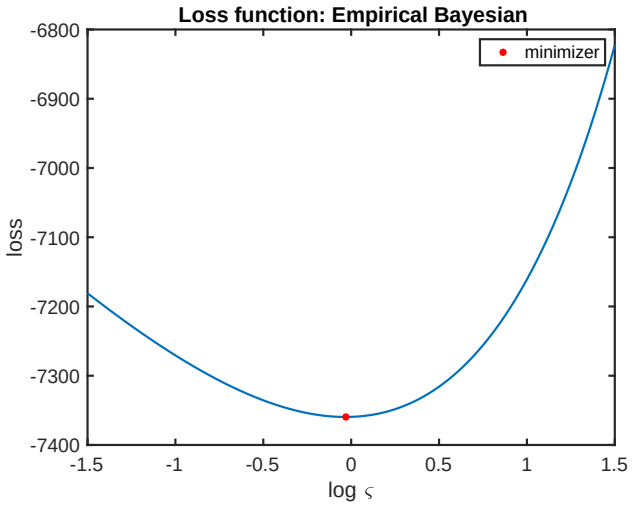


Figure 6.5: EB loss function for recovering σ

to be maximized over to determine our estimate of σ. In our experiments we work with the parameterization $\varsigma = \exp(\varsigma')$ in order to ensure that the estimated σ is positive. Hence, the x-axis of Figure 6.5 is ς'. The figure shows that the minimizer of the loss function is close to the point $\varsigma' = 0$ ($\varsigma = 1$), so the estimator σ^{EB} is close to the ground truth σ.

We can theoretically analyze the convergence. The same set-up in Subsection 6.2.1 is adopted, except now we assume the function is drawn from $\mathcal{N}(0, \sigma^2(-\Delta)^{-s})$ with s known and we want to recover σ by seeing the equidistributed spatial samples on the torus. After calculating the likelihood in such a case, we get the EB estimator

below. Here we abuse the notation to write

$$\begin{aligned}\sigma^{\mathrm{EB}}(q,u^\dagger) &= \arg\min_{\varsigma>0} \mathsf{L}^{\mathrm{EB}}(\varsigma,q,u^\dagger),\\ \mathsf{L}^{\mathrm{EB}}(\varsigma,q,u^\dagger) &:= \frac{\sigma^2\|u(\cdot,s,q)\|_s^2}{\varsigma^2} + \log\det K(s,q) + 2^{qd}\log\varsigma^2\,.\end{aligned} \tag{6.3.1}$$

The definition of $u(\cdot,s,q), K(s,q)$ is the same as in Subsection 6.2.1. Recall that $u(\cdot,s,q)$ is the mean of the GP found by conditioning a prior measure $\mathcal{N}(0,(-\Delta)^{-s})$ on observations of $u^\dagger$ at the observation data with level q. The definition of $\|\cdot\|_s$ also follows from Subsection 6.2.1. We abuse notation to write $\mathsf{L}^{\mathrm{EB}}(\varsigma,q,u^\dagger)$ for the EB loss function used in the estimation of σ; the reader should not confuse this with $\mathsf{L}^{\mathrm{EB}}(t,q,u^\dagger)$ in Subsection 6.2.1 which is used for recovering the regularity parameter s.

In this setting we have the following consistency result:

Theorem 6.3.1. *Fix $\delta > 0$. Suppose $u^\dagger$ is a sample drawn from the Gaussian process $\mathcal{N}(0,\sigma^2(-\Delta)^{-s})$ for some $s \in [d/2+\delta, 1/\delta]$. Then, for the Empirical Bayesian estimator of σ, it holds that*

$$\lim_{q\to\infty} \sigma^{\mathrm{EB}}(q,u^\dagger) = \sigma\,,$$

where the convergence is in probability with respect to randomly chosen $u^\dagger$.

Proof. By taking the derivative of $\mathsf{L}^{\mathrm{EB}}(\varsigma,q,u^\dagger)$ with respect to ς and setting it to 0, we get the explicit formula:

$$\sigma^{\mathrm{EB}}(q,u^\dagger) = \sigma\sqrt{\frac{\|u(\cdot,s,q)\|_s^2}{2^{qd}}}\,. \tag{6.3.2}$$

Due to Proposition 6.2.14, we get our $\|u(\cdot,s,q)\|_s^2 = \sum_{m\in B_q^d}\xi_m^2$. By the Law of Large Numbers, we have

$$\lim_{q\to\infty}\frac{\|u(\cdot,s,q)\|_s^2}{2^{qd}} = 1\,,$$

from which the consistency follows. □

Remark 6.3.2. *We note that consistency results for the amplitude parameter have been well studied in the literature; see [258]. The purpose of this subsection is to tie those results to the rather explicit setting of our result. One important feature of the torus model is that we are able to get an explicit and simple formula for σ^{EB}, so the consistency results are very clear. Moreover, since σ^{EB} is the average*

of i.i.d. Gaussian random variables, one can also easily read off other statistical properties of this estimator (although the result of asymptotic distribution is also not completely new; see for example the discussion on page 201 in [258]).

6.3.1.2 Recovery of s, σ simultaneously

We now build on the previous experiment to study whether the EB method can recover s, σ simultaneously when τ is fixed. We reemphasize that since the measures are mutually singular with respect to changes in σ and s we do expect to be able to recover (σ, s) by EB. The basic set-up is the same as the last subsection, and now we minimize the EB loss function to recover s, σ where, again, $\sigma = \exp(\sigma')$. We run 50 instances (each instance corresponds to a random draw of ξ), and collect the estimators $(s^{\mathrm{EB}}, \log \sigma^{\mathrm{EB}})$ of the EB loss function for each instance. We present the histogram of the two values obtained in the experiments as follows (Figure 6.6).

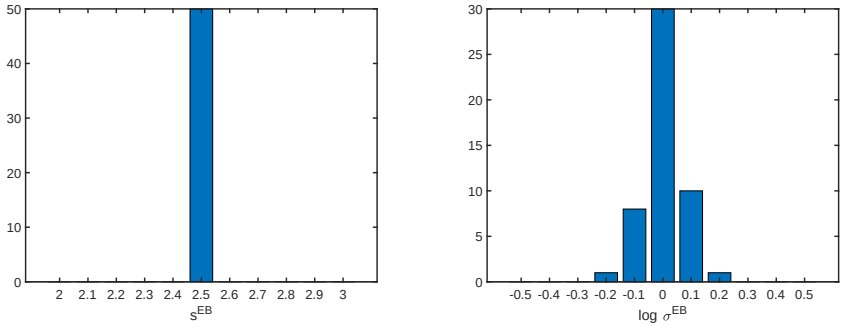


Figure 6.6: Left: histogram of the s^{EB}; right: histogram of the $\log \sigma^{\mathrm{EB}}$

From the figure, we observe that in the 50 runs, the minimizer $(s^{\mathrm{EB}}, \sigma^{\mathrm{EB}})$ is close to the ground truth $(2.5, 1)$. We conclude that the EB method can recover the two parameters simultaneously in such a context.

6.3.1.3 Recovery of τ

We consider whether EB and KF can recover the inverse lengthscale parameter τ. We assume that σ is fixed at 1, s is chosen to be 2.5, and sample $u^\dagger \sim \mathcal{N}(0, (-\Delta + \tau^2 I)^{-s})$ with $\tau = 1$. As in the preceding experiments we consider the one dimensional torus example, and the same discretization precision and data acquisition setting as before. We draw 50 instances of $u^\dagger$, and for each of them, calculate the minimizers of the EB and KF loss function. We write $\tau = \exp(\tau')$ and the estimator is $\log \tau^{\mathrm{EB}}$ for τ', which we constrain to be in the interval $[-2, 2]$. In the EB loss function we fix $t = s$

within the loss function; for the KF method, we select $t = s$ (case 1) and $t = \frac{s-d/2}{2}$ (case 2) respectively within the loss function. The histogram of the minimizers of the resulting EB loss function and KF loss functions (in both cases) are presented in Figure 6.7, expressed in terms of $\log \tau^{\mathrm{EB}}$ and $\log \tau^{\mathrm{KF}}$. In the 50 runs, the EB

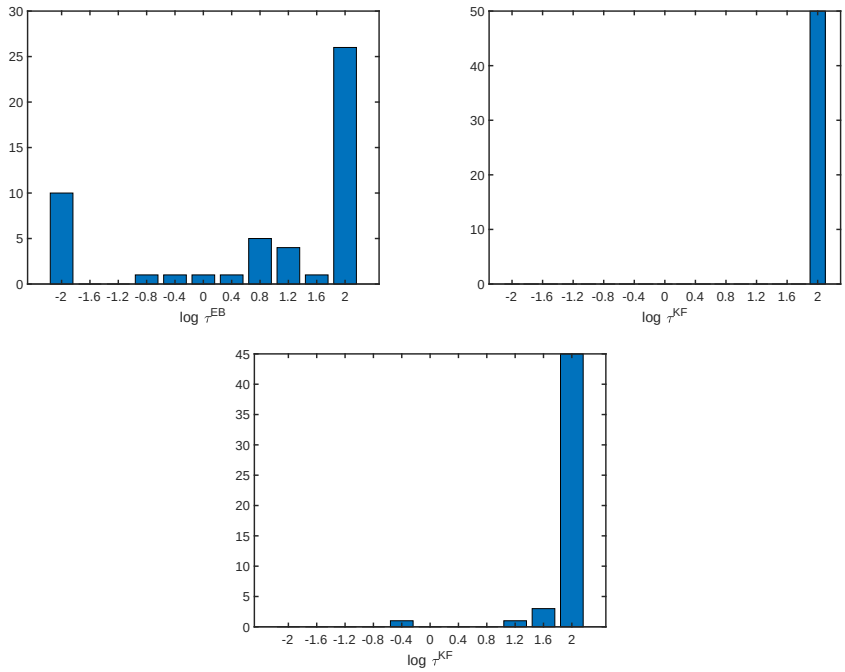


Figure 6.7: Histogram of the $\log \tau^{\mathrm{EB}}$ or $\log \tau^{\mathrm{KF}}$. Upper left: EB loss; upper right: KF loss (case 1); bottom: KF loss (case 2).

estimator takes many different values with no apparent pattern. For both case 1 and case 2, the KF estimator of τ' takes the value 2 very often, which is the maximal value of the constrained decision variable. None of the estimators recover the true $\tau' = 0$.

The behavior of the KF estimator can be explained by the observation that when τ increases, the function drawn from the Gaussian prior becomes smoother, and hence the subsampling step in the KF loss does not sacrifice too much information. Therefore, the KF loss exhibits a tendency to get smaller as τ increases. We can understand why EB cannot recover τ by studying the equivalence of Gaussian measures. As shown in [71], when dimension $d \leq 3$, the Gaussian measures $\mathcal{N}(0, (-\Delta + \tau^2 I)^{-s})$ for different τ are equivalent; thus one cannot expect to recover τ using the information from one sample.

We can also consider the problem of recovering s, τ simultaneously, i.e., we solve

a joint minimization problem to get $s^{\mathrm{EB}}, \log \tau^{\mathrm{EB}}$ and $s^{\mathrm{KF}}, \log \tau^{\mathrm{KF}}$. The set-up is the same as above, with the sample drawn from $\mathcal{N}(0, (-\Delta + \tau^2 I)^{-s})$ for $\tau = 1$ and $s = 2.5$. We form the EB and KF loss for 50 instances of different draws and find the minimizers as corresponding estimators. The histograms of the estimators are shown in Figure 6.8 and 6.9. These figures show that in this joint optimization, the EB method picks the correct value $s^{\mathrm{EB}} = 2.5$ for estimating s, and exhibit no patterns for $\log \tau^{\mathrm{EB}}$; the KF method finds values close to 1 for s^{KF}, as it would in the absence of simultaneous estimation of τ', and selects the largest possible value in the constraint for $\log \tau^{\mathrm{KF}}$, here being 2. The conclusion is that the fact that τ' cannot be learned accurately does not influence the estimation of the regularity parameter s in a context in which the two are learned simultaneously. Indeed, this conclusion also holds when we are recovering the three parameters (s, σ, τ) simultaneously.

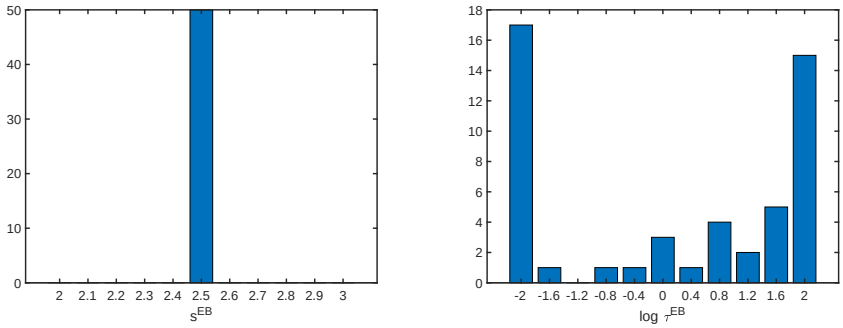


Figure 6.8: EB approach. Left: histogram of the s^{EB}; right: histogram of the $\log \tau^{\mathrm{EB}}$.

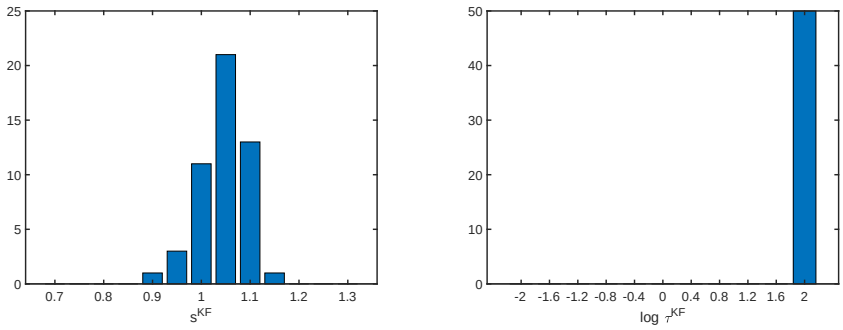


Figure 6.9: KF approach. Left: histogram of s^{KF}; right: histogram of the $\log \tau^{\mathrm{KF}}$.

6.3.2 Other Well-specified Examples

In this subsection, we consider numerical examples for recovering parameters of a random field in the well-specified case, going beyond the Matérn process studied

thus far.

6.3.2.1 Recovery of regularity parameter for variable coefficient elliptic operator

Set $D = [0,1]$ so that $d = 1$. The theoretical result in Section 6.2 assumes the function observed $u^\dagger$ is drawn from $\mathcal{N}(0,(-\Delta)^{-s})$ on a torus. In this subsection, we assume $u^\dagger$ is drawn from $\mathcal{N}(0,(-\nabla\cdot(a\nabla\cdot))^{-s})$ for some non-constant function a, and that the elliptic operator implicit in this defintion of a Gaussian measure is equipped with homogeneous Dirichlet boundary condition on D. We observe its values on the 2^9 equidistributed points of the total 2^{10} grid points used for discretization.

Here we select a coefficient $a(x)$ that exhibits a discontinuity at $x = 1/2$:

$$a(x) = \begin{cases} 1 & x \in [0,1/2] \\ 2 & x \in (1/2,1]\,. \end{cases} \tag{6.3.3}$$

As a consequence the induced operator is not the Laplacian. We pick $s = 2.5$ to draw a sample $u^\dagger$.

In the well-specified case, the GP used in defining the EB and KF estimators is parameterized by $\mathcal{N}(0,(-\nabla\cdot(a\nabla\cdot))^{-t})$ and we aim to learn parameter t given a data calculated using a draw from the same measure with $t = s$. We consider the well-specified case here (the misspecified case will be considered in Subsection 6.4.1.) We output the histogram of the EB and KF estimators for 50 different draws of $u^\dagger$ in Figure 6.10. The experiments show that for the variable coefficient elliptic operator

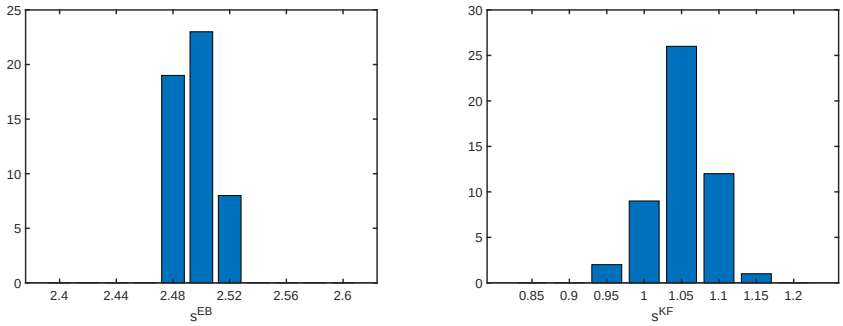


Figure 6.10: Histogram of the regularity estimators for the variable coefficient covariance case. Left: EB; right: KF.

model, EB and KF succeed in converging to the correct limits. We can calculate the

(normalized) variance of the two estimators based on the histograms:

$$\frac{\mathrm{Var}(s^{\mathrm{EB}})}{s^2} \approx 7.8 \times 10^{-5} \quad \text{and} \quad \frac{\mathrm{Var}(s^{\mathrm{KF}})}{((s-d/2)/2)^2} \approx 4 \times 10^{-3}\,.$$

The relative magnitude is similar to the one in Subsection 6.2.8.

6.3.2.2 Recovery of discontinuity position for conductivity field

Define the conductivity field $a_\theta : [0,1] \mapsto \mathbb{R}$, and parameterized by $\theta \in [0,1]$, via

$$a_\theta(x) = \begin{cases} 1 & x \in [0,\theta] \\ 2 & x \in (\theta,1]\,. \end{cases} \tag{6.3.4}$$

In this subsection, we assume that our data $u^\dagger$ is obtained by solving the SPDE

$$-\nabla \cdot (a_{1/2} \nabla u^\dagger) = \xi\,,$$

subject to a homogeneous Dirichlet boundary condition on $[0,1]$. We choose ξ as a random draw from $\mathcal{N}(0,(-\Delta)^{-1})$. We can view $u^\dagger$ is a sample drawn from $\mathcal{N}(0,C_a)$ where

$$C_a = (-\nabla \cdot (a_{1/2}\nabla\cdot))^{-1}(-\Delta)^{-1}(-\nabla \cdot (a_{1/2}\nabla\cdot))^{-1}. \tag{6.3.5}$$

We observe the value of $u^\dagger$ on the 2^9 equidistributed points of the total 2^{10} grid points used for discretization. We use EB and KF to estimate θ from the partial observation of the function $u^\dagger$ based on the GP model $\mathcal{N}(0,C_{a,s})$ where

$$C_{a,s} = (-\nabla \cdot (a_\theta \nabla\cdot))^{-1}(-\Delta)^{-s}(-\nabla \cdot (a_\theta \nabla\cdot))^{-1}. \tag{6.3.6}$$

The model is well-specified for $s = 1$ and misspecified for $s \neq 1$. Here consider the well-specified case in this subsection, i.e., $s = 1$, and $C_{a,s} = C_a$; the misspecified case is covered in Subsection 6.4.2.

We let the domain for θ be $[0.3, 0.7]$ in the definition of EB and KF estimators. We compute the estimators for 50 different draws of $u^\dagger$. The histograms of the EB and KF estimators are shown in Figure 6.11. The loss functions for one random instance are shown in Figure 6.12.

Our experiments show that both EB and KF can recover $\theta = 1/2$, and the recovery is very stable with respect to different draws of $u^\dagger$ from the SPDE. We conclude that the EB and KF can go beyond the Matérn-like kernel model in practice; recovering the point of discontinuity of the conductivity field is an example of this fact.

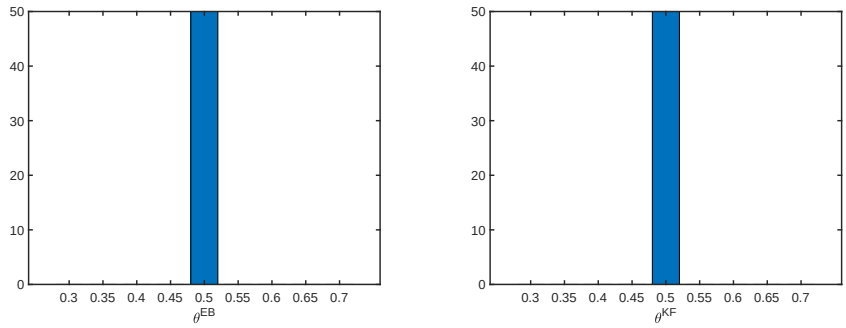


Figure 6.11: Histogram of the discontinuity position estimators (well-specified). Left: EB; right: KF.

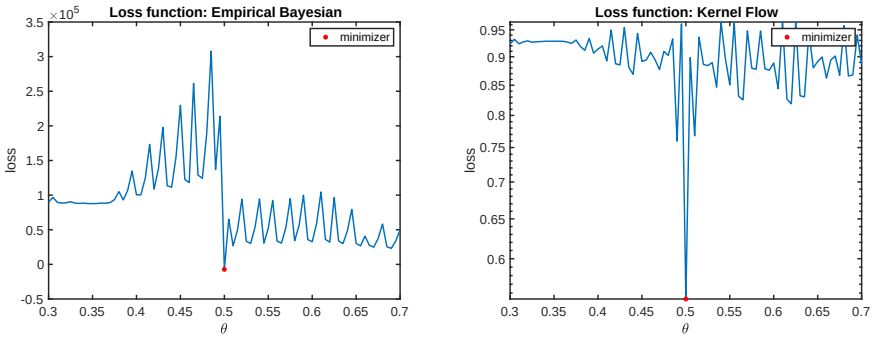


Figure 6.12: Loss function for recovering the discontinuity (well-specified). Left: EB; right: KF.

6.3.3 Computational Aspects

In this subsection, we add some discussions about the computational aspects. We start by remarking on how to compute the kernel function and sample the GP realization generally. Every kernel operator we consider involves certain differential operators. We discretize these differential operators and perform an eigenfunction decomposition of the obtained matrix. Then we use these eigenfunctions and eigenvalues to compute approximation of the kernel matrix, and draw samples from the GP with the covariance matrix being the kernel matrix; see also discussions above Remark 6.2.2. This is similar to the spectral expansion of a kernel function and the Mercer decomposition of a GP.

Practical applications of hierarchical GPR require weighting statistical efficiency against computational complexity. Although the regularity models covered in this chapter appear to produce well-behaved EB and KF loss functions with easily identifiable global minimizers, models with high dimensional parameter space typically

require using algorithms such as gradient descent which do not come with theoretical guarantees on the identification of global minimizers. Furthermore, when the size of the data is large, computation becomes a limiting factor, and subsampling offers a traditional remedy when combined with gradient descent, but again theoretical guarantees are not typically to be expected. The stochastic algorithm presented in [206] for KF can be interpreted as an SGD algorithm aimed at minimizing the average loss

$$\mathbb{E}_{\pi_1}\mathbb{E}_{\pi_2}\mathsf{L}^{\mathrm{KF}}(\theta, \pi_1\mathcal{X}, \pi_2\pi_1\mathcal{X}, u^\dagger),$$

via draws from the distribution of π_1 and π_2 ($\pi_1\mathcal{X}$ is a random subsampling of $\mathcal{X}$, and $\pi_2\pi_1\mathcal{X}$ is a further random subsampling of $\pi_1\mathcal{X}$). The efficacy of an analogous strategy for EB remains unclear due to the presence of the log determinant term in the loss. It is of future interest to explore further the computational aspects of the EB and KF approaches to hierarchical learning.

6.4 Model Misspecification

All our preceding experiments are focused on the well-specified case: the function $u^\dagger$ is drawn from the GP model assumed in the estimation, or equivalently, the model for $u^\dagger$ and for the kernel family K_θ in defining the loss functions are matched. This subsection studies model misspecification. We consider two possible ways to misspecify the model: (1) the function $u^\dagger$ is drawn from a GP which is different from that used in defining the loss function; (2) the function $u^\dagger$ is a fixed deterministic function. The second case may arise, for example, if the function comes from a solution of a PDE with some physical data, and there is no natural stochastic context for its provenance. The aim of this subsection is to study the behavior of the EB and KF estimators to compare their robustness to model misspecification.

6.4.1 Stochastic model misspecification for recovering regularity

In this subsection, we assume $u^\dagger$ is drawn from $\mathcal{N}(0, (-\nabla \cdot (a\nabla\cdot))^{-s})$, while the GP used in defining the EB and KF estimators is still $\mathcal{N}(0, (-\Delta)^{-t})$. This results in a model misspecification corresponding to the well-specified model in Subsection 6.3.2.1. As in Subsection 6.3.2.1, we select a as in (6.3.3) and we set $s = 2.5$ to draw the sample $u^\dagger$. Figure 6.13 shows the histograms of the minimizers of the EB and KF loss functions obtained from 50 independent draws from the Gaussian Process. Despite misspecification, the EB and KF estimators are still concentrated

around 2.5 and 1, respectively. We also calculate the variance:

$$\frac{\mathrm{Var}(s^{\mathrm{EB}})}{s^2} \approx 5.9 \times 10^{-4} \quad \text{and} \quad \frac{\mathrm{Var}(s^{\mathrm{KF}})}{((s-d/2)/2)^2} \approx 6.8 \times 10^{-4}\,.$$

In this example, the (normalized) variance of KF of EB are of similar magnitude. This is different from the well-specified case in Subsection 6.3.2.1 where the variance of EB is much smaller than KF.

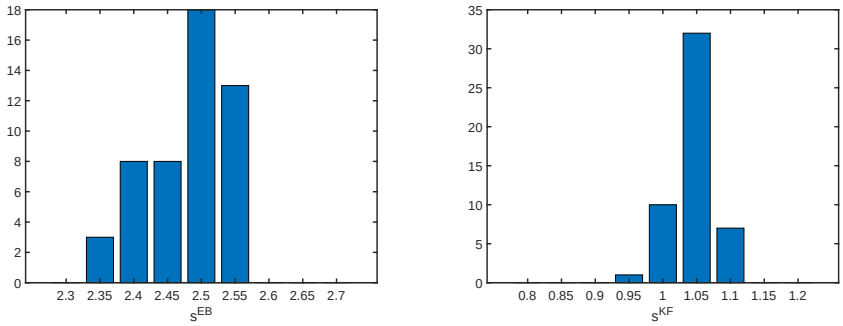


Figure 6.13: Histogram of the regularity estimators under model misspecification. Left: EB; right: KF.

6.4.2 Stochastic model misspecification for recovering discontinuity

In this subsection, we consider the model misspecifications that correspond to the well-specified case in Subsection 6.3.2.2. For the GP defining the EB and KF estimators we use the centred Gaussian with covariance operator given by (6.3.6) with $s = 5$; meanwhile $u^\dagger$ is drawn from the centred Gaussian with covariance operator given by (6.3.5); thus we are in a misspecified version of the setting arising in Subsection 6.3.2.2 and, as there, our aim is to recover the point of discontinuity. We illustrate the loss functions for a single draw of $u^\dagger$ in Figure 6.14. These plots are not sensitive to the particular draw of $u^\dagger$ and illustrate the robustness of KF (and the lack of robustness of EB) to this misspecification. Indeed, the EB estimator gives 0.3 which is the lower boundary of the compact parameter space used in the minmization, while the KF estimator picks the true parameter 0.5. The loss function of KF, shown in Figure 6.14, exhibits a sharp global minimizer at $\theta = 0.5$.

6.4.3 Deterministic model

In this subsection, we consider the EB and KF estimators for the parameter t in the GP model $\mathcal{N}(0, (-\Delta)^{-t})$ where Δ is equipped with homogeneous Dirichlet boundary conditions on $[0, 1]$. However, rather than choosing $u^\dagger$ that is drawn from the GP

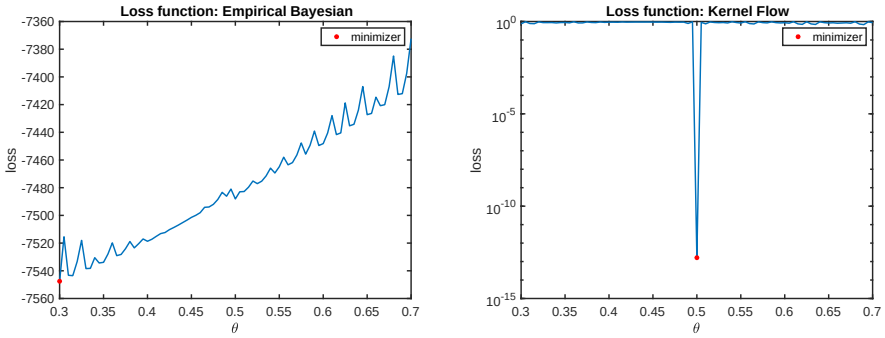


Figure 6.14: Loss function for estimating the discontinuity parameter under model misspecification. Left: EB; right: KF.

$\mathcal{N}(0,(-\Delta)^{-s})$ for some s (as we did in Section 6.2), we choose it be the solution to the equation $(-\Delta)^s u^\dagger(\cdot) = \delta(\cdot - 1/2)$, i.e., $u^\dagger$ is the Green function corresponding to the differential operator $(-\Delta)^s$ and evaluated at $y = 1/2$. Since $u^\dagger$ has no stochastic background, we understand this situation as a deterministic model misspecification.

We observe the value of $u^\dagger$ on the 2^9 equidistributed points of the total 2^{10} grid points used for discretization. We conduct numerical experiments to find the value of the EB and KF estimators. Our experiments show that the EB estimator returns $2s$ and the KF estimator returns s for this one dimensional example. The loss function in the case $s = 1.2$ is shown in Figure 6.15.

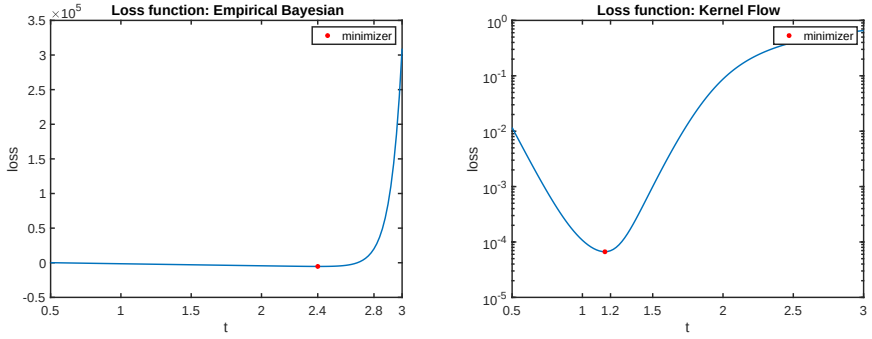


Figure 6.15: Loss function for estimating the regularity parameter under deterministic $u^\dagger$. Left: EB; right: KF.

We now describe some regularity considerations in order to understand the observed phenomenon. In this one dimensional example, $\delta(\cdot - 1/2)$ belongs to $H^\eta([0,1])$ for any $\eta < -1/2$, so the solution $u \in H^{2s+\eta}([0,1])$ for any $\eta < -1/2$. It is of critical regularity $2s - 1/2$, but this criticality is not homogeneous: it is caused by the presence of a singularity induced by the Dirac function.

The discussion in Section 6.2 implies KF will recover $s - 1/4$ while EB recovers $2s$ for a function with homogeneous critical regularity $2s - 1/2$. However, the experiments here show that KF recovers s while EB recovers $2s$, for this function with critical regularity $2s - 1/2$; unlike the setting in Section 6.2, here the ground truth lacks spatial homogeneity. This suggests that the KF estimator for the regularity parameter is sensitive to whether the regularity of the target function is spatially homogeneous or not. This fact is not surprising, considering the vast literature on adaptive approximation for functions with singularities, which implies the presence of a singularity will exert considerable influence on the approximation error resulting from minimizing the KF loss function. In this example, the optimal approximation in KF error comes at $t = s$. We can understand this phenomenon as follows. Recall $u^\dagger = (-\Delta)^{-s}\delta(\cdot - 1/2)$. Using $\mathcal{N}(0, (-\Delta)^{-t})$ in the GPR is equivalent to using the basis functions $\text{span}_{j\in J_q}\{(-\Delta)^{-t}\delta(\cdot - x_j)\}$ (as in Section 6.2.1) with x_i being the data points indexed by $j \in J_q$, to approximate $u^\dagger$. When $t = s$ and one of the $x_j = 1/2$, the ground truth will just be in the basis functions set, so it is straightforward to imagine $t = s$ leads to the smallest approximation error, and KF picks this value.

We understand the fact that EB still picks $t = 2s$ by making the following observation: there are only two terms in the EB loss function. The log determinant term remains the same for each t when $u^\dagger$ changes. For the norm term $\|u(\cdot, t, q)\|_t^2$, the blow-up rate depends on the regularity of $u^\dagger$. Here, it makes no difference whether the regularity of $u^\dagger$ is spatially homogeneous or not.

6.4.4 Discussions

The above numerical experiments reveal complicated behavior of EB and KF with respect to model misspecification. In the second experiment, we found that KF is robust while EB is not, for a certain type of GP model misspecification. This appears natural since EB is based on probabilistic modeling whilst KF is purely based on approximation theoretic criteria. In Subsection 6.4.2 the prior used in EB is mutually singular with respect to the GP that $u^\dagger$ is drawn from and it is not suprising that EB is fragile. On the other hand, KF does not require probabilistic modeling to motivate it, and so its robustness to misspecifications behaves differently. Indeed, in the second experiment, the discontinuity point influences the approximation accuracy significantly, and even the kernel used in defining KF is misspecified, KF still succeeds in selecting the correct parameter, as it focuses on the approximation accuracy rather than statistical inference.

In the well-specified cases, e.g. experiments in Section 6.2, EB outperforms KF in terms of the variance of estimators. Therefore, if $u^\dagger$ is a random object and we know the prior correctly, then EB should be a preferable choice for estimating parameters. If this is not the case and misspecification occurs, EB might be vulnerable and KF could be a potential alternative.

6.5 Conclusions

We have studied the Empirical Bayes and Kernel Flow approaches to hyperparameter learning. The first approach is based on statistical considerations, while the second approach originates from an approximation theoretic viewpoint. Their distinct objectives lead them to different behaviors and different interpretations of optimality.

For the Matérn-like process model, we made a detailed theoretical study of the recovery of the regularity parameter. We proved the EB estimator converges to s, while the KF estimator converges to $\frac{s-d/2}{2}$, both results holding in probability in the large data limit if the regularity of the GP that $u^\dagger$ draws from is s. Our experiments illustrate that, in terms of the L^2 error $\|u(\cdot,t,q) - u^\dagger\|_0^2$, the parameter $t = \frac{s-d/2}{2}$ relates to the minimal t that achieves the fast error rate while $t = s$ relates to the t that achieves the smallest error, averaged over the GP $u^\dagger \sim \mathcal{N}(0,(-\Delta)^{-s})$. This demonstrates the different drivers that guide the EB and KF methods in selecting the parameters. The statistical and approximation theoretic principles behind them lead to the differences between them.

In the theoretical study, we developed a Fourier analysis toolkit for this problem, and as a byproduct, we showed the consistency of recovering σ in the Matérn-like process for the EB method. Recovery of the lengthscale parameter and recovery of several parameters simultaneously was studied via numerical experiments. It is of future interest to perform theoretical studies explaining these empirically observed phenomena. Furthermore, the theory in this chapter is based on an equidistributed design for the data location, and the generalization to randomized design remains a potential further direction. Also, our focus in this work is on the noiseless observation setting, and an extension to the noisy case is of future theoretical interest.

Our numerical experiments for additional well-specified and misspecified models extend the scope of this work beyond the Matérn-like kernels. Both the two estimators work very well in the well-specified models we consider; we would like to explore this more in the future, both theoretically and numerically, potentially

in more complex models that are present in machine learning. The variance and robustness of the estimators behave differently for the misspecified models. The variabilities in robustness are in line with our expectation since these estimators follow from different decision rules; these rules can vary considerably in sensitivity to model mismatches of different kinds. In practice, users should choose the correct approach to avoid high sensitivity to likely model errors present.

As a summary, this work demonstrates some basic aspects of the difference between Bayesian and approximation theoretic approaches for hierarchical learning. Generally, it is of interest to study EB and KF for other types of models and to study other parameter selection criteria based on the two principles beyond EB and KF, such as a fully Bayesian approach or another choice of d for the approximation, and identify their pros and cons under different scenarios. We are interested in exploring the theoretical and practical performance of methods under such a framework, and we believe that a diversity in such methods will enable users to deal with the model misspecification that is to be expected in many applications.

Chapter 7

ADDTIONAL TOPICS IN RANDOMIZED NUMERICS AND POSTERIOR SAMPLING

This chapter covers additional topics related to using probability and statistics for scientific computing, extending beyond previous chapters.

In the first part, which is based on our work [42], we address the computational challenges of Gaussian processes and kernel methods, particularly in high-dimensional problems such as chemistry, where the screening effects explored by the sparse Cholesky factorization in Chapter V may not be substantial. Our goal is to construct an accurate low rank approximation of the dense kernel matrix, and we achieve this by using randomized numerical linear algebra. Our approach strikes a favorable balance between exploration and exploitation, allowing us to efficiently discover low rank structures through *randomness*.

In the second part, which is based on our work [49], we investigate the use of gradient flows for posterior sampling in Bayes inference. Specifically, we focus on identifying *canonical* choices of gradient flows that can lead to favorable properties in sampling. This research contributes to the basic understanding of gradient flows and their applications in Bayesian inference, thus helping harness the uncertainty quantification power of the Gaussian process based approach presented in Sections IV, V, and VI. .

7.1 Randomly Pivoted Cholesky for Scalable Kernel Methods

Suppose we have a collection of points $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\} \subset \Omega \subset \mathbb{R}^d$. The interaction between these points is encoded in a kernel function $k : \Omega \times \Omega \to \mathbb{R}$, using which we get a kernel matrix $\boldsymbol{K} := [k(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{1 \le i,j \le N}$. This matrix plays a central role in Gaussian processes and kernel methods for prediction and inference, as we have seen in Section IV, V, and VI.

The algorithm we developed in Section V computes a sparse Cholesky factorization of $\boldsymbol{K}^{-1}$ based on screening effects in spatial statistics. The screening effect is most effective in low-dimensional physical space since it requires many neighboring points to present. Here we deal with problems where d can be large. In such a high dimensional setting, it is more reasonable to aim for a low-rank approximation of $\boldsymbol{K}$

rather than a full-scale approximation as in the sparse Cholesky factorization.

The column Nyström approximation is a popular technique for constructing a low-rank psd approximation of a kernel matrix $\boldsymbol{K}$. More precisely, it approximates $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ via

$$\hat{\boldsymbol{K}}_{\mathsf{S}} = \boldsymbol{K}(:,\mathsf{S})\boldsymbol{K}(\mathsf{S},\mathsf{S})^{\dagger}\boldsymbol{K}(\mathsf{S},:) \in \mathbb{R}^{N\times N},$$

where $\mathsf{S} = \{s_1, ..., s_k\} \subset \{1, ..., N\}$ is a carefully chosen set of columns. In this expression, $\boldsymbol{K}(:,\mathsf{S})$ is the submatrix with the selected columns, $\boldsymbol{K}(\mathsf{S},:)$ is the submatrix with the selected rows, and $\boldsymbol{K}(\mathsf{S},\mathsf{S})^{\dagger}$ is the Moore–Penrose pseudoinverse of the submatrix with the selected rows and columns.

Many methods exist in the literature for selecting S. A classical approach in experimental design and numerical linear algebra is to select each s_i sequentially in a greedy manner [84]. Having selected an index set S_m with cardinality m, the next index s_{m+1} is selected according to $s_{m+1} = \operatorname{argmax}_i \boldsymbol{R}^{(m)}_{ii}$ where $\boldsymbol{R}^{(m)} = \boldsymbol{K} - \hat{\boldsymbol{K}}_{\mathsf{S}_m}$ is the Schur complement at the m-step. Thus the greedy approach *exploits* large diagonal entries sequentially to form S.

Remark 7.1.1. *The greedy approach is known as the sequential maximum uncertainty design under a Gaussian process model [94], as the diagonals of the Schur complement represent the posterior variances of the Gaussian process. Nyström approximation is also mathematically equivalent to Cholesky factorization in numerical linear algebra, and in this context, the greedy approach is known as complete pivoting [124].*

Another popular approach is to sample points uniformly at random [291] to form S. This method is simple, and the sampled points can *explore* the space freely.

Although the greedy and uniform sampling approaches are widely used, they may not always be effective at identifying the most important columns in a matrix. The greedy method selects the column with the largest diagonal entry in the residual matrix at each step, but it may overlook columns with smaller diagonal entries that could significantly contribute to approximating the target matrix. Conversely, the uniform sampling selects columns randomly, which can include irrelevant or redundant columns, while missing significant columns that may have large diagonal entries.

Remark 7.1.2. *More specifically, we can construct the following failure mode for the greedy approach*

$$A = \begin{pmatrix} \frac{1}{2}\mathbf{1}_{N-N^{1/2}}\mathbf{1}_{N-N^{1/2}}^T & \\ & \boldsymbol{I}_{N^{1/2}} \end{pmatrix}$$

where $\mathbf{1}_m$ *is an all one vector in* $\mathbb{R}^m$*, and* $\boldsymbol{I}_m$ *is the identity matrix in* $\mathbb{R}^{m\times m}$*. In this example, the greedy approach will keep selecting columns corresponding to the lower-right block until* $N^{1/2}$ *columns; with each additional column, the trace norm error will decrease by* 1*. However, selecting only one column corresponding to the upper-left block can decrease the approximation error (measured in the trace norm) by* $(N - N^{1/2})/2$*. The greedy approach exploits too much and does not explore the upper-left block.*

Given the potential failures of the greedy and uniform sampling, we consider instead a randomized approach, Randomly Pivoted Cholesky (RPCholesky), that balances exploration and exploitation. Having selected an index set S_m, the next index s_{m+1} is sampled according to the probability

$$\mathbb{P}\{s_{m+1} = i\} = \boldsymbol{R}_{ii}^{(m)}/\mathrm{tr}\boldsymbol{R}^{(m)},$$

where $\boldsymbol{R}^{(m)} = \boldsymbol{K} - \hat{\boldsymbol{K}}_{\mathsf{S}_m}$ is the Schur complement at the m-step. The sampling probability scales with the values of diagonal entries in the residue matrix, so RPCholesky exploits the large diagonal entries. On the other hand, the randomness allows RPCholesky to explore small diagonal entries.

In [42], we provide theoretical results showing that RPCholesky is provably accurate. See the following theorem. Here $\boldsymbol{K}_r$ is the best possible rank-r approximation of $\boldsymbol{K}$.

Theorem 7.1.3. *Fix* $r \in \mathbb{N}$ *and* $\epsilon > 0$*, and let* $\boldsymbol{K}$ *be a psd matrix. The column Nyström approximation* $\hat{\boldsymbol{K}}_{\mathsf{S}_k}$ *produced by RPCholesky attains the bound*

$$\mathbb{E}\,\mathrm{tr}(\boldsymbol{K} - \hat{\boldsymbol{K}}_{\mathsf{S}_k}) \le (1+\epsilon)\cdot\mathrm{tr}(\boldsymbol{K} - \boldsymbol{K}_r),$$

provided that the number k *of columns satisfies*

$$k \ge \frac{r}{\epsilon} + \min\left\{ r\log_+\left(\frac{1}{\epsilon\eta}\right), r + r\log_+\left(\frac{2^r}{\epsilon}\right)\right\}.$$

The relative error η *is defined by* $\eta := \mathrm{tr}(\boldsymbol{K} - \boldsymbol{K}_r)/\mathrm{tr}(\boldsymbol{K})$*. As usual,* $\log_+(x) := \max\{\log x, 0\}$ *for* $x > 0$ *and the logarithm has base* e.

For more details of the analysis and numerical experiments, we refer to [42].

7.2 Gradient Flow for Sampling: Energy, Metric, and Numerics

In this section, we describe more concretely the topic introduced in Section 1.2.5.2 and provide a mathematical description of the theoretical results we have proven. More comprehensive information can be found in our paper [49].

We remind the reader of the definition of gradient flows on probability space. To describe any gradient flow, we must first specify an energy functional and a metric. Let $\mathcal{P}$ be the probability space in $\mathbb{R}^d$, $\mathcal{E}$ be an energy functional on $\mathcal{P}$ that maps to $\mathbb{R}$, and g_ρ be a Riemannian metric on $\mathcal{P}$ at ρ, where $T_\rho\mathcal{P}$ is the tangent space consisting of measures with zero means. The metric can also be expressed as $g_\rho(\sigma_1, \sigma_2) = \langle \sigma_1, M(\rho)\sigma_2 \rangle_{L^2}$, where $M(\rho)$ is an operator that acts on $T_\rho\mathcal{P}$. The gradient flow equation takes the form:

$$\frac{\partial \rho_t}{\partial t} = -\nabla_g \mathcal{E}(\rho_t) = -M(\rho_t)^{-1} \frac{\delta \mathcal{E}}{\delta \rho}|_{\rho=\rho_t},$$

where ∇_g is the Riemannian gradient operator and $\frac{\delta \mathcal{E}}{\delta \rho}$ is the first variation of $\mathcal{E}$.

In the following, we discuss the choices of energy functionals and metrics that can lead to favorable properties for sampling ρ_{post}, which we know up to a normalization constant. For simplicity, we focus on densities which are supported in a compact set Ω, positive, and smooth. For these densities, the differential structure can be made mathematically precise. But many of our results apply to more general densities; for details see [49].

Energy Functionals A popular choice of $\mathcal{E}$ is the Kullback–Leibler (KL) divergence

$$\mathcal{E}(\rho) = \mathrm{KL}[\rho \| \rho_{\text{post}}] = \int \rho \log\Big(\frac{\rho}{\rho_{\text{post}}}\Big) \mathrm{d}\theta \,.$$

Its first variation is

$$\frac{\delta \mathcal{E}}{\delta \rho} = \log \rho - \log \rho_{\text{post}} - \int (\log \rho - \log \rho_{\text{post}}) \mathrm{d}\theta \,,$$

where we impose $\int \frac{\delta \mathcal{E}}{\delta \rho} \mathrm{d}\theta = 0$. A remarkable property of the KL divergence is that, $\frac{\delta \mathcal{E}}{\delta \rho}$ remains unchanged if we scale ρ_{post} by any positive constant $c > 0$, i.e. if we change ρ_{post} to $c\rho_{\text{post}}$. This property eliminates the need to know the normalization constant of ρ_{post} in order to calculate the first variation. It is common in Bayesian inference for the normalization to be unknown and indeed the fact that MCMC methods do not need the normalization constant is central to their widespread use; it is desirable that the methodology presented here has the same property.

A natural question is whether there are any other energy functionals satisfying the same invariant property with respect to the normalization constant. The answer is *no*. We show that this property of the KL divergence is unique: among all f-divergences with continuously differentiable f defined on the positive reals it is the only one to have this property.

Here the f-divergence between two continuous density functions ρ and ρ_{post}, positive everywhere, is defined as

$$D_f[\rho\|\rho_{\text{post}}] = \int \rho_{\text{post}} f(\frac{\rho}{\rho_{\text{post}}})\mathrm{d}\theta.$$

For convex f with $f(1) = 0$, Jensen's inequality implies that $D_f[\rho\|\rho_{\text{post}}] \geq 0$. In what follow we view this f-divergence as a function of probability measure ρ, parameterized by ρ_{post}; in particular we observe that this parameter-dependent function of probability density ρ makes sense if ρ_{post} is simply a positive function: it does not need to be a probability density; we may thus scale ρ_{post} by any positive real.

Proposition 7.2.1. *Assume that $f : (0,\infty) \to \mathbb{R}$ is continuously differentiable and $f(1) = 0$. Then the KL divergence is the only f-divergence (up to scalar factors) whose first variation is invariant with respect to $\rho_{\text{post}} \mapsto c\rho_{\text{post}}$, for any $c \in (0,\infty)$ and for any $\rho_{\text{post}} \in \mathcal{P}$.*

Proof. First, note that we already know that the KL divergence satisfies the desired property by direct calculation. To show the uniqueness, we consider ρ and ρ_{post} continuous density functions, positive everywhere. We start by calculating the first variation of the f-divergence:

$$\frac{\delta D_f}{\delta \rho} = f'\Big(\frac{\rho}{\rho_{\text{post}}}\Big) - \int f'\Big(\frac{\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta.$$

We assume that this first variation is invariant under scaling of ρ_{post} by a positive real c in order to determine constraints that this imposes on f. We thus have that

$$f'\Big(\frac{\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) - \int f'\Big(\frac{\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta = f'\Big(\frac{c\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) - \int f'\Big(\frac{c\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta,$$

and hence that

$$f'\Big(\frac{\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) - f'\Big(\frac{c\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) = \int f'\Big(\frac{\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta - \int f'\Big(\frac{c\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta, \tag{7.2.1}$$

for any $\theta \in \mathbb{R}^d$ and $c > 0$. Because ρ and ρ_{post} integrate to 1 and they are continuous, there exists $\theta^\dagger$ such that $\rho(\theta^\dagger)/\rho_{\text{post}}(\theta^\dagger) = 1$. Setting $\theta = \theta^\dagger$ in the above identity, we obtain

$$g(c) := f'(1) - f'(c) = \int f'\Big(\frac{\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta - \int f'\Big(\frac{c\rho}{\rho_{\text{post}}}\Big)\mathrm{d}\theta. \tag{7.2.2}$$

Combining (7.2.1), (7.2.2) to eliminate the integrated terms we obtain

$$f'\Big(\frac{\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) - f'\Big(\frac{c\rho(\theta)}{\rho_{\text{post}}(\theta)}\Big) = g(c), \tag{7.2.3}$$

where c is an arbitrary positive number. Note, furthermore, that $g(c)$ is continuous since f is continuously differentiable. Since ρ and ρ_{post} are arbitrary, we can write (7.2.3) equivalently as

$$f'(y) - f'(cy) = g(c), \tag{7.2.4}$$

for any $y, c \in \mathbb{R}_+$. Let $h : \mathbb{R} \to \mathbb{R}$ such that $h(z) = f'(\exp(z))$. Then, we can equivalently formulate (7.2.4) as

$$h(z_1) - h(z_2) = r(z_1 - z_2), \tag{7.2.5}$$

for any $z_1, z_2 \in \mathbb{R}$ and $r : \mathbb{R} \to \mathbb{R}$ such that $r(t) = g(\exp(-t))$.

We can show r is a linear function. Setting $z_1 = z_2$ in (7.2.5) shows that $r(0) = 0$. Note also that, again by (7.2.5),

$$r(z_1 - z_2) + r(z_2 - z_3) = h(z_1) - h(z_3) = r(z_1 - z_3).$$

Hence, since z_1, z_2 and z_3 are arbitrary, we deduce that for any $x, y \in \mathbb{R}$, it holds that

$$r(x) + r(y) = r(x + y). \tag{7.2.6}$$

Furthermore r is continuous since f is assumed continuously differentiable. With the above conditions, it is a standard result in functional equations that $r(x)$ is linear. Indeed, as a sketch of proof, by (7.2.6) we can first deduce $r(n) = nr(1)$ for $n \in \mathbb{Z}$. Then, by setting x, y to be dyadic rationals, we can deduce $r(\frac{i}{2^k}) = \frac{i}{2^k} r(1)$ for any $i, k \in \mathbb{Z}$. Finally, using the continuity of the function r, we get $r(x) = xr(1)$ for any $x \in \mathbb{R}$. For more details, see [87].

Using the fact that r is linear and the equation (7.2.5), we know that h is an affine function and thus $f'(\exp(z)) = az + b$ for some $a, b \in \mathbb{R}$. Equivalently, $f'(y) =$

$a\log(y)+b$. Using the condition $f(1)=0$, we get $f(y)=ay\log(y)+(b-a)(y-1)$. Plugging this f into the formula for D_f, we get

$$D_f[\rho||\rho_{\text{post}}] = a\text{KL}[\rho||\rho_{\text{post}}],$$

noting that the affine term in $f(y)$ has zero contributions in the formula for D_f. The proof is complete. □

Our result justifies that the KL divergence is a unique energy functional ideal for sampling tasks in Bayesian inference.

Metrics As noted in Section 1.2.5.2, many different metrics have been proposed in the literature. Some examples include (here formally, we describe the metric using $M(\rho)$):

- Wasserstein metric [135, 200]: $M(\rho)^{-1}\psi = -\nabla\cdot(\rho\nabla\psi)$;
- Stein metric [168]:

  $$M(\rho)^{-1}\psi = -\nabla_\theta\cdot\left(\rho(\theta)\int\kappa(\theta,\theta',\rho)\rho(\theta')\nabla_{\theta'}\psi(\theta')\mathrm{d}\theta'\right)$$

  where κ is a kernel function;
- Wasserstein-Fisher-Rao metric [175]: $M(\rho)^{-1}\psi = -\nabla\cdot(\rho\nabla\psi)+\rho(\psi-\mathbb{E}_\rho[\psi])$;
- Kalman-Wasserstein metric [95]: $M(\rho)^{-1}\psi = -\nabla\cdot(\rho C(\rho)\nabla\psi)$ where $C(\rho)$ is the covariance matrix of ρ.

All the above metrics involve an elliptic operator $-\nabla\cdot(\rho\nabla\cdot)$ or its variants. This operator has its origin in optimal transport [135, 200, 279]. In general, the convergence rate of the gradient flows under such transport-type metrics depend on the property of ρ_{post}, particularly its log-Sobolev constant. One exception is the Wasserstein-Fisher-Rao gradient flow, where the part $\rho(\psi-\mathbb{E}_\rho[\psi])$ introduces some nonlocality in the flow equation, which accelerates the dynamics.

In fact, we can consider the Fisher-Rao metric directly

$$M(\rho)^{-1}\psi = \rho(\psi-\mathbb{E}_\rho[\psi]),$$

which is a fundamental subject in statistics.

Remark 7.2.2. *The Fisher-Rao metric was introduced by C.R. Rao [227] via the Fisher information matrix. The original definition is in parametric density spaces, and the corresponding Fisher-Rao gradient flow in the parameter space leads to natural gradient descent [6]. The Fisher-Rao metric in infinite dimensional probability spaces was discussed in [88, 255]. The concept underpins information geometry [7, 12].*

With this metric, we can write down the gradient flow equation (with the energy functional to be the KL divergence):

$$\frac{\partial \rho_t}{\partial t} = \rho_t\big(\log \rho_{\text{post}} - \log \rho_t\big) - \rho_t \mathbb{E}_{\rho_t}[\log \rho_{\text{post}} - \log \rho_t]\,.$$

A remarkable property of the above flow is it is *invariant* to any diffeomorphism of the parameter space. In fact, consider any diffeomorphism $\varphi : \mathbb{R}^d \to \mathbb{R}^d$, we define $\tilde{\rho}_t = \varphi\#\rho_t$ and $\tilde{\rho}_{\text{post}} = \varphi\#\rho_{\text{post}}$ where # is the push-forward operator such that

$$\tilde{\rho}_t(\theta) = \rho_t(\varphi^{-1}(\theta))|\det \nabla\varphi^{-1}|$$
$$\tilde{\rho}_{\text{post}}(\theta) = \rho_{\text{post}}(\varphi^{-1}(\theta))|\det \nabla\varphi^{-1}|\,.$$

Then, using the above formula, one can derive that

$$\frac{\partial \tilde{\rho}_t}{\partial t} = \tilde{\rho}_t\big(\log \tilde{\rho}_{\text{post}} - \log \tilde{\rho}_t\big) - \tilde{\rho}_t \mathbb{E}_{\tilde{\rho}_t}[\log \tilde{\rho}_{\text{post}} - \log \tilde{\rho}_t]\,.$$

Thus, the flow equation remains invariant upon any transformation. Since for any reasonable ρ_{post}, it is possible to find a φ such that $\varphi\#\rho_{\text{post}}$ is a Gaussian distribution, the convergence property of the Fisher-Rao gradient flows will be the same for Gaussian posteriors and general posteriors. Naturally, we may wonder if such invariance property could be beneficial for the convergence of the flow.

In [49], along with some contemporary work [176], we show the uniform exponential convergence of the Fisher-Rao gradient flow:

Proposition 7.2.3. *Let ρ_t solve the Fisher-Rao gradient flow. Assume also that there exist constants $K, B > 0$ such that the initial density ρ_0 satisfies*

$$e^{-K(1+|\theta|^2)} \le \frac{\rho_0(\theta)}{\rho_{\text{post}}(\theta)} \le e^{K(1+|\theta|^2)}, \tag{7.2.7}$$

and both $\rho_0, \rho_{\text{post}}$ have bounded second moment

$$\int |\theta|^2 \rho_0(\theta)\mathrm{d}\theta \le B, \quad \int |\theta|^2 \rho_{\text{post}}(\theta)\mathrm{d}\theta \le B. \tag{7.2.8}$$

Then, for any $t \ge \log\big((1+B)K\big)$,

$$\mathrm{KL}[\rho_t \| \rho_{\text{post}}] \le (2 + B + eB)Ke^{-t}. \tag{7.2.9}$$

The convergence rate is uniform irrespective of the posterior distribution ρ_{post}, which is remarkable.

As mentioned above, this strong result could be related to the invariance property of the Fisher-Rao gradient flows. In [49], we describe the mathematical equivalence between the invariance of the flow and the invariance of the metric. In the literature, it has been shown that the Fisher-Rao metric is the only metric, up to scaling, that is invariant under any diffeomorphism of the state space [40, 11, 24]. This justifies the speciality of the Fisher-Rao metric.

Remark 7.2.4. *We note that if we instead assume a weaker condition of "invariance under any invertible affine transformations", namely affine invariance, more metrics can be found to satisfy the property. Examples include the Kalman-Wasserstein metric [95], which is affine invariant.*

Indeed, the idea of affine invariance has been introduced for MCMC methods in [102, 85], motivated by the empirical success of the Nelder-Mead simplex algorithm [195] in optimization. The numerical studies presented in [102] demonstrate that affine-invariant MCMC methods offer significant performance improvements over standard MCMC methods. This idea has been further developed to enhance sampling algorithms in more general contexts. Preconditioning strategies for Langevin dynamics to achieve affine-invariance were discussed in [159]. And in [95], the Kalman-Wasserstein metric was introduced, gradient flows in this metric were advocated and in [96] the methodology was shown to achieve affine invariance. Moreover, the authors in [95, 96, 216] used the empirical covariance of an interacting particle approximation of the mean-field limit, leading to a family of derivative-free sampling approaches in continuous time. Similarly, the work [170] employed the empirical covariance to precondition second-order Langevin dynamics. Affine invariant samplers can also be combined with the pCN (preconditioned Crank–Nicolson) MCMC method [58], to boost the performance of MCMC in function space [59, 72]. Another family of affine-invariant sampling algorithms is based on Newton or Gauss-Newton since using the Hessian matrix as the preconditioner in Newton's method induces the affine invariance property. Such methods include stochastic Newton MCMC [182], and the Newton flow with different metrics [68, 285].

Numerics To get implementable algorithms, we need to simulate the flow numerically. Despite the favorable theoretical property of the Fisher-Rao gradient flow, its numerical simulation requires extra effort. Particle methods based on birth-death

dynamics have been employed in previous studies [175, 176], but their effectiveness depends on the quality of the density estimator for particle distributions. These methods may deteriorate when applied to high-dimensional problems because of the need for many particles.

In [49], we investigate parametric approximations of the Fisher-Rao gradient flows. Specifically, we demonstrate the equivalence between the Gaussian projection (through moment closures) of the flow and natural gradient methods in variational inference. We provide convergence analysis in such cases. We refer to [49] for these discussions, including approximations of other gradient flows.

Additionally, we explore using Kalman methodology to obtain a derivative-free approximation of the Fisher-Rao gradient flow. This approach recovers a recently proposed Kalman-type sampler [132] that has demonstrated success in large-scale Bayes inverse problems.

Our ultimate goal is to enhance these approximations to create provable, efficient, and robust samplers for posterior distributions in Bayes inference. We will continue to work towards this goal.

BIBLIOGRAPHY

[1] Assyr Abdulle, E Weinan, Björn Engquist, and Eric Vanden-Eijnden. The heterogeneous multiscale method. *Acta Numerica*, 21:1–87, 2012.

[2] Robert A Adams and John JF Fournier. *Sobolev Spaces*. Elsevier, 2003.

[3] Ahmed Alaoui and Michael W Mahoney. Fast randomized kernel ridge regression with statistical guarantees. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[4] David M Allen. The relationship between variable selection and data agumentation and a method for prediction. *Technometrics*, 16(1):125–127, 1974.

[5] Robert Altmann, Patrick Henning, and Daniel Peterseim. Numerical homogenization beyond scale separation. *Acta Numerica*, 30:1–86, 2021.

[6] Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[7] Shun-ichi Amari. *Information Geometry and its Applications*, volume 194. Springer, 2016.

[8] Shun-ichi Amari and Si Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789, 1999.

[9] Sivaram Ambikasaran and Eric Darve. An $O(N \log N)$ fast direct solver for partial hierarchically semi-separable matrices: with application to radial basis function interpolation. *Journal of Scientific Computing*, 57:477–501, 2013.

[10] Sivaram Ambikasaran, Daniel Foreman-Mackey, Leslie Greengard, David W Hogg, and Michael O'Neil. Fast direct methods for Gaussian processes. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):252–265, 2015.

[11] Nihat Ay, Jürgen Jost, Hông Vân Lê, and Lorenz Schwachhöfer. Information geometry and sufficient statistics. *Probability Theory and Related Fields*, 162(1):327–364, 2015.

[12] Nihat Ay, Jürgen Jost, Hông V. Lê, and Lorenz Schwachhöfer. *Information Geometry*, volume 64. Springer, 2017.

[13] Abdul Kadir Aziz, R Bruce Kellogg, and Arthur Brooke Stephens. A two point boundary value problem with a rapidly oscillating solution. *Numerische Mathematik*, 53(1):107–121, 1988.

[14] Ivo Babuška, Gabriel Caloz, and John E Osborn. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM Journal on Numerical Analysis*, 31(4):945–981, 1994.

[15] Ivo Babuška and Robert Lipton. Optimal local approximation spaces for generalized finite element methods with application to multiscale problems. *Multiscale Modeling & Simulation*, 9(1):373–406, 2011.

[16] Ivo Babuška, Robert Lipton, Paul Sinz, and Michael Stuebner. Multiscale-spectral GFEM and optimal oversampling. *Computer Methods in Applied Mechanics and Engineering*, 364:112960, 2020.

[17] Ivo Babuška and John Osborn. Can a finite element method perform arbitrarily badly? *Mathematics of Computation*, 69(230):443–462, 2000.

[18] Ivo Babuška and John Osborn. Generalized finite element methods: their performance and their relation to mixed methods. *SIAM Journal on Numerical Analysis*, 20(3):510–536, 1983.

[19] Ivo Babuška and Stefan Sauter. Is the pollution effect of the fem avoidable for the Helmholtz equation considering high wave numbers? *SIAM Journal on numerical analysis*, 34(6):2392–2423, 1997.

[20] François Bachoc. Cross validation and maximum likelihood estimations of hyper-parameters of Gaussian processes with model misspecification. *Computational Statistics & Data Analysis*, 66:55–69, 2013.

[21] François Bachoc, Agnès Lagnoux, and Thi Mong Ngoc Nguyen. Cross-validation estimation of covariance parameters under fixed-domain asymptotics. *Journal of Multivariate Analysis*, 160:42–67, 2017.

[22] Yohai Bar-Sinai, Stephan Hoyer, Jason Hickey, and Michael P Brenner. Learning data-driven discretizations for partial differential equations. *Proceedings of the National Academy of Sciences*, 116(31):15344–15349, 2019.

[23] Pau Batlle, Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. Error analysis of kernel/GP methods for nonlinear and parametric PDEs. *arXiv preprint arXiv:2305.04962*, 2023.

[24] Martin Bauer, Martins Bruveris, and Peter W Michor. Uniqueness of the Fisher–Rao metric on the space of smooth densities. *Bulletin of the London Mathematical Society*, 48(3):499–506, 2016.

[25] Alain Berlinet and Christine Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.

[26] Maximilian Bernkopf, Théophile Chaumont-Frelet, and Jens Markus Melenk. Wavenumber-explicit stability and convergence analysis of hp finite element discretizations of helmholtz problems in piecewise smooth media. *arXiv preprint arXiv:2209.03601*, 2022.

[27] Timo Betcke, Simon N Chandler-Wilde, Ivan G Graham, Stephen Langdon, and Marko Lindner. Condition number estimates for combined potential integral operators in acoustics and their boundary element discretisation. *Numerical Methods for Partial Differential Equations*, 27(1):31–69, 2011.

[28] Gregory Beylkin, Ronald Coifman, and Vladimir Rokhlin. Fast wavelet transforms and numerical algorithms I. *Communications on pure and applied mathematics*, 44(2):141–183, 1991.

[29] Kaushik Bhattacharya, Bamdad Hosseini, Nikola B Kovachki, and Andrew M Stuart. Model reduction and neural networks for parametric PDEs. *The SMAI journal of computational mathematics*, 7:121–157, 2021.

[30] Vladimir Igorevich Bogachev. *Gaussian measures*. American Mathematical Society, 1998.

[31] Gabriele Boncoraglio and Charbel Farhat. Active manifold and model reduction for multidisciplinary analysis and optimization. In *AIAA Scitech 2021 Forum*, page 1694, 2021.

[32] Susanne C Brenner, L Ridgway Scott, and L Ridgway Scott. *The mathematical theory of finite element methods*, volume 3. Springer, 2008.

[33] Haim Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer Science & Business Media, 2010.

[34] Donald L Brown, Dietmar Gallistl, and Daniel Peterseim. Multiscale Petrov-Galerkin method for high-frequency heterogeneous Helmholtz equations. In *Meshfree methods for partial differential equations VIII*, pages 85–115. Springer, 2017.

[35] Andreas Buhr and Kathrin Smetana. Randomized local model order reduction. *SIAM journal on scientific computing*, 40(4):A2120–A2151, 2018.

[36] Daniele Calandriello, Alessandro Lazaric, and Michal Valko. Distributed adaptive sampling for kernel matrix approximation. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1421–1429, 2017.

[37] Jeff Calder. Consistency of lipschitz learning with infinite unlabeled data and finite labeled data. *SIAM Journal on Mathematics of Data Science*, 1(4):780–812, 2019.

[38] Jeff Calder and Dejan Slepčev. Properly-weighted graph laplacian for semi-supervised learning. *Applied Mathematics & Optimization*:1–49, 2019.

[39] Ben Calderhead, Mark Girolami, and Neil D Lawrence. Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 217–224, 2009.

[40] Nikolai Nikolaevich Cencov. *Statistical decision rules and optimal inference*. American Mathematical Soc., 2000.

[41] Ke Chen, Qin Li, Jianfeng Lu, and Stephen J Wright. Randomized sampling for basis function construction in generalized finite element methods. *Multiscale Modeling & Simulation*, 18(2):1153–1177, 2020.

[42] Yifan Chen, Ethan N Epperly, Joel A Tropp, and Robert J Webber. Randomly pivoted Cholesky: practical approximation of a kernel matrix with few entry evaluations. *arXiv preprint arXiv:2207.06503*, 2022.

[43] Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M. Stuart. Solving and learning nonlinear PDEs with Gaussian processes. *Journal of Computational Physics*, 447:110668, 2021. DOI: https://doi.org/10.1016/j.jcp.2021.110668.

[44] Yifan Chen and Thomas Y Hou. Function approximation via the subsampled Poincaré inequality. *Discrete and Continuous Dynamical Systems-A*, 2020. DOI: http://dx.doi.org/10.3934/dcds.2020296. URL: https://www.aimsciences.org/article/doi/10.3934/dcds.2020296.

[45] Yifan Chen and Thomas Y Hou. Multiscale elliptic PDE upscaling and function approximation via subsampled data. *Multiscale Modeling & Simulation*, 20(1):188–219, 2022. DOI: https://doi.org/10.1137/20M1372214. URL: https://epubs.siam.org/doi/10.1137/20M1372214.

[46] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponential convergence for multiscale linear elliptic PDEs via adaptive edge basis functions. *Multiscale Modeling & Simulation*, 19(2):980–1010, 2021. DOI: https://doi.org/10.1137/20m1352922. URL: https://epubs.siam.org/doi/10.1137/20M1352922.

[47] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponentially convergent multiscale finite element method. *Communications on Applied Mathematics and Computation*:1–17, 2023. URL: https://link.springer.com/article/10.1007/s42967-023-00260-2.

[48] Yifan Chen, Thomas Y Hou, and Yixuan Wang. Exponentially convergent multiscale methods for 2d high frequency heterogeneous Helmholtz equations. *To appear in Multiscale Modeling & Simulation, arXiv preprint arXiv:2105.04080*, 2021.

[49] Yifan Chen, Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Gradient flows for sampling: mean-field models, Gaussian approximations and affine invariance. *arXiv preprint arXiv:2302.11024*, 2023.

[50] Yifan Chen, Houman Owhadi, and Florian Schäfer. Sparse Cholesky factorization for solving nonlinear PDEs via Gaussian processes. *arXiv preprint arXiv:2304.01294*, 2023.

[51] Yifan Chen, Houman Owhadi, and Andrew M Stuart. Consistency of empirical Bayes and kernel flow for hierarchical parameter estimation. *Mathematics of Computation*, 2021. URL: https://www.ams.org/journals/mcom/2021-90-332/S0025-5718-2021-03649-2/.

[52] Eric T Chung, Yalchin Efendiev, and Wing Tat Leung. Constraint energy minimizing generalized multiscale finite element method. *Computer Methods in Applied Mechanics and Engineering*, 339:298–319, 2018.

[53] A Cochocki and Rolf Unbehauen. *Neural networks for optimization and signal processing*. John Wiley & Sons, Inc., 1993.

[54] Jon Cockayne, Chris Oates, Tim Sullivan, and Mark Girolami. Bayesian probabilistic numerical methods. *SIAM Review*, 61(4):756–789, 2019.

[55] Jon Cockayne, Chris Oates, Tim Sullivan, and Mark Girolami. Probabilistic numerical methods for PDE-constrained Bayesian inverse problems. In *AIP Conference Proceedings*, volume 1853 of number 1, page 060001, 2017.

[56] Corinna Cortes, Marius Kloft, and Mehryar Mohri. Learning kernels using local rademacher complexity. In *Advances in neural information processing systems*, pages 2760–2768, 2013.

[57] Simon L Cotter, Masoumeh Dashti, and Andrew M Stuart. Approximation of Bayesian inverse problems for PDEs. *SIAM Journal on Numerical Analysis*, 48(1):322–345, 2010.

[58] 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*:424–446, 2013.

[59] Jeremie Coullon and Robert J Webber. Ensemble sampler for infinite-dimensional inverse problems. *Statistics and Computing*, 31(3):1–9, 2021.

[60] Matthieu Darcy, Boumediene Hamzi, Giulia Livieri, Houman Owhadi, and Peyman Tavallali. One-shot learning of stochastic differential equations with data adapted kernels. *Physica D: Nonlinear Phenomena*, 444:133583, 2023.

[61] Masoumeh Dashti, Kody JH Law, Andrew M Stuart, and Jochen Voss. MAP estimators and their consistency in Bayesian nonparametric inverse problems. *Inverse Problems*, 29(9):095017, 2013.

[62] Masoumeh Dashti and Andrew M Stuart. The Bayesian approach to inverse problems. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*, pages 1–118. Springer International Publishing, 2016.

[63] Masoumeh Dashti and Andrew M Stuart. The Bayesian approach to inverse problems. *arXiv preprint arXiv:1302.6989*, 2013.

[64] Arka Daw, Jie Bu, Sifan Wang, Paris Perdikaris, and Anuj Karpatne. Rethinking the importance of sampling in physics-informed neural networks. *arXiv preprint arXiv:2207.02338*, 2022.

[65] Carl De Boor, Ronald A DeVore, and Amos Ron. Approximation from shift-invariant subspaces of $L^2(\mathbb{R}^d)$. *Transactions of the American Mathematical Society*, 341(2):787–806, 1994.

[66] Filip De Roos, Alexandra Gessner, and Philipp Hennig. High-dimensional Gaussian process inference with derivatives. In *International Conference on Machine Learning*, pages 2535–2545. PMLR, 2021.

[67] Tim De Ryck and Siddhartha Mishra. Error analysis for physics-informed neural networks (pinns) approximating kolmogorov pdes. *Advances in Computational Mathematics*, 48(6):1–40, 2022.

[68] Gianluca Detommaso, Tiangang Cui, Youssef Marzouk, Alessio Spantini, and Robert Scheichl. A Stein variational Newton method. *Advances in Neural Information Processing Systems*, 31, 2018.

[69] Persi Diaconis. Bayesian numerical analysis. *Statistical decision theory and related topics IV*, 1:163–175, 1988.

[70] Badis Djeridane and John Lygeros. Neural approximation of PDE solutions: an application to reachability computations. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 3034–3039. IEEE, 2006.

[71] Matthew M Dunlop, Marco A Iglesias, and Andrew M Stuart. Hierarchical bayesian level set inversion. *Statistics and Computing*, 27(6):1555–1584, 2017.

[72] Matthew M Dunlop and Georg Stadler. A gradient-free subspace-adjusting ensemble sampler for infinite-dimensional Bayesian inverse problems. *arXiv preprint arXiv:2202.11088*, 2022.

[73] Yalchin R Efendiev, Thomas Y Hou, and Xiao-Hui Wu. Convergence of a nonconforming multiscale finite element method. *SIAM Journal on Numerical Analysis*, 37(3):888–910, 2000.

[74] Yalchin Efendiev, Juan Galvis, and Thomas Y Hou. Generalized multiscale finite element methods (GMsFEM). *Journal of Computational Physics*, 251:116–135, 2013. ISSN: 0021-9991.

[75] Ahmed El Alaoui, Xiang Cheng, Aaditya Ramdas, Martin J Wainwright, and Michael I Jordan. Asymptotic behavior of l_p-based laplacian regularization in semi-supervised learning. In *Conference on Learning Theory*, pages 879–906, 2016.

[76] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.

[77] Björn Engquist and Lexing Ying. Sweeping preconditioner for the Helmholtz equation: hierarchical matrix representation. *Communications on pure and applied mathematics*, 64(5):697–735, 2011.

[78] Björn Engquist and Lexing Ying. Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers. *Multiscale Modeling & Simulation*, 9(2):686–710, 2011.

[79] Björn Engquist and Hongkai Zhao. Approximate separability of the green's function of the helmholtz equation in the high frequency limit. *Communications on Pure and Applied Mathematics*, 71(11):2220–2274, 2018.

[80] Christian Engwer, Patrick Henning, Axel Målqvist, and Daniel Peterseim. Efficient implementation of the localized orthogonal decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 350:123–153, 2019.

[81] David Eriksson, Kun Dong, Eric Lee, David Bindel, and Andrew G Wilson. Scaling Gaussian process regression with derivatives. *Advances in neural information processing systems*, 31, 2018.

[82] Sofi Esterhazy and Jens M Melenk. On stability of discretizations of the Helmholtz equation. In *Numerical analysis of multiscale problems*, pages 285–324. Springer, 2012.

[83] Lawrence C Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, 2010. ISBN: 978-0-8218-4974-3.

[84] Shai Fine and Katya Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2002. ISSN: 1532-4435.

[85] Daniel Foreman-Mackey, David W Hogg, Dustin Lang, and Jonathan Goodman. EMCEE: The MCMC hammer. *Publications of the Astronomical Society of the Pacific*, 125(925):306, 2013.

[86] Philip Freese, Moritz Hauck, and Daniel Peterseim. Super-localized orthogonal decomposition for high-frequency helmholtz problems. *arXiv preprint arXiv:2112.11368*, 2021.

[87] David Friedman. The functional equation f (x+ y)= g (x)+ h (y). *The American Mathematical Monthly*, 69(8):769–772, 1962.

[88] Thomas Friedrich. Die fisher-information und symplektische strukturen. *Mathematische Nachrichten*, 153(1):273–296, 1991. DOI: https://doi.org/10.1002/mana.19911530125.

[89] Shubin Fu, Eric Chung, and Guanglian Li. Edge multiscale methods for elliptic problems with heterogeneous coefficients. *Journal of Computational Physics*, 396:228–242, 2019.

[90] Shubin Fu and Kai Gao. A fast solver for the Helmholtz equation based on the generalized multiscale finite-element method. *Geophysical Journal International*, 211(2):797–813, 2017.

[91] Shubin Fu, Guanglian Li, Richard Craster, and Sebastien Guenneau. Wavelet-based edge multiscale finite element method for Helmholtz problems in perforated domains. *arXiv preprint arXiv:1906.08453*, 2019.

[92] Reinhard Furrer, Marc G Genton, and Douglas Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523, 2006.

[93] Dietmar Gallistl and Daniel Peterseim. Stable multiscale Petrov–Galerkin finite element method for high frequency acoustic scattering. *Computer Methods in Applied Mechanics and Engineering*, 295:1–17, 2015.

[94] Tingran Gao, Shahar Z Kovalsky, and Ingrid Daubechies. Gaussian process landmarking on manifolds. *SIAM Journal on Mathematics of Data Science*, 1(1):208–236, 2019.

[95] Alfredo Garbuno-Inigo, Franca Hoffmann, Wuchen Li, and Andrew M Stuart. Interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020.

[96] Alfredo Garbuno-Inigo, Nikolas Nüsken, and Sebastian Reich. Affine invariant interacting Langevin dynamics for Bayesian inference. *SIAM Journal on Applied Dynamical Systems*, 19(3):1633–1658, 2020.

[97] Seymour Geisser. The predictive sample reuse method with applications. *Journal of the American statistical Association*, 70(350):320–328, 1975.

[98] Jayanta K Ghosh and RV Ramamoorthi. *Bayesian Nonparametrics*. Springer Science & Business Media, 2003.

[99] David Gilbarg and Neil S Trudinger. *Elliptic partial differential equations of second order*. springer, 2015.

[100] D Gines, G Beylkin, and J Dunn. LU factorization of non-standard forms and direct multiresolution solvers. *Applied and Computational Harmonic Analysis*, 5(2):156–201, 1998.

[101] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1 of number 2. MIT press, 2016.

[102] Jonathan Goodman and Jonathan Weare. Ensemble samplers with affine invariance. *Communications in applied mathematics and computational science*, 5(1):65–80, 2010.

[103] I Graham and S Sauter. Stability and finite element error analysis for the Helmholtz equation with variable coefficients. *Mathematics of Computation*, 89(321):105–138, 2020.

[104] Ivan G Graham, Owen R Pembery, and Euan A Spence. The Helmholtz equation in heterogeneous media: a priori bounds, well-posedness, and resonances. *Journal of Differential Equations*, 266(6):2869–2923, 2019.

[105] Jens A Griepentrog and Lutz Recke. Linear elliptic boundary value problems with non–smooth data: normal solvability on Sobolev–Campanato spaces. *Mathematische Nachrichten*, 225(1):39–74, 2001.

[106] Tamara G Grossmann, Urszula Julia Komorowska, Jonas Latz, and Carola-Bibiane Schönlieb. Can physics-informed neural networks beat the finite element method? *arXiv preprint arXiv:2302.04107*, 2023.

[107] Ming Gu and Luiza Miranian. Strong rank revealing Cholesky factorization. *Electronic Transactions on Numerical Analysis*, 17:76–92, 2004.

[108] Joseph Guinness. Permutation and grouping methods for sharpening Gaussian process approximations. *Technometrics*, 60(4):415–429, 2018.

[109] Peter Guttorp and Tilmann Gneiting. Studies in the history of probability and statistics xlix on the matérn correlation family. *Biometrika*, 93(4):989–995, December 2006. ISSN: 0006-3444. DOI: 10.1093/biomet/93.4.989.

[110] Wolfgang Hackbusch. A sparse matrix arithmetic based on H-matrices. Part I: introduction to H-matrices. *Computing*, 62(2):89–108, 1999.

[111] Wolfgang Hackbusch and Steffen Börm. Data-sparse approximation by adaptive H 2-matrices. *Computing*, 69:1–35, 2002.

[112] Wolfgang Hackbusch and Boris N Khoromskij. A sparse H-matrix arithmetic, part II: application to multi-dimensional problems. *Computing*, 64(1):21–47, 2000.

[113] 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.

[114] Boumediene Hamzi and Houman Owhadi. Learning dynamical systems from data: a simple cross-validation perspective, part i: parametric kernel flows. *Physica D: Nonlinear Phenomena*, 421:132817, 2021.

[115] Jiequn Han, Arnulf Jentzen, and Weinan E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.

[116] Moritz Hauck and Daniel Peterseim. Multi-resolution localized orthogonal decomposition for Helmholtz problems. *arXiv preprint arXiv:2104.11190*, 2021.

[117] Moritz Hauck and Daniel Peterseim. Super-localization of elliptic multiscale problems. *arXiv preprint arXiv:2107.13211*, 2021.

[118] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[119] P. Hennig, M. A. Osborne, and M. Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A*, 471(2179):20150142, 2015. ISSN: 1364-5021.

[120] Philipp Hennig, Michael A Osborne, and Mark Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142, 2015.

[121] Patrick Henning and Daniel Peterseim. Oversampling for the multiscale finite element method. *Multiscale Modeling & Simulation*, 11(4):1149–1175, 2013.

[122] Ulrich Hetmaniuk and Axel Klawonn. Error estimates for a two-dimensional special finite element method based on component mode synthesis. *Electron. Trans. Numer. Anal*, 41:109–132, 2014.

[123] Ulrich Hetmaniuk and Richard Lehoucq. A special finite element method based on component mode synthesis. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(3):401–420, 2010.

[124] Nicholas J Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, 2nd ed edition, 2002. DOI: 10.1137/1.9780898718027.

[125] Michael Hinze, René Pinnau, Michael Ulbrich, and Stefan Ulbrich. *Optimization with PDE constraints*, volume 23. Springer Science & Business Media, 2008.

[126] Nils Lid Hjort, Chris Holmes, Peter Müller, and Stephen G Walker. *Bayesian Nonparametrics*, volume 28. Cambridge University Press, 2010.

[127] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The Annals of Statistics*:1171–1220, 2008.

[128] Thomas Y Hou and Pengfei Liu. Optimal local multi-scale basis functions for linear elliptic equations with rough coefficient. *Discrete and Continuous Dynamical Systems*, 36(8):4451–4476, 2016.

[129] Thomas Y Hou and Xiao-Hui Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, 134(1):169–189, 1997. ISSN: 0021-9991.

[130] Thomas Y Hou, Xiao-Hui Wu, and Zhiqiang Cai. Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Mathematics of computation*, 68(227):913–943, 1999.

[131] Thomas Y Hou and Pengchuan Zhang. Sparse operator compression of higher-order elliptic operators with rough coefficients. *Research in the Mathematical Sciences*, 4:1–49, 2017.

[132] Daniel Zhengyu Huang, Jiaoyang Huang, Sebastian Reich, and Andrew M Stuart. Efficient derivative-free bayesian inference for large-scale inverse problems. *Inverse Problems*, 38(12):125006, 2022.

[133] Thomas JR Hughes, Gonzalo R Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale method—a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166(1):3–24, 1998. ISSN: 0045-7825.

[134] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

[135] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the Fokker–Planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[136] Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.

[137] G. Karniadakis, I. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 2021.

[138] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[139] Matthias Katzfuss. A multi-resolution approximation for massive spatial datasets. *Journal of the American Statistical Association*, 112(517):201–214, 2017.

[140] Matthias Katzfuss, Joseph Guinness, Wenlong Gong, and Daniel Zilber. Vecchia approximations of Gaussian-process predictions. *Journal of Agricultural, Biological and Environmental Statistics*, 25:383–414, 2020.

[141] G. S. Kimeldorf and G. Wahba. A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *Annals of Mathematical Statistics*, 41:495–502, 1970. ISSN: 0003-4851.

[142] George S Kimeldorf and Grace Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.

[143] George Kimeldorf and Grace Wahba. Some results on Tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95, 1971.

[144] Bartek T Knapik, BT Szabó, Aad W Van Der Vaart, and JH Van Zanten. Bayes procedures for adaptive inference in inverse problems for the white noise model. *Probability Theory and Related Fields*, 164(3-4):771–813, 2016.

[145] Bartek T Knapik, Aad W Van Der Vaart, and J Harry van Zanten. Bayesian inverse problems with Gaussian priors. *The Annals of Statistics*, 39(5):2626–2657, 2011.

[146] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14 of number 2, pages 1137–1145. Montreal, Canada, 1995.

[147] Robert Kohn, Craig F Ansley, and David Tharm. The performance of cross-validation and maximum likelihood estimators of spline smoothing parameters. *Journal of the american statistical association*, 86(416):1042–1050, 1991.

[148] Ralf Kornhuber, Daniel Peterseim, and Harry Yserentant. An analysis of a class of variational multiscale methods based on subspace decomposition. *Mathematics of Computation*, 87(314):2765–2774, 2018.

[149] K Krischer, R Rico-Martinez, IG Kevrekidis, HH Rotermund, G Ertl, and JL Hudson. Model identification of a spatiotemporally varying catalytic reaction. *AIChE Journal*, 39(1):89–98, 1993.

[150] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in Neural Information Processing Systems*, 34:26548–26560, 2021.

[151] Rasmus Kyng, Anup Rao, Sushant Sachdeva, and Daniel A Spielman. Algorithms for lipschitz learning on graphs. In *Conference on Learning Theory*, pages 1190–1223, 2015.

[152] Kenneth L. Ho and Lexing Ying. Hierarchical interpolative factorization for elliptic operators: integral equations. *Communications on Pure and Applied Mathematics*, 69(7):1314–1353, 2016.

[153] David Lafontaine, Euan A Spence, and Jared Wunsch. For most frequencies, strong trapping has a weak effect in frequency-domain scattering. *arXiv preprint arXiv:1903.12172*, 2019.

[154] David Lafontaine, Euan A Spence, and Jared Wunsch. Wavenumber-explicit convergence of the hp-fem for the full-space heterogeneous helmholtz equation with smooth coefficients. *Computers & Mathematics with Applications*, 113:59–69, 2022.

[155] Isaac E Lagaris, Aristidis C Likas, and Dimitrios I Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *IEEE transactions on Neural Networks*, 9(5):987–1000, 1998.

[156] Isaac E Lagaris, Aristidis C Likas, and Dimitris G Papageorgiou. Neural-network methods for boundary value problems with irregular boundaries. *IEEE Transactions on Neural Networks*, 11(5):1041–1049, 2000.

[157] FM Larkin. Gaussian measure in hilbert space and applications in numerical analysis. *The Rocky Mountain Journal of Mathematics*:379–421, 1972.

[158] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as Gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

[159] Benedict Leimkuhler, Charles Matthews, and Jonathan Weare. Ensemble preconditioning for markov chain monte carlo simulation. *Statistics and Computing*, 28(2):277–290, 2018.

[160] Guanglian Li. On the convergence rates of GMsFEMs for heterogeneous elliptic problems without oversampling techniques. *Multiscale Modeling & Simulation*, 17(2):593–619, 2019.

[161] Guanglian Li, Daniel Peterseim, and Mira Schedensack. Error analysis of a variational multiscale stabilization for convection-dominated diffusion equations in two dimensions. *IMA Journal of Numerical Analysis*, 38(3):1229–1253, 2018.

[162] Shengguo Li, Ming Gu, Cinna Julie Wu, and Jianlin Xia. New efficient and robust HSS Cholesky factorization of SPD matrices. *SIAM Journal on Matrix Analysis and Applications*, 33(3):886–904, 2012.

[163] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[164] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Andrew Stuart, Kaushik Bhattacharya, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. *Advances in Neural Information Processing Systems*, 2020.

[165] Hua Liang and Hulin Wu. Parameter estimation for differential equation models using a framework of measurement error in regression models. *Journal of the American Statistical Association*, 103(484):1570–1583, 2008.

[166] Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011.

[167] Haitao Liu, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. When gaussian process meets big data: a review of scalable GPs. *IEEE transactions on neural networks and learning systems*, 31(11):4405–4423, 2020.

[168] Qiang Liu. Stein variational gradient descent as gradient flow. *Advances in neural information processing systems*, 30, 2017.

[169] Qiang Liu and Dilin Wang. Stein variational gradient descent: a general purpose Bayesian inference algorithm. *Advances in neural information processing systems*, 29, 2016.

[170] Ziming Liu, Andrew M Stuart, and Yixuan Wang. Second order ensemble langevin method for sampling and inverse problems. *arXiv preprint arXiv:2208.04506*, 2022.

[171] Zichao Long, Yiping Lu, Xianzhong Ma, and Bin Dong. PDE-net: learning PDEs from data. In *International Conference on Machine Learning*, pages 3208–3216. PMLR, 2018.

[172] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.

[173] Tzon-Tzer Lu and Sheng-Hua Shiou. Inverses of 2× 2 block matrices. *Computers & Mathematics with Applications*, 43(1-2):119–129, 2002.

[174] Yiping Lu, Haoxuan Chen, Jianfeng Lu, Lexing Ying, and Jose Blanchet. Machine learning for elliptic pdes: fast rate generalization bound, neural scaling law and minimax optimality. *arXiv preprint arXiv:2110.06897*, 2021.

[175] Yulong Lu, Jianfeng Lu, and James Nolen. Accelerating Langevin sampling with birth-death. *arXiv preprint arXiv:1905.09863*, 2019.

[176] Yulong Lu, Dejan Slepčev, and Lihan Wang. Birth-death dynamics for sampling: global convergence, approximations and their asymptotics. *arXiv preprint arXiv:2211.00450*, 2022.

[177] Chupeng Ma, Christian Alber, and Robert Scheichl. Wavenumber explicit convergence of a multiscale gfem for heterogeneous helmholtz problems. *arXiv preprint arXiv:2112.10544*, 2021.

[178] Chupeng Ma and Robert Scheichl. Error estimates for fully discrete generalized fems with locally optimal spectral approximations. *arXiv preprint arXiv:2107.09988*, 2021.

[179] Chupeng Ma, Robert Scheichl, and Tim Dodwell. Novel design and analysis of generalized fe methods based on locally optimal spectral approximations. *arXiv preprint arXiv:2103.09545*, 2021.

[180] Roland Maier. A high-order approach to elliptic multiscale problems with general unstructured coefficients. *SIAM Journal on Numerical Analysis*, 59(2):1067–1089, 2021.

[181] Axel Målqvist and Daniel Peterseim. Localization of elliptic multiscale problems. en. *Mathematics of Computation*, 83(290):2583–2603, June 2014. ISSN: 0025-5718, 1088-6842. (Visited on 08/23/2019).

[182] James Martin, Lucas C Wilcox, Carsten Burstedde, and Omar Ghattas. A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion. *SIAM Journal on Scientific Computing*, 34(3):A1460–A1487, 2012.

[183] Per-Gunnar Martinsson and Joel A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020. DOI: 10.1017/S0962492920000021.

[184] Jens M Melenk and Ivo Babuška. The partition of unity finite element method: basic theory and applications. *Computer methods in applied mechanics and engineering*, 139(1-4):289–314, 1996.

[185] Jens M Melenk and Stefan Sauter. Convergence analysis for finite element discretizations of the Helmholtz equation with Dirichlet-to-Neumann boundary conditions. *Mathematics of Computation*, 79(272):1871–1914, 2010.

[186] Jens M Melenk and Stefan Sauter. Wavenumber explicit convergence analysis for Galerkin discretizations of the Helmholtz equation. *SIAM Journal on Numerical Analysis*, 49(3):1210–1243, 2011.

[187] Rui Meng and Xianjin Yang. Sparse Gaussian processes for solving nonlinear PDEs. *arXiv preprint arXiv:2205.03760*, 2022.

[188] Charles A Micchelli and Theodore J Rivlin. *A survey of optimal recovery*. Springer, 1977.

[189] Victor Minden, Kenneth L Ho, Anil Damle, and Lexing Ying. A recursive skeletonization factorization based on strong admissibility. *Multiscale Modeling & Simulation*, 15(2):768–796, 2017.

[190] Andrea Moiola and Euan A Spence. Acoustic transmission problems: wavenumber-explicit bounds and resonance-free regions. *Mathematical Models and Methods in Applied Sciences*, 29(02):317–354, 2019.

[191] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[192] Cameron Musco and Christopher Musco. Recursive sampling for the Nyström method. *Advances in neural information processing systems*, 30, 2017.

[193] Boaz Nadler, Nathan Srebro, and Xueyuan Zhou. Semi-supervised learning with the graph laplacian: the limit of infinite unlabelled data. In *Advances in neural information processing systems 22*, pages 1330–1338, 2009.

[194] Radford M Neal. Priors for infinite networks. *Bayesian learning for neural networks*:29–53, 1996.

[195] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[196] Nicholas H Nelsen and Andrew M Stuart. The random feature model for input-output maps between Banach spaces. *SIAM Journal on Scientific Computing*, 43(5):A3212–A3243, 2021.

[197] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer Science & Business Media, 2006.

[198] Assad A Oberai and Peter M Pinsky. A multiscale finite element method for the Helmholtz equation. *Computer Methods in Applied Mechanics and Engineering*, 154(3-4):281–297, 1998.

[199] Mario Ohlberger and Barbara Verfurth. A new heterogeneous multiscale method for the Helmholtz equation with high contrast. *Multiscale Modeling & Simulation*, 16(1):385–411, 2018.

[200] Felix Otto. The geometry of dissipative evolution equations: the porous medium equation. *Comm. Partial Differential Equations*, 26:101–174, 2001.

[201] Houman Owhadi. Bayesian numerical homogenization. *Multiscale Modeling & Simulation*, 13(3):812–828, 2015.

[202] Houman Owhadi. Do ideas have shape? plato's theory of forms as the continuous limit of artificial neural networks. arXiv preprint arXiv:2008.03920, 2020.

[203] Houman Owhadi. Multigrid with rough coefficients and multiresolution operator decomposition from hierarchical information games. *SIAM Review*, 59(1):99–149, 2017.

[204] Houman Owhadi and Clint Scovel. *Operator-Adapted Wavelets, Fast Solvers, and Numerical Homogenization: From a Game Theoretic Approach to Numerical Approximation and Algorithm Design*, volume 35. Cambridge University Press, 2019.

[205] Houman Owhadi, Clint Scovel, and Florian Schäfer. Statistical numerical approximation. *Notices of the AMS*, 2019.

[206] Houman Owhadi and Gene Ryan Yoo. Kernel flows: from learning kernels from data into the abyss. *Journal of Computational Physics*, 389:22–47, 2019.

[207] Houman Owhadi and Lei Zhang. Gamblets for opening the complexity-bottleneck of implicit schemes for hyperbolic and parabolic ODEs/PDEs with rough coefficients. *Journal of Computational Physics*, 347:99–128, 2017.

[208] Houman Owhadi and Lei Zhang. Localized bases for finite-dimensional homogenization approximations with nonseparated scales and high contrast. *Multiscale Modeling & Simulation*, 9(4):1373–1398, 2011.

[209] Houman Owhadi and Lei Zhang. Metric-based upscaling. *Communications on Pure and Applied Mathematics*, 60(5):675–723, 2007.

[210] Houman Owhadi, Lei Zhang, and Leonid Berlyand. Polyharmonic homogenization, rough polyharmonic splines and sparse super-localization. *ESAIM: Mathematical Modelling and Numerical Analysis*, 48(2):517–552, 2014.

[211] Misha Padidar, Xinran Zhu, Leo Huang, Jacob Gardner, and David Bindel. Scaling Gaussian processes with derivative information using variational inference. *Advances in Neural Information Processing Systems*, 34:6442–6453, 2021.

[212] I. Palasti and A. Renyi. On interpolation theory and the theory of games. *MTA Mat. Kat. Int. Kozl*, 1:529–540, 1956.

[213] O Perrin and P Monestiez. Modelling of non-stationary spatial structure using parametric radial basis deformations. In *GeoENV II—Geostatistics for Environmental Applications*, pages 175–186. Springer, 1999.

[214] Daniel Peterseim. Eliminating the pollution effect in Helmholtz problems by local subscale correction. *Mathematics of Computation*, 86(305):1005–1036, 2017.

[215] Daniel Peterseim and Barbara Verfürth. Computational high frequency scattering from high-contrast heterogeneous media. *Mathematics of Computation*, 89(326):2649–2674, 2020.

[216] Jakiw Pidstrigach and Sebastian Reich. Affine-invariant ensemble transform methods for logistic regression. *arXiv preprint arXiv:2104.08061*, 2021.

[217] H. Poincaré. *Calcul des probabilités*. Georges Carrés, 1896.

[218] Jack Poulson, Bjorn Engquist, Siwei Li, and Lexing Ying. A parallel sweeping preconditioner for heterogeneous 3D Helmholtz equations. *SIAM Journal on Scientific Computing*, 35(3):C194–C212, 2013.

[219] Joaquin Quinonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate Gaussian process regression. *The Journal of Machine Learning Research*, 6:1939–1959, 2005.

[220] Vicenţiu D Rădulescu. *Qualitative analysis of nonlinear elliptic partial differential equations: monotonicity, analytic, and variational methods*. Hindawi Publishing Corporation, 2008.

[221] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

[222] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[223] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Inferring solutions of differential equations using noisy multi-fidelity data. *Journal of Computational Physics*, 335:736–746, 2017.

[224] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Numerical Gaussian processes for time-dependent and nonlinear partial differential equations. *SIAM Journal on Scientific Computing*, 40(1):A172–A198, 2018.

[225] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations. *Science*, 367(6481):1026–1030, 2020.

[226] Jim O Ramsay, Giles Hooker, David Campbell, and Jiguo Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007.

[227] C Radhakrishna Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Reson. J. Sci. Educ*, 20:78–90, 1945.

[228] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer, 2003.

[229] Ramiro Rico-Martinez and Ioannis G Kevrekidis. Continuous time modeling of nonlinear systems: a neural network-based approach. In *IEEE International Conference on Neural Networks*, pages 1522–1525. IEEE, 1993.

[230] Lassi Roininen, Markku S Lehtinen, Sari Lasanen, Mikko Orispää, and Markku Markkanen. Correlation priors. *Inverse problems and imaging*, 5(1):167–184, 2011.

[231] Amos Ron. The L^2-Approximation Orders of Principal Shift-Invariant Spaces Generated by a Radial Basis Function. en. In *Numerical Methods in Approximation Theory, Vol. 9*, pages 245–268. Birkhäuser Basel, Basel, 1992. ISBN: 978-3-0348-8619-2. DOI: 10.1007/978-3-0348-8619-2_14. (Visited on 09/23/2019).

[232] Alessandro Rudi, Daniele Calandriello, Luigi Carratino, and Lorenzo Rosasco. On fast leverage score sampling and optimal learning. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[233] Paul D Sampson and Peter Guttorp. Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, 87(417):108–119, 1992.

[234] Huiyan Sang and Jianhua Z Huang. A full scale approximation of covariance functions for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(1):111–132, 2012.

[235] Daniel Sanz-Alonso and Ruiyi Yang. Finite element representations of gaussian processes: balancing numerical and statistical accuracy. *SIAM/ASA Journal on Uncertainty Quantification*, 10(4):1323–1349, 2022.

[236] Daniel Sanz-Alonso and Ruiyi Yang. The SPDE approach to Matérn fields: graph representations. *Statistical Science*, 37(4):519–540, 2022.

[237] Arthur Sard. *Linear approximation*, number 9. American Mathematical Soc., 1963.

[238] Stefan Sauter and Céline Torres. Stability estimate for the Helmholtz equation with rapidly jumping coefficients. *Zeitschrift für angewandte Mathematik und Physik*, 69(6):139, 2018.

[239] Robert Schaback and Holger Wendland. Kernel techniques: from machine learning to meshless methods. *Acta numerica*, 15:543–639, 2006.

[240] Florian Schäfer, Matthias Katzfuss, and Houman Owhadi. Sparse cholesky factorization by Kullback–Leibler minimization. *SIAM Journal on Scientific Computing*, 43(3):A2019–A2046, 2021.

[241] Florian Schäfer, Timothy John Sullivan, and Houman Owhadi. Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity. *Multiscale Modeling & Simulation*, 19(2):688–730, 2021.

[242] Michael Scheuerer, Robert Schaback, and Martin Schlather. Interpolation of spatial data–a stochastic or a deterministic problem? *European Journal of Applied Mathematics*, 24(4):601–629, 2013.

[243] Julia Schleuß and Kathrin Smetana. Optimal local approximation spaces for parabolic problems. *arXiv preprint arXiv:2012.02759*, 2020.

[244] Alexandra M Schmidt and Anthony O'Hagan. Bayesian inference for non-stationary spatial covariance structure via spatial deformations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(3):743–758, 2003.

[245] M Schober, D Duvenaud, and P Hennig. Probabilistic ODE solvers with Runge-Kutta means. In *28th Annual Conference on Neural Information Processing Systems*, pages 739–747, 2014.

[246] Bernhard Scholkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT Press, 2018.

[247] Bernhard Schölkopf, Alexander J Smola, and Francis Bach. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[248] Zuoqiang Shi, Stanley Osher, and Wei Zhu. Weighted nonlocal laplacian on interpolation from sparse data. *Journal of Scientific Computing*, 73(2-3):1164–1177, 2017.

[249] Yeonjong Shin, Jerome Darbon, and George Em Karniadakis. On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type pdes. *arXiv preprint arXiv:2004.01806*, 2020.

[250] Justin Sirignano and Konstantinos Spiliopoulos. DGM: a deep learning algorithm for solving partial differential equations. *Journal of Computational Physics*, 375:1339–1364, 2018.

[251] John Skilling. Bayesian solution of ordinary differential equations. In C. R. Smith, G. J. Erickson, and P. O. Neudorfer, editors, *Maximum entropy and Bayesian methods*, pages 23–37. Springer, 1992.

[252] Dejan Slepcev and Matthew Thorpe. Analysis of p-laplacian regularization in semisupervised learning. *SIAM Journal on Mathematical Analysis*, 51(3):2085–2120, 2019.

[253] Kathrin Smetana and Anthony T Patera. Optimal local approximation spaces for component-based static condensation procedures. *SIAM Journal on Scientific Computing*, 38(5):A3318–A3356, 2016.

[254] Joel Smoller. *Shock Waves and Reaction—Diffusion Equations*. Springer Science & Business Media, 2012.

[255] Anuj Srivastava, Ian Jermyn, and Shantanu Joshi. Riemannian analysis of probability density functions with applications in vision. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[256] Michael L Stein. 2010 Rietz lecture: when does the screening effect hold? *The Annals of Statistics*, 39(6):2795–2819, 2011.

[257] Michael L Stein. A comparison of generalized cross validation and modified maximum likelihood for estimating the parameters of a stochastic process. *The Annals of Statistics*:1139–1157, 1990.

[258] Michael L Stein. *Interpolation of spatial data: some theory for kriging*. Springer Science & Business Media, 1999.

[259] Michael L Stein. The screening effect in kriging. *The Annals of Statistics*, 30(1):298–323, 2002.

[260] Charles J Stone et al. An asymptotically optimal window selection rule for kernel density estimates. *The Annals of Statistics*, 12(4):1285–1297, 1984.

[261] Andrew M Stuart. Inverse problems: a Bayesian perspective. *Acta numerica*, 19:451–559, 2010.

[262] Andrew Stuart and Aretha Teckentrup. Posterior consistency for Gaussian process approximations of Bayesian posterior distributions. *Mathematics of Computation*, 87(310):721–753, 2018.

[263] Al'bert Valentinovich Sul'din. Wiener measure and its applications to approximation methods. i. *Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika*, (6):145–158, 1959.

[264] Laura P Swiler, Mamikon Gulian, Ari L Frankel, Cosmin Safta, and John D Jakeman. A survey of constrained Gaussian process regression: approaches and implementation challenges. *Journal of Machine Learning for Modeling and Computing*, 1(2), 2020.

[265] Eitan Tadmor. A review of numerical methods for nonlinear partial differential equations. *Bulletin of the American Mathematical Society*, 49(4):507–554, 2012.

[266] Luc Tartar. *An introduction to Sobolev spaces and interpolation spaces*, volume 3. Springer Science & Business Media, 2007.

[267] Aretha L Teckentrup. Convergence of Gaussian process regression with estimated hyper-parameters and applications in Bayesian inverse problems. *arXiv preprint arXiv:1909.00232*, 2019.

[268] J. F. Traub, G. W. Wasilkowski, and H. Woźniakowski. *Information-based complexity*. Computer Science and Scientific Computing. Academic Press, Inc., Boston, MA, 1988, pages xiv+523. ISBN: 0-12-697545-0. With contributions by A. G. Werschulz and T. Boult.

[269] Joseph F Traub, Grzegorz W. Wasilkowski, and H Woźniakowski. Average case optimality for linear problems. *Theoretical Computer Science*, 29(1-2):1–25, 1984.

[270] Joseph F Traub, GW Wasilkowski, H Wozniakowski, and Erich Novak. Information-based complexity. *SIAM Review*, 36(3):514–514, 1994.

[271] N. García Trillos, B. Hosseini, and D. Sanz-Alonso. From optimization to sampling through gradient flows. *arXiv preprint arXiv:2302.11449*, 2023.

[272] Tadasu Uchiyama and Noboru Sonehara. Solving inverse problems in nonlinear PDEs by recurrent neural networks. In *IEEE International Conference on Neural Networks*, pages 99–102, 1993.

[273] Aad W Van der Vaart, Sandrine Dudoit, and Mark J van der Laan. Oracle inequalities for multi-fold cross validation. *Statistics & Decisions*, 24(3):351–371, 2006.

[274] Mark J van der Laan, Sandrine Dudoit, and Aad W van der Vaart. The cross-validated adaptive epsilon-net estimator. *Statistics & Decisions*, 24(3):373–395, 2006.

[275] Remco van der Meer, Cornelis Oosterlee, and Anastasia Borovykh. Optimally weighted loss functions for solving PDEs with neural networks. arXiv preprint arXiv:2002.06269, 2020.

[276] Aad W van der Vaart and J Harry van Zanten. Reproducing kernel Hilbert spaces of Gaussian priors. In *Pushing the limits of contemporary statistics: contributions in honor of Jayanta K. Ghosh*, pages 200–222. Institute of Mathematical Statistics, 2008.

[277] Aad van der Vaart and Jon A Wellner. *Weak Convergence And Empirical Processes*. 1996.

[278] Aldo V Vecchia. Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):297–312, 1988.

[279] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[280] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[281] Grace Wahba and James Wendelberger. Some new mathematical methods for variational objective analysis using splines and cross validation. *Monthly weather review*, 108(8):1122–1143, 1980.

[282] Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

[283] Sifan Wang, Hanwen Wang, and Paris Perdikaris. On the eigenvector bias of Fourier feature networks: from regression to solving multi-scale PDEs with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113938, 2021.

[284] Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why PINNs fail to train: a neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022.

[285] Yifei Wang and Wuchen Li. Information Newton's flow: second-order optimization method in probability space. *arXiv preprint arXiv:2001.04341*, 2020.

[286] JJ Warnes and BD Ripley. Problems with likelihood estimation of covariance functions of spatial Gaussian processes. *Biometrika*, 74(3):640–642, 1987.

[287] E Weinan and Bing Yu. The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. *Communications in Mathematics and Statistics*, 6(1):1–12, 2018.

[288] Holger Wendland. *Scattered data approximation*, volume 17. Cambridge university press, 2004.

[289] Peter Whittle. On stationary processes in the plane. *Biometrika*:434–449, 1954.

[290] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[291] Christopher Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. *Advances in neural information processing systems*, 13, 2000.

[292] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pages 370–378. PMLR, 2016.

[293] Andrew Wilson and Hannes Nickisch. Kernel interpolation for scalable structured Gaussian processes (KISS-GP). In *International conference on machine learning*, pages 1775–1784. PMLR, 2015.

[294] Jian Wu, Matthias Poloczek, Andrew G Wilson, and Peter Frazier. Bayesian optimization with gradients. *Advances in neural information processing systems*, 30, 2017.

[295] Ang Yang, Cheng Li, Santu Rana, Sunil Gupta, and Svetha Venkatesh. Sparse approximation for Gaussian process with derivative observations. In *AI 2018: Advances in Artificial Intelligence: 31st Australasian Joint Conference, Wellington, New Zealand, December 11-14, 2018, Proceedings*, pages 507–518. Springer, 2018.

[296] Yuhong Yang et al. Consistency of cross validation for comparing regression procedures. *The Annals of Statistics*, 35(6):2450–2473, 2007.

[297] Zhiliang Ying. Asymptotic properties of a maximum likelihood estimator with data from a gaussian process. *Journal of Multivariate Analysis*, 36(2):280–296, 1991.

[298] Gene Ryan Yoo and Houman Owhadi. Deep regularization and direct training of the inner layers of neural networks with kernel flows. *arXiv preprint arXiv:2002.08335*, 2020.

[299] Qi Zeng, Yash Kothari, Spencer H Bryngelson, and Florian Tobias Schaefer. Competitive physics informed networks. In *The Eleventh International Conference on Learning Representations*, 2023.

[300] Hao Zhang, Yong Wang, et al. Kriging and cross-validation for massive spatial data. *Environmetrics*, 21(3/4):290–304, 2010.

[301] Xiong Zhang, Kang Zhu Song, Ming Wan Lu, and X Liu. Meshless methods based on collocation with radial basis functions. *Computational mechanics*, 26:333–343, 2000.

[302] Xueyuan Zhou and Mikhail Belkin. Semi-supervised learning by higher order regularization. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 892–900, 2011.

[303] Yinhao Zhu and Nicholas Zabaras. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366:415–447, 2018.

Appendix A

APPENDIX TO CHAPTER IV

A.1 Diagonal Regularization of Kernel Matrices

In the context of GP regression it is common to regularize the kernel matrices by addition of a small diagonal term; in that context, doing so has the interpretation of assuming small Gaussian noise on top of the observations. This diagonal regularization is sometimes referred to as a "nugget". Here we discuss a related approach to regularizing kernel matrices (Θ and $\widetilde{\Theta}$) by perturbing them slightly; for brevity we use the terminology "nugget" throughout. In Appendix A.1.1 we present an adaptive approach to constructing a family of nugget terms that is tailored to our kernel matrices. Appendices A.1.2 through A.1.4 gather the detailed choices of nugget terms for the experiments in Subsections 4.3.5.1 through 4.3.5.3. Appendix A.1.5 contains the same details for the experiments in Subsection 4.4.4.

A.1.1 An Adaptive Nugget Term

One of the main computational issues in implementing our methodology is that the kernel matrix Θ is ill-conditioned. As a consequence, we need to regularize this matrix to improve the stability of these algorithms. This regularization may introduce an accuracy floor, so it is important to choose the regularization term so that it has a small effect on accuracy—there is thus an accuracy-stability tradeoff. A traditional strategy for achieving this goal is to add a nugget term ηI to Θ, where $\eta > 0$ is a small number, and I is the identity matrix. By choosing a suitable η, the condition number of $\Theta + \eta I$ can be improved significantly. However, there is an additional level of difficulty in our method since the matrix Θ contains multiple blocks whose spectral properties can differ by orders of magnitude, since they can involve different orders of derivatives of the kernel function K. This observation implies that adding ηI, which is uniform across all blocks, may be suboptimal in terms of the accuracy-stability tradeoff.

In what follows we adopt the same notation as Subsection 4.3.4.1. To address the ill-conditioning of Θ, we consider adding an adaptive block diagonal nugget term. More precisely, without loss of generality we can assume $\Theta^{(1,1)}$ corresponds to the

pointwise measurements, i.e., $L_1^{\mathbf{x}} = \delta_{\mathbf{x}}$, then, we construct a block diagonal matrix

$$R = \begin{bmatrix} I & & & \\ & \frac{\operatorname{tr}(\Theta^{(2,2)})}{\operatorname{tr}(\Theta^{(1,1)})} I & & \\ & & \ddots & \\ & & & \frac{\operatorname{tr}(\Theta^{(Q,Q)})}{\operatorname{tr}(\Theta^{(1,1)})} I \end{bmatrix},$$

where we reweight the identity matrix in each diagonal block by a factor of the trace ratio between $\Theta^{(q,q)}$ and $\Theta^{(1,1)}$. With this matrix, the adaptive nugget term is defined as ηR with a global nugget parameter $\eta > 0$. We find that once the parameter η is chosen suitably, then our Gauss–Newton algorithm converges quickly and in a stable manner. During computations, we can compute the Cholesky factors of $\Theta + \eta R$ offline and use back-substitution to invert them in each iteration of Gauss–Newton.

Example NE. *For the numerical experiments in Subsection 4.1.1.4 pertaining to the nonlinear elliptic PDE* (4.1.1)*, we observed that* Θ *has two distinct diagonal blocks, i.e., one block corresponding to the pointwise evaluation functions and with entries* $K(\mathbf{x}_m, \mathbf{x}_i)$ *and another block corresponding to pointwise evaluations of the Laplacian operator and with entries* $\Delta^{\mathbf{x}}\Delta^{\mathbf{x}'}K(\mathbf{x}, \mathbf{x}')|_{(\mathbf{x}_m, \mathbf{x}_i)}$*. With* $M = 1024$ *collocation points, the trace ratio between these blocks was of order* 4000*. Thus, the difference between* I *and* R *is significant. Our experiments also showed that if we only add* ηI *to regularize the matrix, then choosing* η *as large as* $O(10^{-4})$ *was needed to get meaningful results, while using the nugget term* ηR*, we could choose* $\eta = 10^{-9}$ *which leads to significantly improved results. We further explore these details below and in particular in Table A.1.2.* ◇

A.1.2 Choice of Nugget Terms for the Nonlinear Elliptic PDE

Below we discuss the choice of the nugget term in the numerical experiments of Subsection 4.3.5.1. The results in Figure 4.2 and Table 4.1 were obtained by employing the adaptive nugget term of Appendix A.1.1 with global parameters $\eta = 10^{-13}$ and $\eta = 10^{-12}$ respectively.

We also compared our adaptive nugget term to more standard choices, i.e., nugget terms of the form ηI against our adaptive nugget term ηR with the nonlinearity $\tau(u) = u^3$. Cholesky factorization was applied to the regularized matrix and the subsequent Gauss–Newton iterations were employed to obtain the solutions. The L^2 and L^∞ errors of the converged solutions are shown in Table A.1.2. The results were averaged over 10 instances of a random sampling of the collocation points.

Here, "nan" means the algorithm was unstable and diverged. To obtain these results we terminated all Gauss-Newton iterations after 5 steps. Due to randomness in the initial guess, and in examples where random collocation points were used due to this too, we observed that some random trials did not converge in 5 steps. This variation also explains the non-monotonic behavior of the error in Table A.1.2 as η decreases. These effects were more profound for the standard nugget term. Besides these small variations our results clearly demonstrate the superior accuracy and stability that is provided by our adaptive nugget term versus the standard nugget choice.

η	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
$\Theta + \eta I$: L^2 error	7.77e-02	4.46e-02	2.65e-02	1.56e-02	1.32e-02	1.46e-02
$\Theta + \eta I$: L^∞ error	6.43-01	3.13e-01	1.99e-01	1.47e-01	1.33e-01	1.43e-01
$\Theta + \eta R$: L^2 error	8.49e-02	9.29e-03	9.10e-03	3.34e-03	1.01e-03	3.36e-04
$\Theta + \eta R$: L^∞ error	4.02e-01	7.86e-02	5.58e-02	2.21e-02	7.17e-03	3.87e-03
η	10^{-7}	10^{-8}	10^{-9}	10^{-10}	10^{-11}	10^{-12}
$\Theta + \eta I$: L^2 error	1.37e-02	8.623-03	1.01e-02	1.92e-02	nan	nan
$\Theta + \eta I$: L^∞ error	1.81e-01	8.28e-02	1.07e-01	3.05e-01	nan	nan
$\Theta + \eta R$: L^2 error	1.55e-04	7.05e-05	4.56e-05	6.30e-06	1.73e-06	8.31e-07
$\Theta + \eta R$: L^∞ error	2.41e-03	1.07e-03	7.66e-04	8.92e-05	2.62e-05	1.17e-05

Table A.1: Comparison of solution errors between standard nugget terms and our adaptive nugget terms for the nonlinear elliptic PDE (4.1.1). Collocation points are uniformly sampled with $M = 1024$ and $M_\Omega = 900$ with a Gaussian kernel with lengthscale parameter $\sigma = 0.2$. Results are averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 5.

A.1.3 Choice of Nugget Terms for Burger's Equation

For the numerical experiments in Subsection 4.3.5.2 we primarily used our adaptive nugget term as outlined in Appendix A.1.1. For the results in Figure 4.3 we used a global nugget parameter $\eta = 10^{-10}$. For the convergence analysis in Table 4.2 we varied η for different number of collocation points to achieve better performance. More precisely, for $M \leq 1200$ we used a larger nugget $\eta = 10^{-5}$ and for $M \geq 2400$ we used $\eta = 10^{-10}$. Choosing a smaller nugget for small values of M would still yield equally accurate results but required more iterations of the Gauss-Newton algorithm.

A.1.4 Choice of Nugget Terms for the Eikonal Equation

Our numerical experiments in Subsection 4.3.5.3 also used the adaptive nugget of Appendix A.1.1. Indeed, we followed a similar approach to choosing the global

parameter η as in the case of Burger's equation outlined in Appendix A.1.3 above.

For the results in Figure 4.4 we used $\eta = 10^{-10}$. For the convergence analysis in Table 4.3 we varied η for different values of M, i.e., we chose $\eta = 10^{-5}$ for $M \leq 1200$ and $\eta = 10^{-10}$ for $M \geq 2400$. Analogously to the Burger's experiment we observed that smaller values of η for smaller choices of M cause the Gauss-Newton iterations to converge more slowly. Thus varying η with M improved the efficiency of our framework.

A.1.5 Choice of Nugget Terms for Darcy Flow

Both of the matrices $\Theta, \widetilde{\Theta}$ outlined in Subsection 4.4.1 are dense and ill-conditioned in the IP setting and so an appropriate nugget term should be chosen to regularize them. In the IP setting we propose to add adaptive nuggets for both $\Theta, \widetilde{\Theta}$ using the same strategy as in Appendix A.1.1, except that the nuggets are constructed independently for each matrix. To this end we set $\Theta \leftarrow \Theta + \eta R$ and $\widetilde{\Theta} \leftarrow \widetilde{\Theta} + \tilde{\eta}\tilde{R}$, where the $\tilde{\eta}, \tilde{R}$ denote the global nugget parameter and the re-weighted identity matrix corresponding to $\widetilde{\Theta}$. For the numerical experiments in Figure 4.5 we used $\eta = \tilde{\eta} = 10^{-5}$.

Appendix B

APPENDIX TO CHAPTER V

B.1 Supernodes and Aggregate Sparsity Pattern

The supernode idea is adopted from [240], which allows to *re-use* the Cholesky factors in the computation of (5.3.7) to update multiple columns at once.

We group the measurements into supernodes consisting of measurements whose points are *close in location and have similar lengthscale parameters* l_i. To do this, we select the last index $j \in I$ of the measurements in the ordering that has not been aggregated into a supernode yet and aggregate the indices in $\{i : (i,j) \in S_{P,l,\rho}, l_i \leq \lambda l_j\}$ that have not been aggregated yet into a common supernode, for some $\lambda > 1$. We repeat this procedure until every measurement has been aggregated into a supernode. We denote the set of all supernodes as $\tilde{I}$ and write $i \rightsquigarrow \tilde{i}$ for $i \in I$ and $\tilde{i} \in \tilde{I}$ if $\tilde{i}$ is the supernode to which i has been aggregated.

The idea is to assign the same sparsity pattern to all the measurements of the same supernode. To achieve so, we define the sparsity set for a supernode as the union of the sparsity sets of all the nodes it contains, namely $s_{\tilde{i}} := \{j : \exists i \rightsquigarrow \tilde{i}, j \in s_i\}$. Then, we introduce the aggregated sparsity pattern

$$S_{P,l,\rho,\lambda} := \bigcup_{\tilde{j}} \bigcup_{j \rightsquigarrow \tilde{j}} \{(i,j) \subset I \times I : i \leq j, i \in s_{\tilde{j}}\} .$$

Under mild assumptions (Theorem B.5 in [240]), one can show that there are $O(N/\rho^d)$ number of supernodes and each supernode contains $O(\rho^d)$ measurements. The size of the sparsity set for a supernode $\#s_{\tilde{j}} = O(\rho^d)$. For a visual demonstration of the grouping and aggregate sparsity pattern, see Figure B.1, which is taken from Figure 3 in [240].

Now, we can compute (5.3.7) with the aggregated sparsity pattern more efficiently. Let $s_j^* = \{i : (i,j) \in S_{P,l,\rho,\lambda}\}$ be the individual sparsity pattern for j in the aggregated pattern $S_{P,l,\rho,\lambda}$. In (5.3.7), we need to compute matrix-vector products for $\Theta^{-1}_{s_j^*,s_j^*}$. For that purpose, one can apply the Cholesky factorization to $\Theta_{s_j^*,s_j^*}$. Naïvely computing Cholesky factorizations of every $\Theta_{s_j^*,s_j^*}$ will result in $O(N\rho^{3d})$ arithmetic complexity. However, due to the supernode construction, we can factorize $\Theta_{s_{\tilde{j}},s_{\tilde{j}}}$ once and then use the resulting factor to obtain the Cholesky factors of $\Theta_{s_j^*,s_j^*}$ directly for all $j \rightsquigarrow \tilde{j}$.

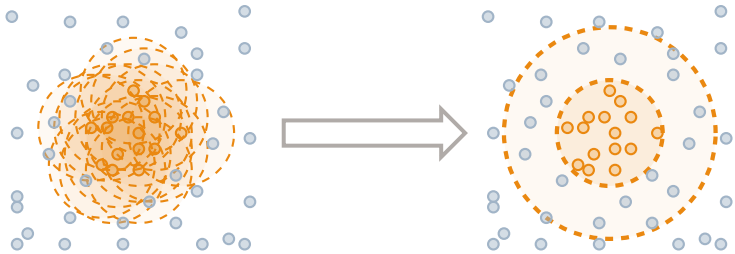


Figure B.1: The figure on the left illustrates the original pattern $S_{P,\ell,\rho}$. For each orange point j, its sparsity pattern s_j includes all points within a circle with a radius of ρ. On the right, all points j that are located close to each other and have similar lengthscales are grouped into a supernode $\tilde{j}$. The supernode can be represented by a list of *parents* (the orange points within an inner sphere of radius $\approx \rho$, or all $j \rightsquigarrow \tilde{j}$) and *children* (all points within a radius $\leq 2\rho$, which correspond to the sparsity set $s_{\tilde{j}}$). Figure reproduced from [240] with author permission.

This is because our construction guarantees that

$$\Theta_{s_j^*,s_j^*} = \Theta_{s_{\tilde{j}},s_{\tilde{j}}}[1 : \#s_j^*, 1 : \#s_j^*],$$

where we used the MATLAB notation. The above relation shows that sub-Cholesky factors of $\Theta_{s_{\tilde{j}},s_{\tilde{j}}}$ become the Cholesky factors of $\Theta_{s_j^*,s_j^*}$ for $j \rightsquigarrow \tilde{j}$.

Therefore, one step of Cholesky factorization works for all $O(\rho^d)$ measurements in the supernode. In total, the arithmetic complexity is upper bounded by $O(\rho^{3d} \times N/\rho^d) = O(N\rho^{2d})$. For more details of the algorithm, we refer to section 3.2, in particular Algorithm 3.2, in [240].

It was shown that the aggregate sparsity pattern could be constructed with time complexity $O(N\log(N) + N\rho^d)$ and space complexity $O(N)$; see Theorem C.3 in [240]. They are of a lower order compared to the time complexity $O(N\log^2(N)\rho^d)$ and space complexity $O(N\rho^d)$ for generating the maximin ordering and the original sparsity pattern $S_{P,l,\rho}$ (see Remark 5.3.4).

B.2 Ball-packing Arguments

The ball-packing argument is useful to bound the cardinality of the sparsity pattern.

Proposition B.2.1. *Consider the maximin ordering (Definition 5.3.2) and the sparsity pattern defined in* (5.3.4)*. For each column j, denote $s_j = \{i : (i,j) \in S_{P,l,\rho}\}$. The cardinality of the set s_j is denoted by $\#s_j$. Then, it holds that $\#s_j = O(\rho^d)$.*

Proof. Fix a j. For any $i \in s_j$, we have $\text{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)}) \leq \rho l_j$. Moreover, by the definition of the maximin ordering, we know that for $i, i' \in s_j$ and $i \neq i'$, it holds that $\text{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(i')}) \geq l_j$. Thus, the cardinality of s_i is bounded by the number of disjoint balls of radius l_j that the ball $B(\mathbf{x}_{P(i)}, 2\rho l_j)$ can contain. Clearly, $\#s_i = O(\rho^d)$. The proof is complete. □

B.3 Explicit Formula for the KL Minimization

By direct calculation, one can show the KL minimization attains an explicit formula. The proof of this explicit formula follows a similar approach to that of Theorem 2.1 in [240], with the only difference being the use of upper Cholesky factors.

Proposition B.3.1. *The solution to* (5.3.6) *is given by* (5.3.7).

Proof. We use the explicit formula for the KL divergence between two multivariate Gaussians:

$$\begin{aligned} &\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(UU^T)^{-1})\right) \\ =&\frac{1}{2}[-\log\det\Big(U^T\Theta U\Big) + \text{tr}(U^T\Theta U) - N]\,. \end{aligned} \tag{B.3.1}$$

To identify the minimizer, the constant and scaling do not matter, so we focus on the $-\log\det\big(U^T\Theta U\big)$ and $\text{tr}(U^T\Theta U)$ parts. By writing $U = [U_{:,1}, ..., U_{:,M}]$, we get

$$-\log\det\Big(U^T\Theta U\Big)+\text{tr}(U^T\Theta U) = \sum_{j=1}^{M}[-2\log U_{jj}+\text{tr}(U_{:,j}^T\Theta U_{:,j})]-\log\det\Theta\,. \tag{B.3.2}$$

Thus the minimization is decoupled to each column of U. Due to the choice of the sparse pattern, we have $U_{:,j}^T\Theta U_{:,j} = U_{s_j,j}^T\Theta_{s_j,s_j}U_{s_j,j}$. We can further simplify the formula

$$\text{tr}(U_{:,j}^T\Theta U_{:,j}) = \text{tr}(U_{s_j,j}^T\Theta_{s_j,s_j}U_{s_j,j})\,. \tag{B.3.3}$$

It suffices to identify the minimizer of $-2\log U_{jj} + \text{tr}(U_{s_j,j}^T\Theta_{s_j,s_j}U_{s_j,j})$. Taking the derivatives, we get the optimality condition:

$$-\frac{2}{U_{jj}}\mathbf{e}_{\#s_j} + 2\Theta_{s_j,s_j}U_{s_j,j} = 0\,, \tag{B.3.4}$$

where $\mathbf{e}_{\#s_j}$ is a standard basis vector in $\mathbb{R}^{\#s_j}$ with the last entry being 1 and other entries equal 0.

Solving this equation leads to the solution

$$U_{s_j,j} = \frac{\Theta_{s_j,s_j}^{-1}\mathbf{e}_{\#s_j}}{\sqrt{\mathbf{e}_{\#s_j}^T\Theta_{s_j,s_j}^{-1}\mathbf{e}_{\#s_j}}}\,. \tag{B.3.5}$$

The proof is complete. □

B.4 Proofs of the Main Theoretical Results

B.4.1 Proof of Theorem 5.4.2

Proof of Theorem 5.4.2. We rely on the interplay between GP regression, linear algebra, and numerical homogenization to prove this theorem. Consider the GP $\xi \sim \mathcal{N}(0, \mathcal{L}^{-1})$. For each measurement functional, we define the Gaussian random variables $Y_i = [\xi, \tilde{\phi}_i] \sim \mathcal{N}(0, [\tilde{\phi}_i, \mathcal{L}^{-1}\tilde{\phi}_i]) = \mathcal{N}(0, \Theta_{ii})$. As mentioned in Remark 5.3.3 and proved in Proposition B.4.4, we have a relation between the Cholesky factor and the GP conditioning, as

$$\frac{U^\star_{ij}}{U^\star_{jj}} = (-1)^{i\neq j} \frac{\mathrm{Cov}[Y_i, Y_j | Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i | Y_{1:j-1\setminus\{i\}}]}, i \leq j \,.$$

□

Moreover, by Proposition B.4.5, one can connect the conditional covariance of GPs with conditional expectation, such that

$$\frac{\mathrm{Cov}[Y_i, Y_j | Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i | Y_{1:j-1\setminus\{i\}}]} = \mathbb{E}[Y_j | Y_i = 1, Y_{1:j-1\setminus\{i\}} = 0] \,. \tag{B.4.1}$$

The above conditional expectation is related to the *Gamblets* introduced in the numerical homogenization literature. Indeed, using the relation $Y_i = [\xi, \tilde{\phi}_i]$, we have

$$\mathbb{E}[Y_j | Y_i = 1, Y_{1:j-1\setminus\{i\}} = 0] = [\mathbb{E}[\xi | Y_i = 1, Y_{1:j-1\setminus\{i\}} = 0], \tilde{\phi}_j] = [\psi^i_j, \tilde{\phi}_j] \,, \tag{B.4.2}$$

where $\psi^i_j(\mathbf{x}) := \mathbb{E}[\xi(\mathbf{x}) | Y_i = 1, Y_{1:j-1\setminus\{i\}} = 0]$; it is named Gamblets in [203, 204].

Importantly, for the conditional expectation, by Proposition B.4.6, we have the following variational characterization [203, 204]:

$$\begin{aligned} \psi^i_j = \mathrm{argmin}_{\psi \in H^s_0(\Omega)} \quad & [\psi, \mathcal{L}\psi] \\ \text{subject to} \quad & [\psi, \tilde{\phi}_k] = \delta_{i,k} \ \text{ for } \ 1 \leq k \leq j-1 \,. \end{aligned} \tag{B.4.3}$$

A main property of the Gamblets is that they can exhibit an exponential decay property under suitable assumptions, which can be used to prove our theorem. We collect the related theoretical results in Appendix B.4.4; we will use them in our proof.

Our proof for the theorem consists of two steps. The first step is to bound $|U^\star_{ij}/U^\star_{jj}|$, and the second step is to bound $|U^*_{jj}|$.

For the first step, we separate the cases $j \leq M$ and $j > M$. Indeed, the case $j \leq M$ has been covered in [241, 240]. Here, our proof is simplified.

For $1 \leq i \leq j \leq M$, by the discussions above, we have the relation:

$$\left|\frac{U^{\star}_{ij}}{U^{\star}_{jj}}\right| = \left|[\psi^i_j, \tilde{\phi}_j]\right| = |\psi^i_j(\mathbf{x}_{P(j)})|\,, \tag{B.4.4}$$

where we used the fact that $\tilde{\phi}_j = \delta_{\mathbf{x}_{P(j)}}$ because all the Diracs measurements are ordered first. To bound $|\psi^i_j(\mathbf{x}_{P(j)})|$, we will use the exponential decay results for Gamblets that we prove in Proposition B.4.13. More precisely, to apply Proposition B.4.13 to this context, we need to verify its assumptions, especially Assumption B.4.7.

In our setting, we can construct a partition of the domain Ω by using the Voronoi diagram. Denote $X_{j-1} = \{\mathbf{x}_{P(1)}, ..., \mathbf{x}_{P(j-1)}\}$. We note that X_M will consist of all the physical points. We define τ_k, for $1 \leq k \leq j-1$, to be the Voronoi cell, which contains all points in Ω that is closer to $\mathbf{x}_{P(k)}$ than to any other in X_{j-1}. Since we assume Ω is convex, τ_k is also convex. And bounded convex domains are uniform Lipschitz.

Furthermore, to verify the other parts in Assumption B.4.7, we analyze the homogeneity parameter of X_{j-1}. By definition,

$$\delta(X_{j-1}; \partial\Omega) = \frac{\min_{\mathbf{x},\mathbf{y}\in X_{j-1}} \operatorname{dist}(\mathbf{x}, \{\mathbf{y}\} \cup \partial\Omega)}{\max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, X_{j-1} \cup \partial\Omega)}\,. \tag{B.4.5}$$

By definition of the maximin ordering, we have

$$\min_{\mathbf{x},\mathbf{y}\in X_{j-1}} \operatorname{dist}(\mathbf{x}, \{\mathbf{y}\} \cup \partial\Omega) = l_{j-1}\,. \tag{B.4.6}$$

Then, by the triangle inequality, it holds that

$$\begin{aligned}\max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, X_{j-1} \cup \partial\Omega) &\leq \max_{\mathbf{x}\in X_M} \operatorname{dist}(\mathbf{x}, X_{j-1} \cup \partial\Omega) + \max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, X_M) \\ &\leq l_j + l_M/\delta(X_M; \partial\Omega) \leq (1 + 1/\delta(X_M; \partial\Omega))l_j\,,\end{aligned} \tag{B.4.7}$$

where in the second inequality, we used the definition of the lengthscales l_j and the homogeneity assumption of X_M (i.e., $\delta(X_M; \partial\Omega) > 0$). Combining the above two estimates, we get $\delta(X_{j-1}; \partial\Omega) \geq 1/(1 + 1/\delta(X_M; \partial\Omega)) > 0$ where we used the fact that $l_j \leq l_{j-1}$. So X_{j-1} is also homogeneously distributed with $\delta(X_{j-1}; \partial\Omega) > 0$.

We are ready to verify Assumption B.4.7. Firstly, the balls $B(\mathbf{x}, l_j/2), \mathbf{x} \in X_{j-1}$ do not intersect, and thus inside each τ_k, there is a ball of center $\mathbf{x}_{P(k)}$ and radius $l_j/2$. Secondly, since $\max_{\mathbf{x}\in\Omega} \operatorname{dist}(\mathbf{x}, X_{j-1} \cup \partial\Omega) \leq (1 + 1/\delta(X_M; \partial\Omega))l_j$, we know that τ_i is contained in a ball of center $\mathbf{x}_{P(k)}$ and radius $(1 + 1/\delta(X_M; \partial\Omega))l_j$. Therefore,

Assumption B.4.7 holds with $h = l_j/2$, $\delta = \min(1/(2 + 2/\delta(X_M; \partial\Omega)), 1)$ and $Q = j - 1$. Assumption B.4.8 readily holds by our choice of measurements.

Thus, applying Proposition B.4.13, we then get

$$\left|\frac{U^\star_{ij}}{U^\star_{jj}}\right| = |\psi^i_j(\mathbf{x}_{P(j)})| \leq C l_j^{-2s} \exp\left(-\frac{\operatorname{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C l_j}\right), \tag{B.4.8}$$

where C is a constant depending on $\Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$. We obtain the exponential decay of $|U^\star_{ij}/U^\star_{jj}|$ for $j \leq M$.

For $j > M$ and $i \leq j$, we have

$$\left|\frac{U^\star_{ij}}{U^\star_{jj}}\right| = \left|[\psi^i_j, \tilde{\phi}_j]\right| \leq \max_{0\leq|\gamma|\leq J} \left|D^J \psi^i_j(\mathbf{x}_{P(j)})\right|, \tag{B.4.9}$$

where we used the fact that $\tilde{\phi}_j$ is of the form $\delta_{\mathbf{x}_{P(j)}} \circ D^\gamma$. Now, for ψ^i_j, the set of points we need to deal with is X_M. Similar to the case $j \leq M$, we define τ_k, for $1 \leq k \leq M$, to be the Voronoi cell of these points in Ω. Then, using the same arguments in the previous case, we know that Assumption B.4.7 holds with $h = l_M/2$, $\delta = \min(\delta(X_M; \partial\Omega)/2, 1)$ and $Q = M$. Assumption B.4.8 readily holds by our choice of measurements. Therefore Proposition B.4.13 implies that

$$\max_{0\leq|\gamma|\leq J} \left|D^J \psi^i_j(\mathbf{x}_{P(j)})\right| \leq C l_M^{-2s} \exp\left(-\frac{\operatorname{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C l_M}\right), \tag{B.4.10}$$

where C is a constant depending on $\Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|, J$.

Summarizing the above arguments, we have obtained that

$$\left|\frac{U^\star_{ij}}{U^\star_{jj}}\right| \leq C l_j^{-2s} \exp\left(-\frac{\operatorname{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C l_j}\right), \tag{B.4.11}$$

for any $1 \leq i \leq j \leq N$, noting that by our definition $l_j = l_M$ for $j > M$.

Now, we analyze $|U^*_{jj}|$. Note that $\Theta = K(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\phi}}) \in \mathbb{R}^{N\times N}$ and $\Theta^{-1} = U^\star U^{\star T}$. By the arguments above, we know that the assumptions in Proposition B.5.1 are satisfied with $h = l_M/2$, $\delta = \min(\delta(X_M; \partial\Omega)/2, 1)$ and $Q = M$. Thus, for any vector $w \in \mathbb{R}^N$, by Proposition B.5.1, we have

$$w^T \Theta^{-1} w \leq C l_M^{-2s+d} |w|^2,$$

for some constant C depending on $\delta(X_M), d, s, \|\mathcal{L}\|, J$. This implies that

$$|U^{\star T} w|^2 \leq C l_M^{-2s+d} |w|^2.$$

Taking w to be the standard basis vector $\mathbf{e}_j$, we get

$$\sum_{k=j}^{N} |U^{\star}_{jk}|^2 \leq C l_M^{-2s+d},$$

which leads to $|U^{\star}_{ij}| \leq C l_M^{-s+d/2}$ for some C depending on $\delta(X_M; \partial\Omega), d, s, \|\mathcal{L}\|, J$.

B.4.2 Proof of Theorem 5.4.3

We need the following lemma, which is taken from Lemma B.8 in [240].

Lemma B.4.1. *Let $\lambda_{\min}, \lambda_{\max}$ be the minimal and maximal eigenvalues of $\Theta \in \mathbb{R}^{N\times N}$, respectively. Then there exists a universal constant $\eta > 0$ such that for any matrix $M \in \mathbb{R}^{N\times N}$, we have*

- *If $\lambda_{\max}\|\Theta^{-1} - MM^T\|_{\text{Fro}} \leq \eta$, then*

$$\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(MM^T)^{-1})\right) \leq \lambda_{\max}\|\Theta^{-1} - MM^T\|_{\text{Fro}};$$

- *If $\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(MM^T)^{-1})\right) \leq \eta$, then*

$$\|\Theta^{-1} - MM^T\|_{\text{Fro}} \leq \lambda_{\min}^{-1} \text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(MM^T)^{-1})\right).$$

With this lemma, we can prove Theorem 5.4.3. The proof is similar to that of Theorem B.6 in [240].

Proof of Theorem 5.4.3. First, by a covering argument, we know $l_M = O(M^{-1/d}) = O(N^{-1/d})$. By Proposition B.5.1 with $h = l_M/2$, we know that

$$\lambda_{\max}(\Theta) \leq C_1 N \quad \text{and} \quad \lambda_{\min}(\Theta) \geq C_1 N^{-2s/d+1} \tag{B.4.12}$$

for some constant C_1 depending on $\delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega), d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|, J$.

Theorem 5.4.2 implies that for $(i,j) \notin S_{P,l,\rho}$, it holds that

$$|U^{\star}_{ij}| \leq C l_M^{-s+d/2} l_j^{-2s} \exp\left(-\frac{\text{dist}(\mathbf{x}_{P(i)}, \mathbf{x}_{P(j)})}{C l_j}\right) \leq C N^{\alpha} \exp\left(-\frac{\rho}{C}\right),$$

for some α depending on s, d, where C is a generic constant that depends on Ω, $\delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega), d, s, J, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$. Moreover, from the proof in the last subsection, we know that $|U^{*}_{ij}| \leq C N^{s/d-1/2}$ for all $1 \leq i \leq j \leq N$.

Now, consider the upper triangular Cholesky factorization $\Theta^{-1} = U^\star U^{\star T}$. Define $M^\rho \in \mathbb{R}^{N\times N}$ such that

$$M^\rho_{ij} = \begin{cases} U^\star_{ij}, & \text{if } (i,j) \in S_{P,l,\rho} \\ 0, & \text{otherwise}\,. \end{cases} \tag{B.4.13}$$

Then, by using the above bounds on $U^\star_{ij}$, we know that there exists a constant β depending on s, d, such that

$$\|\Theta^{-1} - M^\rho M^{\rho T}\|_{\text{Fro}} \le CN^\beta \exp\left(-\frac{\rho}{C}\right)\,. \tag{B.4.14}$$

Since $\lambda_{\max}(\Theta) \le C_1 N$, we know that there exists a constant C', such that when $\rho \ge C' \log(N)$,

$$\lambda_{\max}(\Theta)\|\Theta^{-1} - M^\rho M^{\rho T}\|_{\text{Fro}} \le \eta\,,$$

for the η defined in Lemma B.4.1. Using Lemma B.4.1, we get

$$\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(M^\rho M^{\rho T})^{-1})\right) \le \lambda_{\max}(\Theta)\|\Theta^{-1} - M^\rho M^{\rho T}\|_{\text{Fro}}\,.$$

By the KL optimality, the optimal solution U^ρ will satisfy

$$\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(U^\rho U^{\rho T})^{-1})\right) \le \lambda_{\max}(\Theta)\|\Theta^{-1} - M^\rho M^{\rho T}\|_{\text{Fro}}\,. \tag{B.4.15}$$

Again by Lemma B.4.1, we get

$$\|\Theta^{-1} - U^\rho U^{\rho T}\|_{\text{Fro}} \le \frac{\lambda_{\max}(\Theta)}{\lambda_{\min}(\Theta)}\|\Theta^{-1} - M^\rho M^{\rho T}\|_{\text{Fro}}\,. \tag{B.4.16}$$

Moreover,

$$\|\Theta - (U^\rho U^{\rho T})^{-1}\|_{\text{Fro}} \le \|\Theta^{-1}\|_{\text{Fro}}\|\Theta^{-1} - U^\rho U^{\rho T}\|_{\text{Fro}}\|U^\rho U^{\rho T}\|_{\text{Fro}}\,. \tag{B.4.17}$$

Using (B.4.14), (B.4.15), (B.4.16), and (B.4.17), and the fact that $\lambda_{\max}(\Theta), \lambda_{\min}(\Theta)$ are of polynomial growth in N according to (B.4.12), we know that there exists a constant C depending on $\Omega, \delta(\{\mathbf{x}_i\}_{i\in I}; \partial\Omega), d, s, J, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$, such that when $\rho \ge C\log(N/\epsilon)$, it holds that

$$\text{KL}\left(\mathcal{N}(0,\Theta) \parallel \mathcal{N}(0,(U^\rho U^{\rho T})^{-1})\right) + \|\Theta^{-1} - U^\rho U^{\rho T}\|_{\text{Fro}} + \|\Theta - (U^\rho U^{\rho T})^{-1}\|_{\text{Fro}} \le \epsilon\,.$$

The proof is complete. □

B.4.3 Connections between Cholesky factors, conditional covariance, conditional expectation, and Gamblets

We first present two lemmas.

The first lemma is about the inverse of a block matrix [173].

Lemma B.4.2. *For a positive definite matrix $\Theta \in \mathbb{R}^{N\times N}$, if we write it in the block form*

$$\Theta = \begin{pmatrix} \Theta_{YY} & \Theta_{ZY} \\ \Theta_{YZ} & \Theta_{ZZ} \end{pmatrix}, \tag{B.4.18}$$

then, Θ^{-1} equals

$$\begin{pmatrix} (\Theta_{YY} - \Theta_{YZ}\Theta_{ZZ}^{-1}\Theta_{ZY})^{-1} & -\Theta_{YY}^{-1}\Theta_{YZ}(\Theta_{ZZ} - \Theta_{ZY}\Theta_{YY}^{-1}\Theta_{YZ})^{-1} \\ -\Theta_{ZZ}^{-1}\Theta_{ZY}(\Theta_{YY} - \Theta_{YZ}\Theta_{ZZ}^{-1}\Theta_{ZY})^{-1} & (\Theta_{ZZ} - \Theta_{ZY}\Theta_{YY}^{-1}\Theta_{YZ})^{-1} \end{pmatrix}. \tag{B.4.19}$$

The second lemma is about the relation between the conditional covariance of a Gaussian random vector and the precision matrix.

Lemma B.4.3. *Let $N \geq 3$ and $X = (X_1, ..., X_N)$ be a $\mathcal{N}(0, \Theta)$ Gaussian vector in $\mathbb{R}^N$. Let $Y = (X_1, X_2)$ and $Z = (X_3, ..., X_N)$. Then,*

$$\mathrm{Cov}(Y_1, Y_2|Z) = \frac{-A_{12}}{A_{11}A_{22} - A_{12}^2}, \tag{B.4.20}$$

and

$$\mathrm{Var}(Y_1|Z) = \frac{A_{22}}{A_{11}A_{22} - A_{12}^2}, \tag{B.4.21}$$

where $A = \Theta^{-1}$. Therefore

$$\frac{\mathrm{Cov}(Y_1, Y_2|Z)}{\mathrm{Var}(Y_1|Z)} = \frac{-A_{12}}{A_{22}}. \tag{B.4.22}$$

Proof. First, we know that $\mathrm{Cov}(Y|Z) = \Theta_{YY} - \Theta_{YZ}\Theta_{ZZ}^{-1}\Theta_{ZY}$. By Lemma B.4.2, we have

$$\Theta_{YY} - \Theta_{YZ}\Theta_{ZZ}^{-1}\Theta_{ZY} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}^{-1}. \tag{B.4.23}$$

Inverting this matrix leads to the desired result. □

With these lemmas, we can show the connection between the Cholesky factors and the conditional covariance as follows.

Proposition B.4.4. *Consider the positive definite matrix $\Theta \in \mathbb{R}^{N\times N}$. Suppose the upper triangular Cholesky factorization of its inverse is U, such that $\Theta^{-1} = UU^T$. Let Y be a Gaussian vector with law $\mathcal{N}(0,\Theta)$. Then, we have*

$$\frac{U_{ij}}{U_{jj}} = (-1)^{i\neq j}\frac{\mathrm{Cov}[Y_i, Y_j|Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i|Y_{1:j-1\setminus\{i\}}]}, \quad i \le j\,. \tag{B.4.24}$$

Proof. We only need to consider the case $i \neq j$. We consider $j = N$ first. Denote $A = \Theta^{-1}$. By the definition of the Cholesky factorization $A = UU^T$, we know that $U_{:,N} = A_{:,N}/\sqrt{A_{NN}}$. Thus,

$$\frac{U_{iN}}{U_{NN}} = \frac{A_{iN}}{A_{NN}}\,.$$

For $i \neq N$, by Lemma B.4.3, we have

$$\frac{\mathrm{Cov}[Y_i, Y_N|Y_{1:N-1\setminus\{i\}}]}{\mathrm{Var}[Y_i|Y_{1:N-1\setminus\{i\}}]} = \frac{-A_{iN}}{A_{NN}}\,. \tag{B.4.25}$$

By comparing the above two identities, we obtain the result holds for $j = N$.

For $j < N$, we can use mathematical induction. Note that $U_{1:N-1,1:N-1}$ is also the upper triangular Cholesky factor of $(\Theta_{1:N-1,1:N-1})^{-1}$, noting the fact (by applying Lemma B.4.2 to the matrix A) that

$$(\Theta_{1:N-1,1:N-1})^{-1} = A_{1:N-1,1:N-1} - A_{1:N-1,N}A_{NN}^{-1}A_{N,1:N-1}$$

is the Schur complement, which is exactly the residue block in the Cholesky factorization. Therefore, applying the result for $j = N$ to the matrix $\Theta_{1:N-1,1:N-1}$, we prove (B.4.24) holds for $j = N-1$ as well. Iterating this process to $j = 1$ finishes the proof. □

Furthermore, the conditional covariance is related to the conditional expectation, as follows.

Proposition B.4.5. *For a Gaussian vector $Y \sim \mathcal{N}(0,\Theta)$, we have*

$$\frac{\mathrm{Cov}[Y_i, Y_j|Y_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[Y_i|Y_{1:j-1\setminus\{i\}}]} = \mathbb{E}[Y_j|Y_i = 1, Y_{1:j-1\setminus\{i\}} = 0]\,. \tag{B.4.26}$$

Proof. We show that for any Gaussian vector Z in dimension d, and any vectors $v, w \in \mathbb{R}^d$ (such that $\mathrm{Var}[Z_w] \neq 0$), it holds that

$$\frac{\mathrm{Cov}[Z_v, Z_w]}{\mathrm{Var}[Z_w]} = \mathbb{E}[Z_v|Z_w = 1]\,, \tag{B.4.27}$$

where $Z_v = \langle Z, v\rangle$ and $Z_w = \langle Z, w\rangle$. Indeed this can be verified by direct calculations. We have

$$\mathrm{Cov}(Z_v - \frac{\mathrm{Cov}[Z_v, Z_w]}{\mathrm{Var}[Z_w]} Z_w, Z_w) = 0\,,$$

which implies they are independent since they are joint Gaussians. This yields

$$\mathbb{E}[Z_v - \frac{\mathrm{Cov}[Z_v, Z_w]}{\mathrm{Var}[Z_w]} Z_w | Z_w = 1] = 0\,,$$

which then implies (B.4.27) holds.

Now, we set Z to be the Gaussian vectors Y conditioned on $Y_{1:j-1\setminus\{i\}} = 0$. It is still a Gaussian vector (with a degenerate covariance matrix). Applying (B.4.27) with $v = \mathbf{e}_j, w = \mathbf{e}_i$, we get the desired result. □

Finally, the conditional expectation is connected a variational problem. The following proposition is taken from [203, 204]. One can understand the result as the maximum likelihood estimator for the GP conditioned on the constraint coincides with the conditional expectation since the distribution is Gaussian. Mathematically, it can be proved by writing down the explicit formula for the two problems directly.

Proposition B.4.6. *For a Gaussian process $\xi \sim \mathcal{N}(0, \mathcal{L}^{-1})$ where $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$ satisfies Assumption 5.4.1. Then, for some linearly independent measurements $\phi_1, ..., \phi_l \in H^{-s}(\Omega)$, the conditional expectation*

$$\psi^\star(\mathbf{x}) := \mathbb{E}\left[\xi(\mathbf{x}) | [\xi, \phi_i] = c_i, 1 \le i \le l\right]$$

is the solution to the following variational problem:

$$\begin{aligned} \psi^\star = \mathrm{argmin}_{\psi \in H_0^s(\Omega)} \quad & [\psi, \mathcal{L}\psi] \\ \text{subject to} \quad & [\psi, \phi_i] = c_i \ \text{ for } \ 1 \le i \le l\,. \end{aligned} \tag{B.4.28}$$

The solution of the above variational problem is termed Gamblets in the literature; see Definition B.4.9.

B.4.4 Results regarding the exponential decay of Gamblets

We collect and organize some theoretical results of the exponential decay property of Gamblets from [204]. And we provide some new results concerning the derivative measurements which are not covered in [204].

The first assumption is about the domain and the partition of the domain.

Assumption B.4.7 (Domain and partition: from Construction 4.2 in [204])**.** *Suppose Ω is a bounded domain in $\mathbb{R}^d$ with a Lipschitz boundary. Consider $\delta \in (0,1)$ and $h > 0$. Let $\tau_1, ..., \tau_Q$ be a partition of $\Omega \subset \mathbb{R}^d$ such that the closure of each τ_i is convex, is uniformly Lipschitz, contains a ball of center $\mathbf{x}_i$ and radius δh, and is contained in the ball of center $\mathbf{x}_i$ and radius h/δ.*

The second assumption is regarding the measurement functionals related to the partition of the domain.

Assumption B.4.8 (Measurement functionals: from Construction 4.12 in [204])**.** *Let Assumption B.4.7 holds. For each $1 \leq i \leq Q$, let $\phi_{i,\alpha}, \alpha \in \mathsf{T}_i$ (where T_i is an index set) be elements of $H^{-s}(\Omega)$ that the following conditions hold:*

- *Linear independence: $\phi_{i,\alpha}, \alpha \in \mathsf{T}_i$ are linearly independent when acting on the subset $H_0^s(\tau_i) \subset H_0^s(\Omega)$.*
- *Locality: $[\phi_{i,\alpha}, \psi] = 0$ for every $\psi \in C_0^\infty(\Omega \backslash \tau_i)$ and $\alpha \in \mathsf{T}_i$.*

With the measurement functionals, we can define Gamblets as follows via a variational problem. Note that according to Proposition B.4.6, Gamblets are also conditional expectations of some GP given the measurement functionals.

Definition B.4.9 (Gamblets: from Section 4.5.2.1 in [204])**.** *Let Assumptions B.4.7 and B.4.8 hold, and the operator $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$ satisfies Assumption 5.4.1. The Gamblets $\psi_{i,\alpha}, 1 \leq i \leq Q, \alpha \in \mathsf{T}_i$ associated with the operator $\mathcal{L}$ and measurement functionals $\phi_{i,\alpha}, 1 \leq i \leq Q, \alpha \in \mathsf{T}_i$ are defined as*

$$\begin{aligned} \psi_{i,\alpha} = & \underset{\psi \in H_0^s(\Omega)}{\arg\min} \quad [\psi, \mathcal{L}\psi] \\ & \text{subject to} \quad [\psi, \phi_{k,\beta}] = \delta_{ik}\delta_{\alpha\beta} \ \text{ for } \ 1 \leq k \leq Q, \beta \in \mathsf{T}_k \, . \end{aligned} \tag{B.4.29}$$

A crucial property of Gamblets is that they exhibit exponential decay; see the following Theorem B.4.11.

Remark B.4.10. *Exponential decay results regarding the solution to optimization problems of the type* (B.4.29) *are first established in [181], where the measurement functionals are piecewise linear nodal functions in finite element methods and the operator $\mathcal{L} = -\nabla \cdot (a\nabla \cdot)$. Then, the work [203] extends the result to piecewise constant measurement functionals and uses it to develop a multigrid algorithm for*

elliptic PDEs with rough coefficients. Later on, the work [131] extends the analysis to a class of strongly elliptic high-order operators with piecewise polynomial-type measurement functionals, and the work [44, 45] focuses on detailed analysis regarding the subsampled lengthscale for subsampled measurement functionals. All these results rely on similar mass-chasing arguments, which are difficult to extend to general higher-order operators.

The paper [148] presents a simpler and more algebraic proof of the exponential decay in [181] based on the exponential convergence of subspace iteration methods. Then, the work [204] extends this technique (by presenting necessary and sufficient conditions) to general arbitrary integer order operators and measurement functionals. Specifically, the authors in [241] use the conditions in [204] to show the desired exponential decay when the operator $\mathcal{L}$ satisfies Assumption 5.4.1, and the measurement functionals are Diracs.

In Theorem B.4.11, we present the sufficient conditions in [204] that ensure the exponential decay and verify that the derivative-type measurements considered in this paper indeed satisfy these conditions; see Propositions B.4.12 and B.4.13. ◊

Theorem B.4.11 (Exponential decay of Gamblets)**.** *Let Assumptions B.4.7 and B.4.8 hold. We define the function space*

$$\Phi^{\perp} := \{f \in H_0^s(\Omega) : [f, \phi_{i,\alpha}] = 0 \text{ for any } \alpha \in \mathsf{T}_i, 1 \leq i \leq Q\}.$$

Assume, furthermore the following conditions hold:

$$|f|_{H^t(\Omega)} \leq C_0 h^{s-t} \|f\|_{H_0^s(\Omega)} \text{ for any } 0 \leq t \leq s \text{ and } f \in \Phi^{\perp}; \tag{B.4.30}$$

$$\sum_{1 \leq i \leq Q, \alpha \in \mathsf{T}_i} [f, \phi_{i,\alpha}]^2 \leq C_0 \sum_{t=0}^{s} h^{2t} |f|^2_{H^t(\Omega)} \text{ for any } f \in H_0^s(\Omega); \tag{B.4.31}$$

$$|y| \leq C_0 h^{-s} \| \sum_{\alpha \in \mathsf{T}_i} x_\alpha \phi_{i,\alpha} \|_{H^{-s}(\tau_i)} \text{ for any } 1 \leq i \leq Q \text{ and } y \in \mathbb{R}^{|\mathsf{T}_i|}. \tag{B.4.32}$$

Here, $|\cdot|_{H^t(\Omega)}$ is the Sobolev seminorm in Ω of order t.

Then, for the Gamblets in Definition B.4.9, we have

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \leq C h^{-s} \exp\left(-\frac{\operatorname{dist}(\mathbf{x}, \mathbf{x}_i)}{Ch}\right),$$

for any $1 \leq i \leq Q, \alpha \in \mathsf{T}_i$ and multi-index γ satisfying $|\gamma| < s - d/2$. Here C is a constant depending on $C_0, \Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$.

Proof. We use results in the book [204]. Conditions (B.4.30), (B.4.31), and (B.4.32) are equivalently Condition 4.15 in [204]. Let

$$\Omega_i = \operatorname{int}\left(\bigcup_{j:\operatorname{dist}(\tau_i,\tau_j)\le \delta h} \tau_j \right) \subset B(\mathbf{x}_i, 3h/\delta)\,. \tag{B.4.33}$$

Then, by Theorem 4.16 in [204], there exists a constant C that depends on C_0, δ, d, s, $\|\mathcal{L}\|$, $\|\mathcal{L}^{-1}\|$ such that

$$\|\psi_{i,\alpha} - \psi_{i,\alpha}^n\|_{H_0^s(\Omega)} \le C h^{-s} \exp(-n/C)\,, \tag{B.4.34}$$

where $\psi_{i,\alpha}^n$ is the minimizer of (B.4.29) after replacing the condition $\psi \in H_0^s(\Omega)$ by $\psi \in H_0^s(\Omega_i^n)$. Here, Ω_i^n is the collection of Ω_j whose graph distance to Ω_i is not larger than n; see the definition of graph distance in Definition 4.13 of [204]. Intuitively, one can understand Ω_i^n as the n-layer neighborhood of Ω_i. We have $\Omega_i^0 = \Omega_i$.

By the definition of Ω_i^n and Assumption B.4.7, we know that

$$B(\mathbf{x}_i, (n+1)\delta h) \cap \Omega \subset \Omega_i^n \subset B(\mathbf{x}_i, 9(n+1)h/\delta) \cap \Omega\,. \tag{B.4.35}$$

Now, using the Sobolev embedding theorem and the fact that $\psi_{i,\alpha}^n$ is supported in Ω_i^n, we get

$$\begin{aligned} \|D^\gamma \psi_{i,\alpha}\|_{L^\infty(\Omega\setminus B(\mathbf{x}_i, 9nh/\delta))} &\le \|D^\gamma \psi_{i,\alpha}\|_{L^\infty(\Omega\setminus\Omega_i^n)} \\ &\le \|D^\gamma \psi_{i,\alpha} - D^\gamma \psi_{i,\alpha}^n\|_{L^\infty(\Omega)} \\ &\le C_1 \|\psi_{i,\alpha} - \psi_{i,\alpha}^n\|_{H_0^s(\Omega)} \\ &\le C_1 C h^{-s} \exp(-n/C)\,, \end{aligned} \tag{B.4.36}$$

where C_1 is a constant depending on Ω, d, J that satisfies

$$\sup_{0\le|\gamma|\le J} \|D^\gamma u\|_{L^\infty(\Omega)} \le C_1 \|u\|_{H_0^s(\Omega)}\,,$$

for any $u \in H_0^s(\Omega)$.

The result (B.4.36) implies that for any $n \in \mathbb{N}$, once $\operatorname{dist}(\mathbf{x}, \mathbf{x}_i) \ge 9(n+1)h/\delta$, it holds that

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \le C h^{-s} \exp\Big(-\frac{n}{C}\Big)\,,$$

where C is some constant depending on $C_0, \Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|$. Here we used the fact that the function $C \to C h^{-s} \exp\big(-\frac{n}{C}\big)$ is increasing.

In particular, the above inequality holds when $9(n+1)h/\delta \leq \text{dist}(\mathbf{x}, \mathbf{x}_i) \leq 9(n+2)h/\delta$; the relation yields $n \sim \text{dist}(\mathbf{x}, \mathbf{x}_i)/h$. By replacing n in terms of $\text{dist}(\mathbf{x}, \mathbf{x}_i)/h$, we obtain that there exists a constant C depending on the same set of variables, such that

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \leq Ch^{-s} \exp\left(-\frac{\text{dist}(\mathbf{x}, \mathbf{x}_i)}{Ch}\right).$$

The proof is complete. □

In the following, we show that conditions (B.4.30), (B.4.31), and (B.4.32) are satisfied by the derivative measurements that we are focusing on in the paper. We need to scale each derivative measurement by a power of h; this will only change the resulting Gamblets by a corresponding scaling, which would not influence the exponential decay result; see Proposition B.4.13.

Proposition B.4.12. *Let Assumptions B.4.7 and B.4.8 hold. Consider $J \in \mathbb{N}$ and $J < s - d/2$. For each $1 \leq i \leq Q$, we choose $\{h^{d/2}\delta_{\mathbf{x}_i}\} \subset \{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} \subset \{h^{d/2+|\gamma|}\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \leq |\gamma| \leq J\}$. Then, Conditions* (B.4.30)*,* (B.4.31)*,* (B.4.32) *hold with the constant C_0 depending on δ, d, s and J only.*

Proof. We will verify the conditions one by one.

Condition (B.4.30) follows from Section 15.4.5 of the book [204], where the case of $\{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} = \{h^{d/2}\delta_{\mathbf{x}_i}\}$ is covered. More precisely, in our case, we have more measurement functionals compared to these Diracs, and thus the space $\Phi^\perp$ is smaller than that considered in Section 15.4.5 of the book [204]. This implies that their results directly apply and (B.4.30) readily holds in our case.

For Conditions (B.4.31) and (B.4.32), we can assume $\{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} = \{h^{d/2+|\gamma|}\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \leq |\gamma| \leq J\}$ since this could only make the constant C_0 larger. We start with Condition (B.4.31). We note that in our proof, C represents a generic constant and can vary from place to place; we will specify the variables it can depend on.

Consider the unit ball $B(0,1) \subset \mathbb{R}^d$. Since $J < s - d/2$, the Sobolev embedding theorem implies that

$$\sum_{0 \leq |\gamma| \leq J} \|D^\gamma f\|_{L^\infty(B(0,1))} \leq C\|f\|_{H^s(B(0,1))}, \tag{B.4.37}$$

for any $f \in H^s(B(0,1))$, where C is a constant depending on d and s. This implies that

$$\sum_{0 \leq |\gamma| \leq J} |D^\gamma f(0)|^2 \leq C \sum_{t=0}^{s} |f|^2_{H^t(B(0,1))}, \tag{B.4.38}$$

where C depends on d, s, and J. Consequently, using the change of variables $\mathbf{x} = \mathbf{x}_i + \delta h \mathbf{x}'$, we get

$$\sum_{0\le|\gamma|\le J} \delta^d h^{d+2|\gamma|} |D^\gamma f(\mathbf{x}_i)|^2 \le C \sum_{t=0}^{s} \delta^{2t} h^{2t} |f|^2_{H^t(B(\mathbf{x}_i,\delta h))}. \tag{B.4.39}$$

We can absorb δ into C and obtain

$$\sum_{0\le|\gamma|\le J} h^{d+2|\gamma|} |D^\gamma f(\mathbf{x}_i)|^2 \le C \sum_{t=0}^{s} h^{2t} |f|^2_{H^t(B(\mathbf{x}_i,\delta h))}, \tag{B.4.40}$$

where C depends on δ, d, s, and J, for any $f \in H^s(B(\mathbf{x}_i, \delta h))$. Now, using the fact that $H^s(B(\mathbf{x}_i, \delta h)) \subset H^s_0(\Omega)$ and $\{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} = \{h^{d/2-|\gamma|}\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \le |\gamma| \le J\}$, we arrive at

$$\sum_{\alpha\in\mathsf{T}_i} [f, \phi_{i,\alpha}]^2 \le C \sum_{t=0}^{s} h^{2t} |f|^2_{H^t(B(\mathbf{x}_i,\delta h))}, \tag{B.4.41}$$

for any $f \in H^s_0(\Omega)$. Summing the above inequalities for all $\mathbf{x}_i$, we get

$$\sum_{1\le i\le Q, \alpha\in\mathsf{T}_i} [f, \phi_{i,\alpha}]^2 \le C \sum_{t=0}^{s} h^{2t} |f|^2_{H^t(\Omega)} \text{ for any } f \in H^s_0(\Omega), \tag{B.4.42}$$

where C depends on δ, d, s and J. This verifies Condition (B.4.31).

For Condition (B.4.32), we have

$$\begin{aligned} \| \sum_{\alpha\in\mathsf{T}_i} y_\alpha \phi_{i,\alpha} \|_{H^{-s}(\tau_i)} &\ge \| \sum_{\alpha\in\mathsf{T}_i} y_\alpha \phi_{i,\alpha} \|_{H^{-s}(B(\mathbf{x}_i,\delta h))} \\ &= \| \sum_{0\le|\gamma|\le J} y_\gamma h^{d/2+|\gamma|} \delta_{\mathbf{x}_i} \circ D^\gamma \|_{H^{-s}(B(\mathbf{x}_i,\delta h))}, \end{aligned} \tag{B.4.43}$$

where we abuse the notations to write y_α by y_γ.

Now, by definition, we get

$$\begin{aligned} &\| \sum_{0\le|\gamma|\le J} y_\gamma h^{d/2+|\gamma|} \delta_{\mathbf{x}_i} \circ D^\gamma \|_{H^{-s}(B(\mathbf{x}_i,\delta h))} \\ &= \sup_{v\in H^s_0(B(\mathbf{x}_i,\delta h))} \frac{\sum_{0\le|\gamma|\le J} y_\gamma D^\gamma v(\mathbf{x}_i) h^{d/2+|\gamma|}}{\|v\|_{H^s_0(B(\mathbf{x}_i,\delta h))}} \\ &= \sup_{v\in H^s_0(B(0,1))} \frac{\sum_{0\le|\gamma|\le J} y_\gamma D^\gamma v(0) h^{d/2+|\gamma|} (\delta h)^{-|\gamma|}}{\|v\|_{H^s_0(B(0,1))} (\delta h)^{d/2-s}} \\ &\ge C h^{-s} \| \sum_{0\le|\gamma|\le J} y_\gamma \delta_0 \circ D^\gamma \|_{H^{-s}(B(0,1))}, \end{aligned} \tag{B.4.44}$$

where C is a constant that depends on δ, s. In the second identity, we used the change of coordinates $\mathbf{x} = \mathbf{x}_i + \delta h \mathbf{x}'$.

Now, since $\delta_0 \circ D^\gamma, 0 \leq |\gamma| \leq J$ are linearly independent, we know that there exists C' depending on d, J, such that

$$\| \sum_{0 \leq |\gamma| \leq J} y_\gamma \delta_0 \circ D^\gamma \|_{H^{-s}(B(0,1))} \geq C' C |y| \tag{B.4.45}$$

holds for any y. Let $C_0 = C'C$, then Condition (B.4.32) is verified. □

By Proposition B.4.12 and rescaling, we can obtain the following results for the unscaled measurements.

Proposition B.4.13. *Let Assumptions B.4.7 and B.4.8 hold. Consider $J \in \mathbb{N}$ and $J < s - d/2$. For each $1 \leq i \leq Q$, we choose $\{\delta_{\mathbf{x}_i}\} \subset \{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} \subset \{\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \leq |\gamma| \leq J\}$. Then, for the Gamblets in Definition B.4.9, we have*

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \leq C h^{-2s} \exp\left(-\frac{\mathrm{dist}(\mathbf{x}, \mathbf{x}_i)}{Ch}\right),$$

for any $1 \leq i \leq Q, \alpha \in \mathsf{T}_i$ and multi-index γ satisfying $|\gamma| < s - d/2$. Here C is a constant depending on $\Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|, J$.

Proof. Based on Theorem B.4.11 and Proposition B.4.12, we know that

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \leq C h^{-s} \exp\left(-\frac{\mathrm{dist}(\mathbf{x}, \mathbf{x}_i)}{Ch}\right)$$

holds if $\psi_{i,\alpha}$ is the Gamblet corresponding to the measurement functionals satisfying $\{h^{d/2}\delta_{\mathbf{x}_i}\} \subset \{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} \subset \{h^{d/2+|\gamma|}\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \leq |\gamma| \leq J\}$. Using the fact that $|\gamma| + d/2 \leq s$ and the definition of Gamblets, we know that when the measurement functionals change to $\{\delta_{\mathbf{x}_i}\} \subset \{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} \subset \{\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \leq |\gamma| \leq J\}$, the corresponding Gamblets will satisfy

$$|D^\gamma \psi_{i,\alpha}(\mathbf{x})| \leq C h^{-2s} \exp\left(-\frac{\mathrm{dist}(\mathbf{x}, \mathbf{x}_i)}{Ch}\right).$$

The proof is complete. □

B.5 Eigenvalue Bounds on the Kernel Matrices

This section is devoted to the lower and upper bounds of the eigenvalues of the kernel matrix.

Proposition B.5.1. *Let Assumptions B.4.7 and B.4.8 hold. Consider $J \in \mathbb{N}$ and $J < s - d/2$. For each $1 \le i \le Q$, we choose $\{\delta_{\mathbf{x}_i}\} \subset \{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} \subset \{\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \le |\gamma| \le J\}$. Let $\boldsymbol{\phi}$ be the collection of all $\phi_{i,\alpha}, 1 \le i \le Q, \alpha \in \mathsf{T}_i$. Let the operator $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$ satisfies Assumption 5.4.1 and assume the kernel function to be the Green function $K(\mathbf{x}, \mathbf{y}) := [\delta_{\mathbf{x}}, \mathcal{L}^{-1}\delta_{\mathbf{y}}]$. Then, for the kernel matrix $K(\boldsymbol{\phi}, \boldsymbol{\phi})$, we have*

$$C_{\max} h^{-d} \mathrm{I} \ge K(\boldsymbol{\phi}, \boldsymbol{\phi}) \ge C_{\min} h^{2s-d} \mathrm{I}, \tag{B.5.1}$$

where $C_{\min}$ is a constant that depends on $\delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|, J$, and $C_{\max}$ additionally depends on Ω. And I is the identity matrix.

Proof. It suffices to consider the case $\{\phi_{i,\alpha}, \alpha \in \mathsf{T}_i\} = \{\delta_{\mathbf{x}_i} \circ D^\gamma, 0 \le |\gamma| \le J\}$; the kernel matrix in other cases can be seen as a principal submatrix of the case considered here. The eigenvalues of the principal submatrix can be lower and upper bounded by the smallest and largest eigenvalues of the original matrix, respectively.

Suppose $K(\boldsymbol{\phi}, \boldsymbol{\phi}) \in \mathbb{R}^{N \times N}$. For any $y \in \mathbb{R}^N$, by definition, we have

$$y^T K(\boldsymbol{\phi}, \boldsymbol{\phi}) y = [\sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma, \mathcal{L}^{-1}(\sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma)]\,. \tag{B.5.2}$$

We first deal with the largest eigenvalue. By (B.5.2), we get

$$y^T K(\boldsymbol{\phi}, \boldsymbol{\phi}) y \le \|\mathcal{L}^{-1}\| \| \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|^2_{H^{-s}(\Omega)}\,.$$

Due to the assumption $J < s - d/2$, there exists a constant C_1 depending on Ω, d, J such that

$$\sup_{0 \le |\gamma| \le J} \|D^\gamma u\|_{L^\infty(\Omega)} \le C_1 \|u\|_{H_0^s(\Omega)}\,,$$

for any $u \in H_0^s(\Omega)$. This implies that for any $0 \le |\gamma| \le J$ and $\mathbf{x}_i \in \Omega$, it holds

$$\|\delta_{\mathbf{x}_i} \circ D^\gamma\|_{H^{-s}(\Omega)} = \sup_{u \in H_0^s(\Omega)} \frac{|D^\gamma u(\mathbf{x}_i)|}{\|u\|_{H_0^s(\Omega)}} \le C_1\,.$$

Therefore, we get

$$\begin{aligned} &\| \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|_{H^{-s}(\Omega)} \\ \le &\sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} |y_{i,\gamma}| \cdot \|\delta_{\mathbf{x}_i} \circ D^\gamma\|_{H^{-s}(\Omega)} \\ \le &C_1 \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} |y_{i,\gamma}| \\ \le &C_1 C_J \sum_{1 \le i \le Q} |y_i| \le C_1 C_J \sqrt{Q} |y|\,, \end{aligned} \tag{B.5.3}$$

where in the inequality, we used the triangle inequality. In the third and fourth inequalities, we used the Cauchy-Schwarz inequality, and C_J is a constant that depends on J. Here, we abuse the notation and write y_i to be the vector collecting $y_{i,\gamma}, 0 \le |\gamma| \le J$.

Now, by a covering argument, we know that $Q = O(h^{-d})$. Therefore, combining (B.5.3) and (B.5.2), we arrive at

$$y^T K(\boldsymbol{\phi}, \boldsymbol{\phi}) y \le \|\mathcal{L}^{-1}\| C_1^2 C_J^2 Q |y|^2 \le C_{\max} h^{-d} |y|^2 ,$$

where $C_{\max}$ is a constant that depends on $\Omega, \delta, d, s, \|\mathcal{L}\|, \|\mathcal{L}^{-1}\|, J$. We have obtained the upper bound for the largest eigenvalue of $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ as desired.

For the smallest eigenvalue, using (B.5.2) again, we have

$$y^T K(\boldsymbol{\phi}, \boldsymbol{\phi}) y \ge \|\mathcal{L}\|^{-1} \| \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|^2_{H^{-s}(\Omega)} .$$

Now, using the Fenchel duality, we get

$$\| \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|^2_{H^{-s}(\Omega)} = \sup_{v \in H_0^s(\Omega)} 2 \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} D^\gamma v(\mathbf{x}_i) - \|v\|^2_{H_0^s(\Omega)} . \tag{B.5.4}$$

By restricting $v \in H_0^s(\Omega)$ to $v = \sum_{1 \le i \le Q} v_i$ with each $v_i \in H_0^s(B(\mathbf{x}_i, \delta h))$, we obtain

$$\begin{aligned} &\sup_{v \in H_0^s(\Omega)} 2 \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} D^\gamma v(\mathbf{x}_i) - \|v\|^2_{H_0^s(\Omega)} \\ &\ge \sum_{1 \le i \le Q} \left(\sup_{v_i \in H_0^s(B(\mathbf{x}_i, \delta h))} 2 \sum_{0 \le |\gamma| \le J} y_{i,\gamma} D^\gamma v_i(\mathbf{x}_i) - \|v_i\|_{H_0^s(B(\mathbf{x}_i, \delta h))} \right) \\ &= \sum_{1 \le i \le Q} \| \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|^2_{H^{-s}(B(\mathbf{x}_i, \delta h))} , \end{aligned} \tag{B.5.5}$$

where in the first inequality, we used the fact that the balls $B(\mathbf{x}_i, \delta h), 1 \le i \le Q$ are disjoint.

By (B.4.44) and (B.4.45), we know that

$$\| \sum_{0 \le |\gamma| \le J} y_{i,\gamma} h^{d/2 + |\gamma|} \delta_{\mathbf{x}_i} \circ D^\gamma \|_{H^{-s}(B(\mathbf{x}_i, \delta h))} \ge C h^s |y_i| , \tag{B.5.6}$$

for some constant C depending on δ, d, s, J. Here again, we write y_i to be the vector collecting $y_{i,\gamma}, 0 \le |\gamma| \le J$. By change of variables, the above inequality implies that

$$\| \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|_{H^{-s}(B(\mathbf{x}_i, \delta h))} \ge C h^{s - d/2} |y_i| , \tag{B.5.7}$$

for some constant C depending on δ, d, s, J.

Combining (B.5.4), (B.5.5), (B.5.7), we obtain

$$\| \sum_{1 \le i \le Q} \sum_{0 \le |\gamma| \le J} y_{i,\gamma} \delta_{\mathbf{x}_i} \circ D^\gamma \|^2_{H^{-s}(\Omega)} \ge C h^{2s-d} |y|^2 . \tag{B.5.8}$$

With (B.5.2), we obtain

$$y^T K(\boldsymbol{\phi}, \boldsymbol{\phi}) y \ge C \|\mathcal{L}\|^{-1} h^{2s-d} |y|^2 . \tag{B.5.9}$$

The proof is complete. □

Appendix C

APPENDIX TO CHAPTER VI

C.1 Appendix: Proofs

C.1.1 Proof of Proposition 6.2.11

Proof. Let $\varphi_j(x) = (-\Delta)^{-t}\delta(x - x_j)$ and in particular $\varphi_0(x) = (-\Delta)^{-t}\delta(x)$. We have for $m \in \mathbb{Z}^d$,

$$\hat{\varphi}_0(m) = \begin{cases} (4\pi^2)^{-t}|m|^{-2t}, & m \neq 0 \\ 0, & m = 0\,. \end{cases}$$

We introduce the translation operator $\tau_{j2^{-q}}$ which acts on function $u : \mathbb{T}^d \to \mathbb{R}$ and is defined by

$$(\tau_{j2^{-q}}u)(x) = u(x_1 - j_1 2^{-q}, x_2 - j_2 2^{-q}, ..., x_d - j_d 2^{-q})$$

for $j = (j_1, j_2, ..., j_d) \in \mathbb{Z}^d$ and $x = (x_1, x_2, ..., x_d) \in \mathbb{R}^d$. Then, for $j \in J_q$, we have the relation $\delta(\cdot - x_j) = \tau_{j2^{-q}}\delta(\cdot)$. Using the property of the Fourier coefficients, we obtain

$$\hat{\varphi}_j(m) = \hat{\varphi}_0(m)e^{-2\pi i\langle j2^{-q},m\rangle} = \begin{cases} (4\pi^2)^{-t}|m|^{-2t}e^{-2\pi i\langle j2^{-q},m\rangle}, & m \neq 0 \\ 0, & m = 0\,. \end{cases}$$

By definition, $\hat{\mathcal{F}}_{t,q}$ is the span of such $\hat{\varphi}_j$ for $j \in J_q$. Hence, for any $g \in \hat{\mathcal{F}}_{t,q}$, it can be written as a linear combination of these functions. Equivalently, there exists a 2^q-periodic function p such that

$$g(m) = \begin{cases} |m|^{-2t}p(m), & m \neq 0 \\ 0, & m = 0\,. \end{cases}$$

This gives the desired representation of g. □

C.1.2 Proof of Theorem 6.2.13

Proof. By Proposition 6.2.11, there exists a 2^q-periodic function $p_1(m)$ on $\mathbb{Z}^d$, such that

$$\hat{u}(m,t,q) = \begin{cases} |m|^{-2t}p_1(m), & m \neq 0 \\ 0, & m = 0\,. \end{cases}$$

By the definition of GPR, we have $[u^\dagger(\cdot) - u(\cdot, t, q), \delta(\cdot - x_j)] = 0$ for every data point x_j. In the Fourier domain, according to the characterization of $\hat{\mathcal{F}}_{t,q}$, this orthogonality leads to

$$\sum_{m\in\mathbb{Z}^d} (\hat{u}(m) - \hat{u}(m,t,q))p(m) = 0 \tag{C.1.1}$$

for $p : \mathbb{Z}^d \to \mathbb{C}$ being any 2^q-periodic function. Recalling Definition 6.2.12, we have

$$(T_q\hat{u})(m) = \sum_{\beta\in\mathbb{Z}^d} \hat{u}(m + 2^q\beta)\,. \tag{C.1.2}$$

The fact that the above sum converges may be seen as a consequence of the Cauchy–Schwarz inequality and the regularity of u (recall $t \geq d/2 + \delta$). Using (C.1.2) and the representation of $\hat{u}(m,t,q)$, we reformulate (C.1.1) as

$$\sum_{m\in B_q^d} \Big((T_q\hat{u})(m) - M_q^t(m)p_1(m)\Big)\, p(m) = 0\,.$$

The above formula holds for any 2^q-periodic function p. Let $g(m) = (T_q\hat{u})(m) - M_q^t(m)p_1(m)$, then we get that g is a 2^q-periodic function on $\mathbb{Z}^d$ and that

$$\sum_{m\in B_q^d} g(m)p(m) = 0$$

holds for any 2^q-periodic function p. This implies that $g(m) = 0$. Hence, we get

$$p_1(m) = \frac{(T_q\hat{u})(m)}{M_q^t(m)}\,.$$

Plugging this expression into the above representation formula for $\hat{u}(m,t,q)$ leads to

$$\hat{u}(m,t,q) = \begin{cases} 0, & \text{if } m = 0 \\ |m|^{-2t}\frac{(T_q\hat{u})(m)}{M_q^t(m)}, & \text{else}\,. \end{cases}$$

This completes the proof. □

C.1.3 Proof of Lemma 6.2.15

Proof. Recall the definition

$$M_q^t(m) := \begin{cases} \sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |2^q\beta|^{-2t}, & \text{if } m = j\cdot 2^q \text{ for some } j\in\mathbb{Z}^d \\ \sum_{\beta\in\mathbb{Z}^d} |m + 2^q\beta|^{-2t}, & \text{else}\,. \end{cases}$$

Because of the periodicity of M_q^t, we need only to study $m \in B_q^d$. We split it into two cases.

1. If $m = 0$, then $M_q^t(m) = \sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |2^q\beta|^{-2t} = 2^{-2qt}\sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |\beta|^{-2t} \simeq 2^{-2qt}$.

2. If $m \in B_q^d\setminus\{0\}$, then $M_q^t(m) = \sum_{\beta\in\mathbb{Z}^d} |m + 2^q\beta|^{-2t} = |m|^{-2t} + \sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |m + 2^q\beta|^{-2t}$. Since $B_q^d = [-2^{q-1}, 2^{q-1} - 1]^{\otimes d}$, each component of $m \in B_q^d$ is bounded by 2^{q-1} in amplitude, and therefore each component of $2^{-q}m$ is bounded by $1/2$ in amplitude. So, it follows that

$$\sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |m + 2^q\beta|^{-2t} = 2^{-2qt}\sum_{\beta\in\mathbb{Z}^d\setminus\{0\}} |2^{-q}m + \beta|^{-2t} \simeq 2^{-2qt}\,.$$

   Then, we get $|m|^{-2t} \le M_q^t(m) \lesssim |m|^{-2t} + 2^{-2qt} \lesssim |m|^{-2t}$ where we have used the fact that $|m| \lesssim 2^q$. Therefore, it holds that $M_q^t(m) \simeq |m|^{-2t}$.

As a byproduct of the above proof, we also get $M_q^t(m) - |m|^{-2t} \simeq 2^{-2qt}$. □

C.1.4 Proof of Lemma 6.2.17

Proof. First, we prove the pointwise convergence (i.e., for each fixed r), then move on to prove uniform convergence. To achieve this, we calculate the variance:

$$\begin{aligned}\mathrm{Var}(\alpha(r,q)) &\simeq 2^{-2rq}\sum_{m\in B_q^d\setminus\{0\}} |m|^{2r-2d}\\ &\lesssim 2^{-2rq}\int_1^{2^q} x^{2r-2d+d-1}\,\mathrm{d}x = 2^{-2rq}\int_1^{2^q} x^{2r-d-1}\,\mathrm{d}x\,.\end{aligned}$$

For $r = d/2$, the integral gives $\log(2^q) = q\log 2$; for $r \neq d/2$, it is $\frac{1}{2r-d}(2^{q(2r-d)} - 1)$. In both cases, we have $\lim_{q\to\infty}\mathrm{Var}(\alpha(r,q)) = 0$. Thus, $\alpha(r,q)$ converges in L^2 to the limit of its expectation, which we may calculate as follows:

$$\lim_{q\to\infty}\mathbb{E}\alpha(r,q) = \lim_{q\to\infty}\sum_{m\in B_q^d\setminus\{0\}} (2^{-q})^d|2^{-q}m|^{r-d} = \int_{[-1/2,1/2]^d} |y|^{r-d}\,\mathrm{d}y := \gamma(r) > 0\,.$$

Hence, we get $\lim_{q\to\infty}\alpha(r,q) = \gamma(r) > 0$ in L^2 for every $r \in [\epsilon, 1/\epsilon]$, and the convergence also holds in probability. We may now proceed to show uniform convergence. We rely on Exercise 3.2.3 in [277]. Based on that, it suffices to prove $\alpha(r,q)$ is uniformly Lipschitz continuous as a function of r for $q \in \mathbb{N}$. Pick any $r_1, r_2 \in [\epsilon, 1/\epsilon]$, then

$$\begin{aligned}&|\alpha(r_1,q) - \alpha(r_2,q)|\\ =&\sum_{m\in B_q^d\setminus\{0\}} 2^{-qd}|(|2^{-q}m|^{r_1-d} - |2^{-q}m|^{r_2-d})|\\ \le&\sum_{m\in B_q^d\setminus\{0\}} 2^{-qd}|r_1 - r_2|(|2^{-q}m|^{\epsilon-d} + |2^{-q}m|^{1/\epsilon-d})|\log(2^{-q}m)|\xi_m^2\,,\end{aligned}$$

where in the last step we have used the fact that $||2^{-q}m|^{r_1-d} - |2^{-q}m|^{r_2-d}| = ||2^{-q}m|^{\eta-d}\log(2^{-q}m)(r_1-r_2)|$ for some η that lies between r_1 and r_2, and we use the bound $r_1, r_2 \in [\epsilon, 1/\epsilon]$. Now, we define the random series:

$$\mathsf{L}(q) := 2^{-qd} \sum_{m \in B_q^d \setminus \{0\}} (|2^{-q}m|^{\epsilon-d} + |2^{-q}m|^{1/\epsilon-d})|\log(2^{-q}m)|\xi_m^2 \,.$$

We calculate its variance as follows:

$$\begin{aligned}
\mathrm{Var}(\mathsf{L}(q)) &\simeq 2^{-2dq} \sum_{m \in B_q^d \setminus \{0\}} (|2^{-q}m|^{2\epsilon-2d} + |2^{-q}m|^{2/\epsilon-2d}) \log^2 |2^{-q}m| \\
&\lesssim 2^{-qd} \left(\int_{2^{-q}}^1 t^{2\epsilon-2d+d-1} \log^2 t \, \mathrm{d}t + \int_{2^{-q}}^1 t^{2/\epsilon-2d+d-1} \log^2 t \, \mathrm{d}t \right) \\
&= 2^{-qd} \int_{2^{-q}}^1 (t^{2\epsilon-d-1} + t^{2/\epsilon-d-1}) \log^2 t \, \mathrm{d}t \,, \\
&\lesssim 2^{-qd} \int_{2^{-q}}^1 (t^{\epsilon-d-1} + t^{1/\epsilon-d-1}) \, \mathrm{d}t \lesssim 2^{-q\epsilon} \,.
\end{aligned}$$

The last term will go to 0 as q goes to infinity. Thus, $\mathsf{L}(q)$ converges in L^2 (and thus in probability) to $\mathsf{L}^* = \lim_{q\to\infty} \mathbb{E}\mathsf{L}(q)$, which is

$$\begin{aligned}
\lim_{q\to\infty} \mathbb{E}\mathsf{L}(q) &= \lim_{q\to\infty} 2^{-qd} \sum_{m \in B_q^d \setminus \{0\}} (|2^{-q}m|^{\epsilon-d} + |2^{-q}m|^{1/\epsilon-d}) \log^2 |2^{-q}m| \\
&= \int_{[-1/2,1/2]^d} (|y|^{\epsilon-d} + |y|^{1/\epsilon-d}) \log^2 |y| \, \mathrm{d}y \\
&\lesssim \int_{[-1/2,1/2]^d} (|y|^{\epsilon/2-d} + |y|^{1/(2\epsilon)-d}) \, \mathrm{d}y < \infty \,.
\end{aligned}$$

Using Markov's inequality we deduce that, for any $\epsilon' > 0$, it holds that

$$\mathbb{P}(|\mathsf{L}(q) - \mathsf{L}^*| \geq \epsilon') \leq \frac{\mathbb{E}|\mathsf{L}(q) - \mathsf{L}^*|^2}{(\epsilon')^2} \leq \frac{2^{-q\epsilon}}{(\epsilon')^2} \,.$$

Thus,

$$\sum_{q=1}^{\infty} \mathbb{P}(|\mathsf{L}(q) - \mathsf{L}^*| \geq \epsilon') \leq \sum_{q=1}^{\infty} \frac{2^{-q\epsilon}}{(\epsilon')^2} < \infty \,.$$

From the Borel-Cantelli lemma it follows that $\lim_{q\to\infty} \mathsf{L}(q) = \mathsf{L}^*$ almost surely, and therefore $\mathsf{L}(q)$ is bounded uniformly for q almost surely. Since $|\alpha(r_1, q) - \alpha(r_2, q)| \leq \mathsf{L}(q)|r_1 - r_2|$, it follows that $\alpha(r, q)$ is uniformly Lipschitz continuous as a function of r for $q \in \mathbb{N}$. Invoking Exercise 3.2.3 in [277] concludes this case.

For the case $r = 0$, we follow the same strategy as in the previous case. First, we

calculate the corresponding variance:

$$\begin{aligned}\mathrm{Var}(\alpha(0,q)) &\simeq \frac{1}{q^2}\sum_{m\in B_q^d\setminus\{0\}} |m|^{-2d} \\ &\lesssim \frac{1}{q^2}\int_1^{2^q} x^{-2d+d-1}\,\mathrm{d}x \lesssim \frac{1}{q^2},\end{aligned}$$

where the last term goes to 0 as q goes to infinity. Then, we calculate the expectation:

$$\mathbb{E}\alpha(0,q) = \frac{1}{q}\sum_{m\in B_q^d\setminus\{0\}} |m|^{-d}.$$

The limit when $q\to\infty$ is identified through the following calculations:

$$\begin{aligned}\lim_{q\to\infty}\frac{1}{q}\sum_{m\in B_q^d\setminus\{0\}} |m|^{-d} &= \lim_{q\to\infty}\sum_{m\in B_{q+1}^d\setminus B_q^d} |m|^{-d} \\ &= \lim_{q\to\infty} 2^{-qd}\sum_{m\in B_{q+1}^d\setminus B_q^d} |2^{-q}m|^{-d} \\ &= \int_{[-1,1]^d\setminus[-1/2,1/2]^d} |x|^{-d}\,\mathrm{d}x < \infty;\end{aligned}$$

here we have used the definition of the Riemann integral. Finally, we conclude that $\lim_{q\to\infty}\alpha(0,q) = \gamma(0)$ in probability for $\gamma(0)\in(0,\infty)$. □

C.1.5 Proof of Proposition 6.2.18

Proof. First, we have the relation

$$\log\det K(t,q) = \log\det K(t,q-1) + \log\det(K(t,q)/K(t,q-1)),$$

where $K(t,q)/K(t,q-1)$ is the Schur complement of $K(t,q-1)$ in $K(t,q)$. Due to the variational property of the Schur complement (see Lemma 13.24 in [204]), the smallest and largest eigenvalues of $K(t,q)/K(t,q-1)$ satisfy (in the dual norm $\|\cdot\|_{-t}$)

$$\begin{aligned}&\lambda_{\min}(K(t,q)/K(t,q-1)) \geq \inf_{y\in\mathbb{R}^{|J_q|}} \frac{\|\sum_{j\in J_q} y_j\delta(x-x_j)\|_{-t}^2}{|y|^2}, \quad \text{and} \\ &\lambda_{\max}(K(t,q)/K(t,q-1)) \\ &\qquad = \sup_{y\in\mathbb{R}^{|J_q|}}\inf_{z\in\mathbb{R}^{|J_{q-1}|}} \frac{\|\sum_{j\in J_q} y_j\delta(x-x_j) - \sum_{j'\in J_{q-1}} z_{j'}\delta(x-x_{j'})\|_{-t}^2}{|y|^2}.\end{aligned} \tag{C.1.3}$$

These two formulae will be crucial in the subsequent analysis. We start by estimating the smallest and largest eigenvalues of the Schur complement. Let

$w = (-\Delta)^{-t} \sum_{j\in J_q} y_j \delta(x - x_j)$, whose Fourier coefficients are

$$\hat{w}(m) = \begin{cases} 0, & \text{if } m = 0 \\ (4\pi^2)^{-t}|m|^{-2t} g(m), & \text{else}\,, \end{cases} \tag{C.1.4}$$

where, the function $g(m)$ is defined by

$$g(m) = \sum_{j\in J_q} y_j \exp(2\pi i \langle j 2^{-q}, m\rangle)\,. \tag{C.1.5}$$

For the smallest eigenvalue, we write

$$\begin{aligned} \|\sum_{j\in J_q} y_j \delta(x - x_j)\|_{-t}^2 = \|w\|_t^2 &= (4\pi^2)^t \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{2t} |\hat{w}(m)|^2 \\ &= (4\pi^2)^{-t} \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t} |g(m)|^2. \end{aligned}$$

Notice that

$$\sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t}|g(m)|^2 = \sum_{m\in B_q^d} M_q^t(m)|g(m)|^2 \gtrsim 2^{-2tq} \sum_{m\in B_q^d} |g(m)|^2$$

and

$$\begin{aligned} \sum_{m\in B_q^d} |g(m)|^2 &= \sum_{m\in B_q^d} |\sum_{j\in J_q} y_j \exp(2\pi i\langle j 2^{-q}, m\rangle)|^2 \\ &= \sum_{m\in B_q^d} \sum_{j\in J_q} \sum_{l\in J_q} y_j y_l \exp(2\pi i\langle (j-l) 2^{-q}, m\rangle) \\ &= \sum_{j\in J_q} \sum_{l\in J_q} y_j y_l \sum_{m\in B_q^d} \exp(2\pi i\langle (j-l) 2^{-q}, m\rangle) \\ &\simeq 2^{qd}|y|^2\,. \end{aligned} \tag{C.1.6}$$

In the last line we have used the fact that

$$\sum_{m\in B_q^d} \exp(2\pi i\langle (j-l)2^{-q}, m\rangle) = \begin{cases} 0, & \text{if } j - l \neq 0 \\ \sum_{m\in B_q^d} 1 \simeq 2^{qd}, & \text{if } j - l = 0\,. \end{cases}$$

Thus, combining the above results, we obtain the bound on the smallest eigenvalue

$$\lambda_{\min}(K(t,q)/K(t,q-1)) \gtrsim 2^{-q(2t-d)}\,.$$

We then move to consider the largest eigenvalue. First, notice that

$$\inf_{z\in\mathbb{R}^{|J_{q-1}|}} \|\sum_{j\in J_q} y_j \delta(x - x_j) - \sum_{j'\in J_{q-1}} z_{j'} \delta(x - x_{j'})\|_{-t}^2 = \inf_{v\in\mathcal{F}_{t,q-1}} \|w - v\|_t^2\,.$$

Naturally, one can express the optimal v in the above variational formulation using the Fourier series representation explained before. However, this will lead to many interactions between different frequencies. To make the analysis cleaner, we adopt another strategy. We first approximate the function w by a band-limited function, whose projection into $\mathcal{F}_{t,q-1}$ will be more concise. Precisely, define a band limited version of w, written as w_h, by

$$\hat{w}_h(m) = \begin{cases} \hat{w}(m), & \text{if } m \in B^d_{q-1} \\ 0, & \text{if } m \in (B^d_{q-1})^c \,. \end{cases} \tag{C.1.7}$$

To estimate $\inf_{v\in\mathcal{F}_{t,q-1}} \|w - v\|_t^2$, we follow the two steps below:

Step 1: we prove $\|w - w_h\|_t^2 \lesssim 2^{-q(2t-d)}|y|^2$. Let us calculate the quantity directly:

$$\begin{aligned} \|w - w_h\|_t^2 &= (4\pi^2)^{-t} \sum_{m\in(B^d_{q-1})^c} |m|^{-2t}|g(m)|^2 \\ &= (4\pi^2)^{-t}\left(\sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t}|g(m)|^2 - \sum_{m\in B^d_{q-1}} |m|^{-2t}|g(m)|^2\right) \\ &= (4\pi^2)^{-t}\left(\sum_{m\in B^d_q} M_q^t(m)|g(m)|^2 - \sum_{m\in B^d_{q-1}} |m|^{-2t}|g(m)|^2\right) \\ &\lesssim 2^{-2qt}\sum_{m\in B^d_q} |g(m)|^2 \lesssim 2^{-q(2t-d)}|y|^2. \end{aligned}$$

Here we have used the fact that $M_q^t(m) - |m|^{-2t} \lesssim 2^{-2qt}$ for $m \in B^d_{q-1}$ and $M_q^t(m) \lesssim 2^{-2qt}$ for $m \in B^d_q \backslash B^d_{q-1}$, according to the results in Lemma 6.2.15. In the last line, the bound (C.1.6) is applied.

Step 2: We prove $\inf_{v\in\mathcal{F}_{t,q-1}} \|w_h - v\|_t^2 \lesssim 2^{-q(2t-d)}|y|^2$. Based on Theorem 6.2.13, we know the optimal v for this variational problem has the Fourier coefficients

$$\hat{v}(m) = \begin{cases} 0, & \text{if } m = 0 \\ |m|^{-2t}\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}, & \text{else} \,. \end{cases}$$

Then, using the Fourier representation of the norm, we get

$$\begin{aligned}
&\|w_h - v\|_t^2 \\
&\simeq \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{2t}|\hat{w}_h(m) - \hat{v}(m)|^2 \\
&= \sum_{m\in B^d_{q-1}\setminus\{0\}} |m|^{-2t}|g(m) - \frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 + \sum_{m\in (B^d_{q-1})^c} |m|^{-2t}|\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 .
\end{aligned}$$

For the first term, since w_h is band-limited, we know if $m \in B^d_{q-1}\setminus\{0\}$, then $(T_{q-1}\hat{w}_h)(m) = |m|^{-2t}g(m)$. Thus, we can write this term as

$$\begin{aligned}
&\sum_{m\in B^d_{q-1}\setminus\{0\}} |m|^{-2t}|g(m)|^2\Big(1 - \frac{|m|^{-2t}}{M^t_{q-1}(m)}\Big)^2 \\
&= \sum_{m\in B^d_{q-1}\setminus\{0\}} |m|^{-2t}|g(m)|^2\Big(\frac{M^t_{q-1}(m) - |m|^{-2t}}{M^t_{q-1}(m)}\Big)^2 \\
&\overset{a)}{\lesssim} \sum_{m\in B^d_{q-1}\setminus\{0\}} |m|^{-2t}|g(m)|^2\Big(\frac{2^{-4tq}}{|m|^{-4t}}\Big) \\
&\overset{b)}{\lesssim} \sum_{m\in B^d_{q-1}\setminus\{0\}} 2^{-2tq}|g(m)|^2 \lesssim 2^{-q(2t-d)}|y|^2 ,
\end{aligned}$$

where in a) we have used the fact that $M^t_{q-1}(m) - |m|^{-2t} \simeq 2^{-2tq}$ and $M^t_{q-1}(m) \simeq |m|^{-2t}$ for $m \in B^d_{q-1}\setminus\{0\}$ based on Lemma 6.2.15. In b), we have used $|m| \lesssim 2^q$. The last inequality is obtained by recalling (C.1.6).

For the second term, we write

$$\begin{aligned}
&\sum_{m\in (B^d_{q-1})^c} |m|^{-2t}|\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 \\
&= \sum_{m\in\mathbb{Z}^d\setminus\{0\}} |m|^{-2t}|\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 - \sum_{m\in B^d_{q-1}\setminus\{0\}} |m|^{-2t}|\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 \\
&\overset{c)}{=} \sum_{m\in B^d_{q-1}\setminus\{0\}} (M^t_{q-1}(m) - |m|^{-2t})|\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}|^2 \\
&= \sum_{m\in B^d_{q-1}\setminus\{0\}} (M^t_{q-1}(m) - |m|^{-2t})|\frac{|m|^{-2t}g(m)}{M^t_{q-1}(m)}|^2 \\
&\lesssim 2^{-2tq} \sum_{m\in B^d_{q-1}\setminus\{0\}} |g(m)|^2 \lesssim 2^{-q(2t-d)}|y|^2 ,
\end{aligned}$$

where in c), we have used the periodicity of the function $\frac{(T_{q-1}\hat{w}_h)(m)}{M^t_{q-1}(m)}$.

Now, combining Step 1 and 2 leads to the conclusion

$$\inf_{v\in\mathcal{F}_{t,q-1}} \|w-v\|_t^2 \lesssim 2^{-q(2t-d)}|y|^2\,,$$

and in particular, it implies

$$\lambda_{\max}(K(t,q)/K(t,q-1)) \lesssim 2^{-q(2t-d)}\,.$$

As a consequence of the upper and lower bounds for the eigenvalues of the matrix $K(t,q)/K(t,q-1)$, we deduce that they are all on the scale of $2^{-q(2t-d)}$. Let C be a constant independent of t,q such that $C^{-1}2^{-q(2t-d)} \le K(t,q)/K(t,q-1) \le C2^{-q(2t-d)}$. Then,

$$\begin{aligned}(2^{qd}-2^{(q-1)d})((2t-d)(-q)\log 2 - C) &\le \log\det K(t,q)/K(t,q-1)\\ &\le (2^{qd}-2^{(q-1)d})((2t-d)(-q)\log 2 + C)\,.\end{aligned}$$

Using the implied bounds on the recursion relation, we get

$$(2t-d)g_1(q) - Cg_2(q) + K(t,0) \le \log\det K(t,q) \le (2t-d)g_1(q) + Cg_2(q) + K(t,0)\,,$$

where $g_1(q) = \sum_{k=1}^{q}(2^{kd}-2^{(k-1)d})(-k\log 2)$ and $g_2(q) = (2^{qd}-1)(2t-d)$. Summing the series in $g_1(q)$ leads to $g_1(q) \simeq -q2^{qd}\log 2 \simeq -q2^{qd}$. The proof of Proposition 6.2.18 is completed.

Remark C.1.1. *The above technique of using the Schur complements is quite general and could be potentially applied to other operators such as heterogeneous Laplacians; see [204]. However, for the homogeneous Laplacian on the torus in this paper, we may also prove the result via a simpler approach. The key observation is that there is an explicit formula for the spectrum of $K(t,q)$, as also exploited in [258, Sec. 6.7]. Indeed, using the formula for the spectrum given in Lemma C.1.2 below, we get*

$$\begin{aligned}\log\det K(t,q) &= \sum_{m\in B_q^d} \log\left(2^{qd}(4\pi^2)^{-t}M_q^t(m)\right)\\ &= qd2^{qd}\log 2 - 2^{qd}t\log\left(4\pi^2\right) + \sum_{m\in B_q^d}\log M_q^t(m)\,.\end{aligned}$$

By Lemma 6.2.15, it holds that

$$M_q^t(m) \simeq \begin{cases} 2^{-2qt}, \text{ if } m=0\\ |m|^{-2t}, \text{ if } m\in B_q^d\backslash\{0\}\,.\end{cases}$$

That is, there exists a constant C independent of t such that

$$-2t\log|m| - \log C \le \log M_q^t(m) \le -2t\log|m| + \log C$$

for $m \in B_q^d\backslash\{0\}$, and $-2^{qt}\log 2 - \log C \le \log M_q^t(0) \le -2^{qt}\log 2 + \log C$. Since

$$\sum_{m\in B_q^d\backslash\{0\}} \log|m| \simeq \int_0^{2^q} r^{d-1}\log r\,\mathrm{d}r \simeq q2^{qd}\,,$$

and $2^{qd} = o(q2^{qd})$, we get

$$-(2t-d)q2^{qd} - C2^{qd} \lesssim \log\det K(t,q) \lesssim -(2t-d)q2^{qd} + C2^{qd}\,.$$

This completes the alternative proof of Proposition 6.2.18.

□

Lemma C.1.2. *The eigenvalues of $K(t,q)$ are $2^{qd}(4\pi^2)^{-t}M_q^t(m)$ for $m \in B_q^d$, where $M_q^t(m)$ is defined in* (6.2.12)*, with the corresponding eigenfunctions $\phi_m(\mathcal{X}_q) \in \mathbb{R}^{2^{qd}}$.*

Proof. We can prove this claim using Mercer's decomposition as follows. First, for $x_i, x_j \in \mathcal{X}_q$, it holds that

$$\begin{aligned} K(t,q)_{i,j} &= \sum_{m\in\mathbb{Z}^d\backslash\{0\}} (4\pi^2)^{-t}|m|^{-2t}\phi_m(x_i)\phi_m^*(x_j) \\ &= \sum_{m\in B_q^d} (4\pi^2)^{-t}M_q^t(m)\phi_m(x_i)\phi_m^*(x_j)\,, \end{aligned}$$

where we have used the fact that $\phi_{m+2^q\beta}(x_i) = \phi_m(x_i)$ for any $\beta \in \mathbb{Z}^d$ and $x_i \in \mathcal{X}_q$. Thus, for every $n \in B_q^d$, we get

$$\begin{aligned} \sum_{x_j\in\mathcal{X}_q} K(t,q)_{i,j}\phi_n(x_j) &= \sum_{m\in B_q^d} (4\pi^2)^{-t}M_q^t(m)\phi_m(x_i) \sum_{x_j\in\mathcal{X}_q} \phi_m^*(x_j)\phi_n(x_j) \\ &= \sum_{m\in B_q^d} (4\pi^2)^{-t}M_q^t(m)\phi_m(x_i)2^{qd}\delta_{mn} \\ &= 2^{qd}(4\pi^2)^{-t}M_q^t(m)\phi_n(x_i)\,, \end{aligned}$$

where in the second equality we used the property of Fourier series. This implies $\phi_n(\mathcal{X}_q)$ is an eigenfunction. The proof of the lemma is completed. □

C.1.6 Proof of Theorem 6.2.19

Proof. Recall the definition,

$$s^{\mathrm{EB}}(q) = \arg\min_t \mathsf{L}^{\mathrm{EB}}(t,q) := \|u(\cdot,t,q)\|_t^2 + \log\det K(t,q)\,.$$

Define a rescaled version of the loss function by

$$\tilde{L}_{\mathrm{EB}}(t,q) = \frac{1}{|g_1(q)|}\mathsf{L}^{\mathrm{EB}}(t,q) = \underbrace{\frac{1}{|g_1(q)|}\|u(\cdot,t,q)\|_t^2}_{①} + \underbrace{\frac{1}{|g_1(q)|}\log\det K(t,q)}_{②}\,.$$

We note that by Proposition 6.2.18, we have $|g_1(q)| \sim q2^{qd}$. Now, we estimate the growth rate of ① and ② separately. From Proposition 6.2.16 and 6.2.18, we get

$$① \simeq \underbrace{\frac{1}{q}2^{-q(2s-2t+d)}\xi_0^2}_{③} + \underbrace{\frac{1}{q}2^{-q(2s-2t)}\sum_{m\in B_q^d\setminus\{0\}} 2^{-q(2t-2s+d)}|m|^{2t-2s}\xi_m^2}_{④},$$

and for the log det part, it holds that

$$d - 2t + \frac{-Cg_2(q) + K(t,0)}{|g_1(q)|} \le ② \le d - 2t + \frac{Cg_2(q) + K(t,0)}{|g_1(q)|}\,.$$

It follows that $\lim_{q\to\infty} ② = d - 2t$. Thus, our remaining task is to analyze terms ③, ④ in ①. We split the problem into four cases.

Case 1: $t = s$. It is easy to see $\lim_{q\to\infty} ③ = 0$ and

$$④ = \frac{1}{q}2^{-qd}\sum_{m\in B_q^d\setminus\{0\}}\xi_m^2 = \frac{1}{q}\alpha(d,q)\,,$$

so that $\lim_{q\to\infty} ④ = 0$. Here we use the definition of α in Lemma 6.2.17. Therefore, $\lim_{q\to\infty}\tilde{L}_{\mathrm{EB}}(s,q) = d - 2s$.

Case 2: $1/\delta \ge t \ge s + \epsilon$. We have ③ ≥ 0. The term ④ can be written as

$$④ = \frac{1}{q2^{-q(2t-2s)}}\alpha(2t-2s+d,q)\,,$$

where we recall the definition of the function α in Lemma 6.2.17. According to this lemma, we get the uniform convergence

$$\lim_{q\to\infty}\alpha(2t-2s+d,q) = \gamma(2t-2s+d) > 0$$

in probability. In the meantime, $\lim_{q\to\infty} q2^{-q(2t-2s)} = 0$. So, $\lim_{q\to\infty} ④ = \infty$ in probability, and uniformly in $1/\delta \geq t \geq s+\epsilon$. In terms of $\tilde{L}_{\mathrm{EB}}(t,q)$, this corresponds to $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(t,q) = \infty$.

Case 3: $s - \epsilon \geq t \geq s - d/2 + \epsilon$. In this case, $2t - 2s + d \geq \epsilon$ so Lemma 6.2.17 can be applied. We write the term

$$④ = \frac{2^{-q(2s-2t)}}{q}\alpha(2t-2s+d,q)\,.$$

This will converge to 0 as q goes to infinity, since $\lim_{q\to\infty} \frac{2^{-q(2s-2t)}}{q} = 0$ and $\lim_{q\to\infty} \alpha(2t-2s+d,q) = \gamma(2t-2s+d) \in (0,\infty)$. The term ③ also converges to 0. Thus, $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(t,q) = d - 2t$ in probability, and uniformly for $s - \epsilon \geq t \geq s - d/2 + \epsilon$.

Case 4: $s - d/2 + \epsilon \geq t \geq d/2 + \delta$. We still have that ③ converges to 0. For term ④, we have

$$④ = \frac{2^{-qd}}{q}\sum_{m\in B_q^d\setminus\{0\}} |m|^{2t-2s}\xi_m^2 \leq \frac{2^{-qd}}{q}\sum_{m\in B_q^d\setminus\{0\}} |m|^{2(s-d/2+\epsilon)-2s}\xi_m^2\,,$$

where we have used the monotonicity of the function $|m|^{2t-2s}$ with respect to t. Then, it reduces to the case $t = s - d/2 + \delta$, which is covered by Case 3. Hence, we have $\lim_{q\to\infty} ④ = 0$ uniformly for $s - d/2 + \delta \geq t \geq d/2 + \delta$. Therefore, we get $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(t,q) = d-2t$ in probability, and uniformly for $s-d/2+\delta \geq t \geq d/2+\delta$.

Let us make a summary of the arguments above. We have established that, for any small $\epsilon > 0$, $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(t,q) = \infty$ uniformly for $1/\delta \geq t \geq s+\epsilon$, and $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(t,q) = d-2t$ uniformly for $s-\epsilon \geq t \geq d/2+\delta$, and $\lim_{q\to\infty} \tilde{L}_{\mathrm{EB}}(s,q) = d - 2s$. All the convergence is in probability. Note that s^{EB} is the minimizer of $L_{\mathrm{EB}}(t,q)$, hence also of $\tilde{L}_{\mathrm{EB}}(t,q)$. The above convergence results for $\tilde{L}_{\mathrm{EB}}(t,q)$ imply that $s^{\mathrm{EM}} \in (s-\epsilon, s+\epsilon)$ with probability 1 as q goes to infinity, for any $\epsilon > 0$. Thus, we must have

$$\lim_{q\to\infty} s^{\mathrm{EB}}(q) = s\,.$$

The proof is complete. □

C.1.7 Proof of Proposition 6.2.21

Proof. In order to write the interaction terms as a random series with some desired independence pattern for the random variables involved, we need to consider the geometry of the lattice carefully. We introduce another set $S_q := \{m \in \mathbb{Z} : -2^{q-2} \leq$

$m \leq 3 \times 2^{q-2} - 1\}$ and let $S_q^d = S_q \otimes S_q \otimes \cdots \otimes S_q$ denote the tensor product of d multiples of S_q. The set S_q is a shift of B_q, and S_q^d is a shift of B_q^d.

Define the set $B_{q-1}^d + 2^{q-1}k := \{m + 2^{q-1}k : m \in B_{q-1}^d\}$ for $k \in \mathbb{Z}^d$. We have the relation

$$S_q^d = \bigcup_{k \in \mathbb{Z}_2^d} (B_{q-1}^d + 2^{q-1}k),$$

where $\mathbb{Z}_2^d = \{0, 1\}^d$. Note that for $k_1 \neq k_2$, the intersection between $B_{q-1}^d + 2^{q-1}k_1$ and $B_{q-1}^d + 2^{q-1}k_2$ is empty.

Using (6.2.13) and the periodicity of the functions involved, we get

$$\begin{aligned}
&\|u(\cdot, t, q) - u(\cdot, t, q-1)\|_t^2 \\
=&(4\pi^2)^t \sum_{m \in B_q^d} M_q^t(m) \left(\frac{T_q \hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \right)^2 \\
=&(4\pi^2)^t \sum_{m \in S_q^d} M_q^t(m) \left(\frac{T_q \hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \right)^2 \\
=&(4\pi^2)^t \sum_{k \in \mathbb{Z}_2^d} \sum_{m \in (B_{q-1}^d + 2^{q-1}k)} M_q^t(m) \left(\frac{T_q \hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \right)^2 .
\end{aligned}$$

Recall the relation

$$T_{q-1}\hat{u}(m) = \sum_{l \in \mathbb{Z}_2^d} T_q \hat{u}(m + 2^{q-1}l),$$

based on which we get

$$\begin{aligned}
&\frac{T_q \hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)} \\
=&(\frac{1}{M_q^t(m)} - \frac{1}{M_{q-1}^t(m)}) T_q \hat{u}(m) - \frac{1}{M_{q-1}^t} \sum_{l \in \mathbb{Z}_2^d \setminus \{0\}} T_q \hat{u}(m + 2^{q-1}l).
\end{aligned}$$

Since $u^\dagger \sim \mathcal{N}(0, (-\Delta)^{-s})$, it holds $\hat{u}(m) \sim \mathcal{N}(0, (4\pi^2)^{-s}|m|^{-2s})$. Moreover, for different m, these Gaussian random variables are independent from each other. Thus, for a fixed k and for $m \in (B_{q-1}^d + 2^{q-1}k)$, the Gaussian random variables

$$\frac{T_q \hat{u}(m)}{M_q^t(m)} - \frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)}$$

are independent from each other. Furthermore, by calculating their variance, we can write

$$
\begin{aligned}
&M_q^t(m)\left(\frac{T_q\hat{u}(m)}{M_q^t(m)}-\frac{T_{q-1}\hat{u}(m)}{M_{q-1}^t(m)}\right)^2\\
=&(4\pi^2)^{-s}\left[(\frac{1}{M_q^t(m)}-\frac{1}{M_{q-1}^t(m)})^2M_q^t(m)M_q^s(m)+\frac{M_q^t(m)}{(M_{q-1}^t(m))^2}\sum_{l\in\mathbb{Z}_2^d\setminus\{0\}}M_q^s(m+2^{q-1}l)\right]\xi_{k,m}^2\\
=&(4\pi^2)^{-s}\left[\frac{M_q^s(m)(M_q^t(m)-M_{q-1}^t(m))^2}{M_q^t(m)(M_{q-1}^t(m))^2}+\frac{M_q^t(m)}{(M_{q-1}^t(m))^2}\sum_{l\in\mathbb{Z}_2^d\setminus\{0\}}M_q^s(m+2^{q-1}l)\right]\xi_{k,m}^2\\
=&:A_{k,m}\xi_{k,m}^2,
\end{aligned}
$$

where $\{\xi_{k,m}\}_m$ are independent unit scalar Gaussian random variables. Clearly, we have the lower bound

$$A_{k,m}\geq(4\pi^2)^{-s}\frac{M_q^t(m)}{(M_{q-1}^t(m))^2}M_q^s(m-2^{q-1}k)\,.$$

Thus, denoting $e_1=(1,0,...,0)\in\mathbb{Z}^d$, we get

$$
\begin{aligned}
&\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2\\
\geq&(4\pi^2)^{t-s}\sum_{k\in\mathbb{Z}_2^d}\sum_{m\in(B_{q-1}^d+2^{q-1}k)}\frac{M_q^t(m)}{(M_{q-1}^t(m))^2}M_q^s(m-2^{q-1}k)\xi_{k,m}^2\\
\geq&(4\pi^2)^{t-s}\sum_{m\in(B_{q-1}^d+2^{q-1}e_1)}\frac{M_q^t(m)}{(M_{q-1}^t(m))^2}M_q^s(m-2^{q-1}e_1)\xi_{e_1,m}^2\\
=&(4\pi^2)^{t-s}\sum_{m\in(B_{q-1}^d+2^{q-1}e_1)}\frac{M_q^t(m)}{(M_{q-1}^t(m-2^{q-1}e_1))^2}M_q^s(m-2^{q-1}e_1)\xi_{e_1,m}^2\\
\gtrsim&\sum_{m\in(B_{q-1}^d\setminus\{0\}+2^{q-1}e_1)}\frac{2^{-2qt}}{|m-2^{q-1}e_1|^{-4t}}|m-2^{q-1}e_1|^{2s}\xi_{e_1,m}^2\\
=&\sum_{m\in B_{q-1}^d\setminus\{0\}}2^{-2qt}|m|^{4t-2s}\xi_{e_1,m+2^{q-1}e_1}^2\,.
\end{aligned}
$$

In the above derivation, we have used the fact that for $m\in B_{q-1}^d$, it holds that $M_q^s(m)\simeq|m|^{-2s}, M_{q-1}^t(m)\simeq|m|^{-2t}$, and in particular, $M_q^t(m)\simeq|m|^{-2t}\simeq 2^{-2qt}$ for $m\in(B_{q-1}^d\setminus\{0\}+2^{q-1}e_1)$. Renaming the subscripts in $\xi_{e_1,m+2^{q-1}e_1}$ completes the proof. □

C.1.8 Proof of Proposition 6.2.22

Proof. We need to upper bound $A_{k,m}$ for $k \in \mathbb{Z}_2^d, m \in B_{q-1}^d + 2^{q-1}k$, which is defined in the proof of Proposition 6.2.21. First, we have

$$\sum_{l \in \mathbb{Z}_2^d \setminus \{0\}} M_q^s(m + 2^{q-1}l) = M_{q-1}^s(m) - M_q^s(m),$$

and the estimate $0 \le M_{q-1}^t(m) - M_q^t(m) \le M_{q-1}^t(m)$ for any $d/2 + \delta \le t \le 1/\delta$. Based on this observation, for $k \in \mathbb{Z}^d \setminus \{0\}$ and $m \in B_{q-1}^d \setminus \{0\} + 2^{q-1}k$, we have the bound

$$\begin{aligned} A_{k,m} &\lesssim \frac{M_q^s(m)}{M_q^t(m)} + M_q^t(m)\frac{M_{q-1}^s(m)}{(M_{q-1}^t(m))^2} \\ &\lesssim 2^{-q(2s-2t)} + 2^{-2tq}|m - 2^{q-1}k|^{4t-2s}, \end{aligned}$$

where we have used the fact that for $m \in B_{q-1}^d \setminus \{0\} + 2^{q-1}k$, it holds that $M_q^s(m) \simeq 2^{-2sq}, M_q^t(m) \simeq 2^{-2tq}, M_{q-1}^s(m) \simeq |m - 2^{q-1}k|^{-2s}, M_{q-1}^t(m) \simeq |m - 2^{q-1}k|^{-2t}$, according to Lemma 6.2.15. For $m = 2^{q-1}k$, we get $A_{k,m} \lesssim 2^{-q(2s-2t)}$. So in general, we can write $A_{k,m} \lesssim 2^{-q(2s-2t)} + 2^{-2tq}|m - 2^{q-1}k|^{4t-2s}$ for $m \in B_{q-1}^d + 2^{q-1}k$ where we use the convention that $|m|^\alpha = 0$ for $m = 0$ and any $\alpha \in \mathbb{R}$ to make the notation more compact.

When $k = 0$, using Lemma 6.2.15 again, we get for $m \in B_{q-1}^d \setminus \{0\}$,

$$\begin{aligned} A_{k,m} &\lesssim \frac{M_q^s(m)(M_q^t(m) - M_{q-1}^t(m))^2}{M_q^t(m)(M_{q-1}^t(m))^2} + \frac{M_q^t(m)}{(M_{q-1}^t(m))^2}(M_{q-1}^s(m) - M_q^s(m)) \\ &\lesssim \frac{|m|^{-2s}2^{-4tq}}{|m|^{-6t}} + \frac{|m|^{-2t}}{|m|^{-4t}}2^{-2sq} \\ &= |m|^{6t-2s}2^{-4tq} + |m|^{2t}2^{-2sq} \\ &\lesssim 2^{-2tq}|m|^{4t-2s} + 2^{-q(2s-2t)}, \end{aligned}$$

where in the last line we used the relation $|m| \lesssim 2^q$. For $m = 0$, based on the above calculation, we can get $A_{k,m} \lesssim 2^{-q(2s-2t)}$. Thus, generally, we can write $A_{k,m} \lesssim 2^{-2tq}|m|^{4t-2s} + 2^{-q(2s-2t)}$ for $m \in B_{q-1}^d$ by using the notational convention above.

Combining these estimates, we arrive at

$$\begin{aligned}
&\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2\\
&\lesssim \sum_{k\in\mathbb{Z}_2^d}\sum_{m\in(B_{q-1}^d+2^{q-1}k)} A_{k,m}\xi_{k,m}^2\\
&\lesssim \sum_{k\in\mathbb{Z}_2^d}\sum_{m\in(B_{q-1}^d+2^{q-1}k)} (2^{-q(2s-2t)}+2^{-2tq}|m-2^{q-1}k|^{4t-2s})\xi_{k,m}^2\\
&= \sum_{k\in\mathbb{Z}_2^d}\sum_{m\in B_{q-1}^d} (2^{-q(2s-2t)}+2^{-2tq}|m|^{4t-2s})\xi_{k,m+2^{q-1}k}^2\,.
\end{aligned}$$

After a change of notation, we get the desired estimate. □

C.1.9 Proof of Theorem 6.2.23

Proof. Recall

$$s^{\mathrm{KF}}(q)=\operatorname*{arg\,min}_{t\in[d/2+\delta,1/\delta]} \mathsf{L}^{\mathrm{KF}}(t,q):=\frac{\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2}{\|u(\cdot,t,q)\|_t^2}\,.$$

We analyze the denominator and numerator separately. We start with the numerator. Let

$$V_1(t,q)=\frac{1}{q}2^{q(s-d/2)}\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2\,.$$

Case 1: $t=\frac{s-d/2}{2}$. We derive an upper bound on V_1. By Proposition 6.2.22,

$$\|u(\cdot,t,q)-u(\cdot,t,q-1)\|_t^2\lesssim \sum_{k\in\mathbb{Z}_2^d}\sum_{m\in B_{q-1}^d}(2^{-q(2s-2t)}+2^{-2tq}|m|^{4t-2s})\xi_{k,m}^2\,.$$

Take $t=\frac{s-d/2}{2}$. For each $k\in\mathbb{Z}_2^d$, consider the term

$$\begin{aligned}
V_1^k(t,q)&=\frac{1}{q}2^{q(s-d/2)}\sum_{m\in B_{q-1}^d}(2^{-q(2s-2t)}+2^{-2tq}|m|^{4t-2s})\xi_{k,m}^2\\
&=\frac{1}{q}2^{q(s-d/2)}\sum_{m\in B_{q-1}^d}(2^{-q(s+d/2)}+2^{-q(s-d/2)}|m|^{-d})\xi_{k,m}^2\\
&=\frac{1}{q}\sum_{m\in B_{q-1}^d}(2^{-qd}+|m|^{-d})\xi_{k,m}^2\\
&\lesssim\frac{1}{q}\sum_{m\in B_{q-1}^d}|m|^{-d}\xi_{k,m}^2\,.
\end{aligned}$$

By Lemma 6.2.17, $\lim_{q\to\infty}\frac{1}{q}\sum_{m\in B_{q-1}^d}|m|^{-d}\xi_{k,m}^2=\gamma(0)\in(0,\infty)$. Thus, $V_1^k(t,q)$ remains bounded for $q\in\mathbb{N}$. Since $V_1(t,q)=\sum_{k\in\mathbb{Z}_2^d}V_1^k(t,q)$, it follows that $V_1(t,q)$

remains bounded for $q \in \mathbb{N}$, in the case $t = \frac{s-d/2}{2}$.

Case 2: $1/\delta \geq t \geq \frac{s-d/2}{2} + \epsilon$. We provide a lower bound of V_1 here. Using Proposition 6.2.21, we get

$$\begin{aligned}
V_1(t,q) &\gtrsim \frac{1}{q} 2^{q(s-d/2)} \sum_{m \in B_{q-1}^d \setminus \{0\}} 2^{-2tq} |m|^{4t-2s} \xi_m^2 \\
&= \frac{1}{q} 2^{q(s-d/2-2t)} \sum_{m \in B_{q-1}^d \setminus \{0\}} |m|^{4t-2s} \xi_m^2 \\
&= \frac{1}{q} 2^{q(s-d/2-2t)} 2^{(q-1)(4t-2s+d)} \cdot \left(2^{-(q-1)(4t-2s+d)} \sum_{m \in B_{q-1}^d \setminus \{0\}} |m|^{4t-2s} \xi_m^2 \right) \\
&= \frac{1}{q} 2^{(q/2-1)(4t-2s+d)} \alpha(4t-2s+d, q-1) \,.
\end{aligned}$$

By Lemma 6.2.17, $\lim_{q\to\infty} \alpha(4t-2s+d, q-1) = \gamma(4t-2s+d) > 0$ uniformly for $1/\delta \geq t \geq \frac{s-d/2}{2} + \epsilon$. Since $\lim_{q\to\infty} \frac{1}{q} 2^{(q/2-1)(4t-2s+d)} = \infty$, we get $\lim_{q\to\infty} V_1(t,q) = \infty$ and its growth rate is $\gtrsim \frac{1}{q} 2^{(q/2-1)(4t-2s+d)}$.

Case 3: $\frac{s-d/2}{2} - \epsilon \geq t \geq d/2 + \delta$. We provide a lower bound on V_1 here. Similarly to our analysis in Case 2, we have

$$\begin{aligned}
V_1(t,q) &\gtrsim \frac{1}{q} 2^{q(s-d/2-2t)} \sum_{m \in B_{q-1}^d \setminus \{0\}} |m|^{4t-2s} \xi_m^2 \\
&\gtrsim \frac{1}{q} 2^{q(s-d/2-2t)} \xi_1^2 \,.
\end{aligned}$$

Then, it holds that

$$\mathbb{P}\left(\frac{1}{q} 2^{q(s-d/2-2t)} \xi_1^2 \geq 2^{q(s-d/2-2t)/2}\right) = \mathbb{P}\left(\xi_1^2 \geq q 2^{-q(s-d/2-2t)/2}\right) \to 1$$

as $q \to \infty$. Thus, we get $\lim_{q\to\infty} V_1(t,q) = \infty$ uniformly for this range of t and the growth rate is $\gtrsim 2^{q(s-d/2-2t)/2}$. We have finished the analysis of the numerator. Now we proceed to analyze the denominator, which comprises the norm term. From Proposition 6.2.16, we have

$$\|u(\cdot,t,q)\|_t^2 \simeq 2^{-q(2s-2t)} \xi_0^2 + \sum_{m \in B_q^d \setminus \{0\}} |m|^{2t-2s} \xi_m^2 \,, \tag{C.1.8}$$

where $\{\xi_m\}_{m \in B_q^d}$ are independent unit scalar Gaussian random variables. Recall that our final target in this theorem is to show that, for any $\epsilon > 0$,

$$\lim_{q\to\infty} \mathbb{P}[s^{\mathrm{KF}}(q) \in (\frac{s-d/2}{2} - \epsilon, \frac{s-d/2}{2} + \epsilon)] = 1 \,.$$

Let $I_\epsilon = [d/2+\delta, 1/\delta]/[\frac{s-d/2}{2}-\epsilon, \frac{s-d/2}{2}+\epsilon]$. By rewriting the loss function, it suffices to show

$$\lim_{q\to\infty} \mathbb{P}[\frac{V_1(\frac{s-d/2}{2}, q)}{\|u(\cdot, \frac{s-d/2}{2}, q)\|^2_{\frac{s-d/2}{2}}} \geq \inf_{t\in I_\epsilon} \frac{V_1(t,q)}{\|u(\cdot,t,q)\|_t^2}] = 0\,.$$

Let us write

$$r(t,q) = \frac{V_1(t,q)}{V_1(\frac{s-d/2}{2}, q)} \cdot \frac{\|u(\cdot, \frac{s-d/2}{2}, q)\|^2_{\frac{s-d/2}{2}}}{\|u(\cdot,t,q)\|_t^2}\,, \tag{C.1.9}$$

then all we need is to show

$$\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon} r(t,q) \leq 1] = 0\,.$$

For $t \in I_\epsilon^1 = [d/2+\delta, \frac{s-d/2}{2}-\epsilon]$, according to the analysis for the numerator, we have that for some constant C independent of q,

$$\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon^1} \frac{V_1(t,q)}{2^{q(s-d/2-2t)/2}} \geq C] = 1\,, \tag{C.1.10}$$

and also, $V_1(\frac{s-d/2}{2}, q)$ remains uniformly bounded for $q \in \mathbb{N}$. Furthermore, the equation (C.1.8) implies the following relation:

$$\inf_{t\in I_\epsilon^1} \frac{\|u(\cdot, \frac{s-d/2}{2}, q)\|^2_{\frac{s-d/2}{2}}}{\|u(\cdot,t,q)\|_t^2} \gtrsim 1\,, \tag{C.1.11}$$

due to the inequality $t \leq \frac{s-d/2}{2} - \epsilon$. Combining the above two estimates in (C.1.10)(C.1.11), and recalling the expression for $r(t,q)$ in (C.1.9), we get

$$\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon^1} r(t,q) \leq 1] = 0\,. \tag{C.1.12}$$

Then, let $I_\epsilon^2 = [\frac{s-d/2}{2}+\epsilon, 1/\delta]$. We also need to show $\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon^2} r(t,q) \leq 1] = 0$, or equivalently,

$$\lim_{q\to\infty} \mathbb{P}[\frac{\|u(\cdot, \frac{s-d/2}{2}, q)\|^2_{\frac{s-d/2}{2}}}{V_1(\frac{s-d/2}{2}, q)} \leq \sup_{t\in I_\epsilon^2} \frac{\|u(\cdot,t,q)\|_t^2}{V_1(t,q)}] = 0\,.$$

Since $V_1(\frac{s-d/2}{2}, q)$ remains bounded according to the result in the above Case 1, it suffices to show

$$\lim_{q\to\infty} \sup_{t\in I_\epsilon^2} \frac{\|u(\cdot,t,q)\|_t^2}{V_1(t,q)} = 0$$

in probability. Using the estimate of $V_1(t,q)$ in Case 2 that $V_1(t,q) \gtrsim \frac{1}{q}2^{(q/2-1)(4t-2s+d)}$, it suffices to show

$$\lim_{q\to\infty} \sup_{t\in I_\epsilon^2} q2^{-(q/2-1)(4t-2s+d)} \|u(\cdot,t,q)\|_t^2 = 0\,.$$

To achieve this, we recall the expression of the norm term and write

$$\begin{aligned} &q2^{-(q/2-1)(4t-2s+d)} \|u(\cdot,t,q)\|_t^2 \\ \simeq &q2^{-q(s+d)+4t-2s+d}\xi_0^2 + q2^{-(q/2-1)(4t-2s+d)} \sum_{m\in B_q^d\setminus\{0\}} |m|^{2t-2s}\xi_m^2 \end{aligned}$$

Clearly, the first term on the right hand side converges to 0, so we only need to deal with the second term. Let

$$\beta(t,q) = q2^{-(q/2-1)(4t-2s+d)} \sum_{m\in B_q^d\setminus\{0\}} |m|^{2t-2s}\xi_m^2\,.$$

Consider $t \in [s-d/2+\epsilon', 1/\delta]$ where ϵ' is a parameter to be tuned. We have $2t-2s+d \geq \epsilon' > 0$ so we are able to write

$$\begin{aligned} \beta(t,q) &= q2^{-(q/2-1)(4t-2s+d)}2^{q(2t-2s+d)}\alpha(2t-2s+d,q) \\ &= q2^{-q(s-d/2)+4t-2s+d}\alpha(2t-2s+d,q)\,. \end{aligned}$$

By Lemma 6.2.17, $\lim_{q\to\infty}\alpha(2t-2s+d,q) = \gamma(2t-2s+d)$ in probability uniformly for $t \in [s-d/2+\epsilon', 1/\delta]$. Since $\lim_{q\to\infty} q2^{-(q/2-1)(4t-2s+d)}2^{q(2t-2s+d)} = 0$, we get $\lim_{q\to\infty}\sup_{t\in[s-d/2+\epsilon',1/\delta]}\beta(t,q) = 0$.
For $t \in [\frac{s-d/2}{2}+\epsilon, s-d/2+\epsilon']$, we have the estimate

$$q2^{-(q/2-1)(4t-2s+d)} \leq \left(q2^{-(q/2-1)(4t-2s+d)}\right)_{t=\frac{s-d/2}{2}+\epsilon} = q2^{-2q\epsilon+4\epsilon}\,,$$

and

$$\sum_{m\in B_q^d\setminus\{0\}} |m|^{2t-2s}\xi_m^2 \leq \sum_{m\in B_q^d\setminus\{0\}} |m|^{-d+2\epsilon'}\xi_m^2$$

where we have used the fact that t is upper bounded by $s-d/2+\epsilon'$. Hence,

$$\begin{aligned} \sup_{t\in[\frac{s-d/2}{2}+\epsilon, s-d/2+\epsilon']} \beta(t,q) &\leq q2^{-2q\epsilon+4\epsilon} \sum_{m\in B_q^d\setminus\{0\}} |m|^{-d+2\epsilon'}\xi_m^2 \\ &= q2^{-2q\epsilon+4\epsilon}2^{2q\epsilon'}\alpha(2\epsilon',q)\,. \end{aligned}$$

Now, we set $\epsilon' = \epsilon/2$ such that $\lim_{q\to\infty} q2^{-2q\epsilon+4\epsilon}2^{2q\epsilon'} = 0$. Lemma 6.2.17 leads to $\lim_{q\to\infty}\alpha(2\epsilon',q) = \gamma(2\epsilon') < \infty$, from which we can conclude $\lim_{q\to\infty}\sup_{t\in I_\epsilon^2}\beta(t,q) = 0$. Therefore, we get

$$\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon^2} r(t,q) \leq 1] = 0\,. \tag{C.1.13}$$

Combining (C.1.12) and (C.1.13) gives

$$\lim_{q\to\infty} \mathbb{P}[\inf_{t\in I_\epsilon} r(t,q) \leq 1] = 0\,. \tag{C.1.14}$$

Based on the definition of $r(t,q)$ in (C.1.9) and the arguments therein, we obtain

$$\lim_{q\to\infty} \mathbb{P}[s^{\mathrm{KF}}(q) \in (\frac{s-d/2}{2} - \epsilon, \frac{s-d/2}{2} + \epsilon)] = 1\,,$$

from which the consistency of the KF estimator follows. □