Multiscale Numerical Methods for Elliptic and Wave-Type PDEs and Their Inverse Problems

by
Shi Chen

A dissertation submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy
(Mathematics)

at the
UNIVERSITY OF WISCONSIN-MADISON
2024

Date of Final Oral Exam: 5/31/2024
The dissertation is approved by the following members of the Final Oral Committee:
Qin Li Stechmann, Associate Professor, Mathematics
Stephen J. Wright, Professor, Computer Science
Christopher Rycroft, Professor, Mathematics
Leonardo Andres Zepeda-Núñez, Assistant Professor, Mathematics

Multiscale Numerical Methods for Elliptic and Wave-Type PDEs and Their Inverse Problems

Shi Chen

Abstract

Partial differential equations (PDE) that arise from physics are usually multiscale in nature. These PDEs include small parameters that differ significantly from the scales at which the equations are typically considered, making it difficult to solve. Traditionally, solving such multiscale systems requires a careful integration of analytical insights into numerical solvers. However, modern multiscale PDEs are usually nonlinear, complex and high-dimensional, making analytical characterization challenging and often leading to the failure of classical numerical solvers. With the increasing ability to collect and process massive volumes of data, a pertinent question arises: can data be leveraged to solve these multiscale problems? This dissertation aims to explore this possibility for elliptic and wave-type equations and their inverse problems. For nonlinear elliptic equations, we investigate how data can be used to enhance the efficiency of multiscale PDE solvers. Specifically, we propose a domain decomposition framework that makes use of the compressibility of solution manifolds and incorporates two strategies from data science: neural networks and manifold learning. We demonstrate the effectiveness of our numerical method across various nonlinear elliptic PDEs. For wave-type equations, we explore how to select appropriate measured data for solving multiscale inverse problems. We propose two formulations of inverse scattering problems with new data collection processes in both time-dependent and time-independent settings, inheriting the well-posedness of the Liouville inverse scattering problem in the high-frequency limit.

Acknowledgements

Upon completing my dissertation, I can't help but remember the morning when I received my first PhD admission offer. It brought me to Madison and began my journey as a foreigner leaving my home country for the first time. Six years ago, I never imagined I would receive so much help from so many wonderful people, and now I want to thank them all.

First of all, I want to express my deepest gratitude to my advisor, Professor Qin Li, for her always inspiring and enthusiastic discussions on mathematics, her great patience in guiding my research, her tremendous support, both professionally and morally, that I never asked for, and her constant belief in my potential. I am forever indebted to her, wherever I may be, and I hope I have at least met her lowest expectations at this point.

I am very fortunate to have had Prof. Stephen J. Wright, Prof. Leonardo Zepeda-Núñez, Prof. Song Gao, Prof. Oliver Tse, Prof. Jianfeng Lu and Prof. Xu Yang as my collaborators and for the mentorship and immense professional support they have provided me. I also feel deeply honored to have Prof. Christopher Rycroft as a member of my thesis committee for his invaluable feedback on my dissertation. I am grateful to Prof. Weiran Sun, Prof. Xiuyuan Cheng, and Prof. Yuehaw Khoo for the stimulating discussions and their great professional supports.

My journey at UW-Madison would not have been complete without my dear colleagues. I would like to thank my collaborators Dr. Zhiyan Ding, Dr. Ke Chen, Sixu Li, and Yewei Xu for their dedication, hard work, and fresh perspectives that have significantly contributed to the progress of our research. I also thank all the members of the kinetic theory group, especially Dr. Ruhui Jin, Dr. Anjali Nair, Dr. Yukun Yue, Borong Zhang, Martín Guerra and Peiyi Chen, for their invaluable enthusiasm.

I would like to thank the University of Wisconsin-Madison and the Institute for Foundations of Data Science (IFDS) for providing me with financial support during my PhD pursuit.

Lastly, my sincerest thanks go to my parents. I couldn't have overcome the obstacles I faced without their love and support. I would also like to thank my friends in graduate school, especially Dr. Junguang Yu, for the support, encouragement and companionship.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Multiscale modeling and simulation is a body of theory and methods to solve problems that have important features across multiple temporal and spatial scales. The latest surge of multiscale modelling traces back to the mid-1980s when US national labs started to reduce nuclear underground tests, and the idea of simulation-based design and analysis concepts were birthed [125]. Since then, a wealth of multiscale numerical methods has been developed, commonly referred to as "numerical homogenization" [15, 128, 90] or "asymptotic preserving" [84, 106, 138] methods. In the design of such multiscale solvers, it is crucial to fuse the analytical understanding of the governing equations into suitable computational methods [8]. However, modern multiscale systems are often nonlinear, complex, and high-dimensional, making their analytical properties difficult to characterize and classical numerical methods prone to failure. Designing multiscale numerical methods tailored to such systems remains an intractable task.

In recent years, the fast development of techniques in data science has provided completely new perspective and methodologies in modern computational methods. With the substantial increase in computing power and the ability to collect and process massive volumes of data, whether synthetic or experimental, the integration of data to support computation has become fundamental. Among the myriad of methods driving this trend, Physics Informed Neural Networks (PINN) have emerged as particularly popular in simu-

lating physical systems [188, 85]. While PINNs are versatile enough to solve any PDE using data, effectively integrating analytical insights is challenging and their advantage over classical numerical methods is still elusive [188]. Since different types of PDEs possess unique analytical properties, efficient data-driven methods should consider these distinctions and be specifically tailored to each type.

In this dissertation, we explore data-driven numerical methods for elliptic and wave-type PDEs and their inverse problems. Our research are guided by two primary questions:

- *Given data of solutions, how can we design efficient solvers for multiscale PDEs?*

Multiscale PDEs usually contain a small scale parameter that makes the PDEs rather stiff: to ensure accuracy and stability, classical methods require the mesh size to resolve the small parameter, resulting in large degrees of freedom as the parameter approaches zero. Without prior information about the PDE, this complexity is inevitable [19, 18]. However, with data of solutions, this issue can be mitigated. By using a set of PDE solutions, we can judiciously compress the data into a set of basis that effectively represent the solution space. This approach has been proved successful for linear elliptic PDEs [52, 210, 65]. However, applying the same approach to nonlinear PDEs—which do not form a linear space—might not be straightforward. Here we specifically focus on elliptic multiscale PDEs, and introduce two different new strategies to compress the data, inspired by techniques from data science. In both approaches, we adopt a domain decomposition framework with Schwarz iteration. In the first approach, we propose a multiscale solver that use two-layer neural networks to approximate the nonlinear low dimensional solution map for the subdomains [68]. In the second approach, inspired by manifold learning techniques, we exploit the tangent spaces spanned by the nearest neighbors to compress local solution manifolds [71]. Both strategies demonstrate significant improvement in efficiency and good accuracy, and can be applied to a wide class of PDEs. Specifically, we demonstrate that our manifold learning approach can be applied to a nonlinear radiative transfer equation in the diffusion regime.

- *Given a multiscale PDE, how can we choose the appropriate data for solving inverse problems?*

Partial differential equations that arise from physics usually include a variety of parameters that cannot be directly measured from experiments. The strategies for estimating such unknown or partially known parameters from measured data fall under the category of inverse problems. We are interested in inverse problems whose governing equation is a multiscale PDE. While connecting multiscale PDEs to their asymptotic limit is a classical topic in traditional applied mathematics, their counterpart for inverse problems is still an area ripe for exploration. For radiative transport equation, connecting their inverse problem to its asymptotic limit has been previous studied in [60, 153]. Here we study the problems associated to wave-type PDEs. We study two inverse problems: a time-dependent inverse scattering problem [66], and a time-independent inverse scattering problem [69]. The key reason of failure in connecting classical inverse problems lies at the disparity of data. This could lead to stability issue in the reconstruction of unknown parameters, making the inverse scattering problem difficult to solve in the high frequency limit. By choosing the right data, we can design new inverse scattering problems that are asymptotically stable in the small parameter regime. This fact stands in contrast with the unstable reconstruction for classical inverse scattering problems.

The rest of this dissertation is organized as follows: Chapter 2 provides an overview of analytical properties of the elliptic and wave-type PDEs. We first review the homogenization limits of elliptic PDEs that serve as a motivation of many numerical homogenization methods. We then describe the domain decomposition framework and Schwarz iteration that will be used in the subsequent chapters. For wave-type PDEs, we introduce the Wigner transform techniques and showcase their application in deriving the classical limit of the Schrödinger equation. Chapter 3 and Chapter 4 describe our strategies for solving elliptic multiscale PDEs: a method based on two-layer neural networks and a method inspired by manifold learning. These strategies are applicable to a broader class of PDEs, and we demonstrate their use in solving a nonlinear radiative transfer equation in Chap-

ter 4. Next, Chapter 5 and Chapter 6 use the Wigner transform to connect the inverse scattering problem to its asymptotic limit in the time-dependent and time-independent setting, respectively. In the time-dependent setting, we explore linearized problems across three different scenarios. In the time-independent setting, we introduce an asymptotically stable Helmholtz inverse scattering problem by selecting data that aligns with its asymptotic counterpart, the Liouville inverse scattering problem. A conclusion is provided in Chapter 7.

Chapter 2

Homogenization of Elliptic and Wave-Type PDEs

In this chapter, we consider partial differential equations that involves multiple scales. Typically, these PDEs include small parameters that crucially influence solution behaviors. As the small parameters approach zero, the multiscale PDEs converge to different asymptotic limits that do not have scale separations. Deriving such asymptotic limiting equations as accurate surrogates of the original equations is termed homogenization. Depending on the type of PDE, diverse homogenization techniques are invented across different parameter regimes. In this chapter, we review these homogenization techniques for elliptic and wave-type PDEs, which will be useful in motivating our data-driven multiscale numerical methods. This chapter is organized as follows. In Section 2.1, we describe nonlinear multiscale elliptic PDEs and discussed the homogenization limit of elliptic equations with highly oscillatory media. Next, in Section 2.2, we outline the domain decomposition framework and the Schwarz iteration strategy. This framework, which is widely used in solving elliptic PDEs, will be further integrated with data-driven approaches in subsequent chapters. Then in Section 2.3, we introduce the Wigner transform, a technique that is used in deriving the classical limit of the Schrödinger equation.

2.1 Homogenization limit of nonlinear elliptic equations

Consider the following general class of nonlinear elliptic PDEs with Dirichlet boundary conditions:

$$\begin{cases} F^\varepsilon\left(D^2u^\varepsilon(x), Du^\varepsilon(x), u^\varepsilon(x), x\right) = 0, & x \in \Omega\,, \\ u^\varepsilon(x) = \phi(x), & x \in \partial\Omega\,, \end{cases} \tag{2.1}$$

where $\Omega \subset \mathbb{R}^d$ is a domain in d-dimensional space, $\varepsilon > 0$ represents the small scale, and $F^\varepsilon : S^{d\times d}\times\mathbb{R}^d\times\mathbb{R}\times\Omega \to \mathbb{R}$ (where $S^{d\times d}$ denotes the space of real symmetric $d\times d$ matrices) is a smooth function. To ensure ellipticity, we require for all $(R, p, u, x) \in S^{d\times d}\times\mathbb{R}^d\times\mathbb{R}\times\Omega$ that

$$F^\varepsilon(R+Q, p, u, x) \le F^\varepsilon(R, p, u, x)\,,$$

for all nonnegative semidefinite $Q \in S^{d\times d}$. We assume that for an appropriately chosen boundary condition ϕ, the PDE (2.1) has a unique (viscosity) solution $u^\varepsilon \in C(\overline{\Omega})$. For details on the theory of fully nonlinear elliptic equations, see for example, [54, 136].

This class of problems has fundamental importance in modern science and engineering, in such areas as synthesis of composite materials, discovery of geological structures, and design of aerospace structures. The primary computational challenges behind all these problems lie in the complicated interplay between the nonlinearity and the extremely high number of degrees of freedom necessitated by the smallest scale.

To achieve a desired level of numerical error, classical numerical methods require refined discretization strategies with a mesh width $\Delta x = o(\varepsilon)$, making the leading to at least $O(\varepsilon^{-d})$ degrees of freedom in the discretized problem. The resulting numerical cost is prohibitive when ε is small. The homogenization limit of (2.1) as $\varepsilon \to 0$ can be specified under additional assumptions, such as when the medium is pseudo-periodic. Let

$$F^\varepsilon(R, p, u, x) = F\left(R, p, u, x, \frac{x}{\varepsilon}\right) \tag{2.2}$$

for some $F : S^{d\times d}\times\mathbb{R}^d\times\mathbb{R}\times\Omega\times\mathbb{R}^d \to \mathbb{R}$ that is periodic in the last argument with period

Y. We have the following theorem.

Theorem 2.1 ([101], Theorem 3.3)**.** *Suppose that the nonlinear function F^ε is uniform elliptic and $u \mapsto F^\varepsilon(\cdot,\cdot,u,\cdot)$ is nondecreasing. Let F^ε be pseudo-periodic as defined in* (2.2)*. The solution $u^\varepsilon \in C(\overline{\Omega})$ to* (2.1) *converges uniformly over $\overline{\Omega}$ as $\varepsilon \to 0$ to the unique viscosity solution u^* of the following equation*

$$\begin{cases} \bar{F}(D^2u^*(x), Du^*(x), u^*(x), x) = 0, & x \in \Omega\,, \\ u^*(x) = \phi(x), & x \in \partial\Omega\,, \end{cases} \tag{2.3}$$

where the homogenized nonlinear function $\bar{F}(R,p,u,x)$ is defined as follows: For a fixed set of $(R,p,u,x) \in S^{d\times d} \times \mathbb{R}^d \times \mathbb{R} \times \Omega$, there exists a unique real number λ for which the following cell problem has a unique viscosity solution $v \in C^{1,\gamma}(\mathbb{R}^d)$ for some $\gamma > 0$:

$$\begin{cases} F(D_y^2 v(y) + R, p, u, x, y) = \lambda, & y \in \mathbb{R}^d\,, \\ v(y+Y) = v(y), & y \in \mathbb{R}^d\,, \end{cases} \tag{2.4}$$

(where Y is the period in the last argument of F). We set $\bar{F}(R,p,u,x) = \lambda$.

This result can be viewed as the extension of a linear homogenization result [8]. Although the medium is highly oscillatory for small ε, the solution u^ε approaches that of a certain limiting equation with a one-scale structure, as $\varepsilon \to 0$. In practice, the form of the limit $\bar{F}$ is typically unknown, but this observation has led to an exploration of numerical homogenization algorithms, in which one seeks to capture the limit numerically without resolving the fine scale ε. We view this problem as one of manifold reduction. The solution u^ε can be "compressed" significantly; its "information" is stored mostly in u^*, which can be computed from (2.4) using mesh width $\Delta x = O(1)$, in contrast to the $\Delta x = o(\varepsilon)$ required to solve (2.1). In other words, the $O(\varepsilon^{-d})$-dimensional solution manifold can potentially be compressed into an $O(1)$-dimensional solution manifold, up to small homogenization error that vanishes as $\varepsilon \to 0$.

The literature for numerical homogenization is rich, particularly for the linear setting.

Relevant approaches include the multiscale finite element method (MsFEM) [128, 98, 129], the heterogeneous multiscale method (HMM) [90, 4, 93], the generalized finite element method [17, 16], upscaling based on harmonic coordinates [181], elliptic solvers based on $\mathcal{H}$-matrices [40, 119], the reduced basis method [3, 2], the use of localization [171], and the methods based on random SVD [65, 64, 63], to name a few. The analytical understanding of the homogenized equation is essential in the construction of these methods [8]. When randomness presents, one can also look for low dimensional representation of the solutions in the random space [78, 130, 127, 157].

The literature for nonlinear problems is not as rich. There are several works on quasi-linear problems, all of which can be seen as extensions of classical methods, including the MsFEM [97, 73, 96], the HMM [93, 5], the generalized finite element method [99], the local orthogonal decomposition method [122], the reduced basis method [3] and nonlocal multicontinua upscaling [79]. These solvers must be designed carefully for specific nonlinear equations. By contrast, our method makes use of the low-rankness of the solution sets and could be applied with minor modification to different equations.

2.2 Domain Decomposition and the Schwarz method for multiscale elliptic PDEs

A popular framework for solving partial differential equations is domain decomposition, where the problem is decomposed and solved separately in different subdomains, with boundary conditions chosen iteratively to ensure regularity of the solution across the full domain. This approach is naturally parallelizable, with potential savings in memory and computational cost. It essentially translates the inversion of a large matrix into the composition of inversions of many smaller matrices. The many variants of domain decomposition include the Schwarz iteration strategy that we adopt in this dissertation. This strategy makes use of a partition-of-unity function that resolves the mismatch between two solutions in adjacent subdomains. We briefly review the method here.

For simplicity we describe the case of $d = 2$ and assume throughout the chapter

that $\Omega = [0, L]^2$ for some $L > 0$. The approach partitions the domain Ω into multiple overlapping subdomains, also called *patches*. It starts with an initial guess of the solution on the boundaries of all subdomains, and solves the Dirichlet problem on each patch. The computed solutions then serve as the boundary conditions for neighboring patches, for purposes of computing the next iteration. The entire process is repeated until convergence.

In the current setting, the overlapping rectangular patches are defined as follows:

$$\Omega = \bigcup_{m \in J} \Omega_m, \quad \text{with} \quad \Omega_m = (x_{m_1}^{(1)}, x_{m_1}^{(2)}) \times (y_{m_2}^{(1)}, y_{m_2}^{(2)}), \tag{2.5}$$

where $m = (m_1, m_2)$ is a multi-index and J is the collection of the indices

$$J = \{m = (m_1, m_2) : \, m_1 = 1, \ldots, M_1, \, m_2 = 1, \ldots, M_2\}.$$

We plot the setup in Figure 2.1. For each patch we define the associated partition-of-unity function χ_m, which has $\chi_m(x) \geq 0$ and

$$\chi_m(x) = 0 \quad \text{on } x \in \Omega \setminus \Omega_m, \quad \sum_m \chi_m(x) = 1, \quad \forall x \in \Omega. \tag{2.6}$$

We set $\partial\Omega_m$ to be the boundary of patch Ω_m and denote by $\mathscr{N}(m)$ the collection of indices of the neighbors of Ω_m. In this 2D case, we have

$$\mathscr{N}(m) = \{(m_1 \pm 1, m_2)\} \cup \{(m_1, m_2 \pm 1)\} \subset J. \tag{2.7}$$

Naturally, indices that are out of range, which correspond to patches adjacent to the boundary $\partial\Omega$, are omitted from $\mathscr{N}(m)$.

In the framework of domain decomposition, the full-domain problem is decomposed into multiple smaller problems supported on the subdomains. Define the local Dirichlet

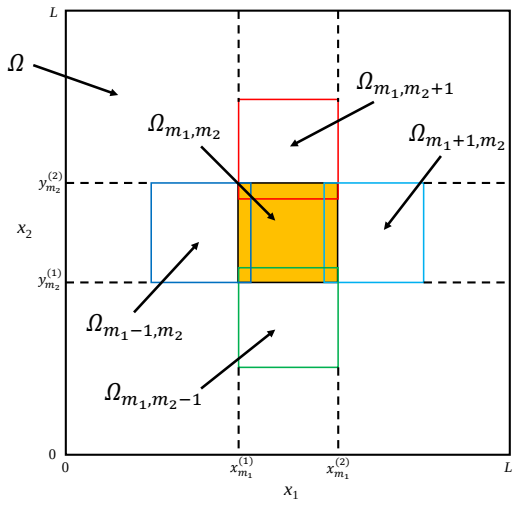


Figure 2.1: Domain decomposition for a square 2D geometry. Each patch is labeled by a multi-index $m = (m_1, m_2)$. The patches adjacent to Ω_m are those on its north/south/west/east sides.

problem on patch Ω_m by:

$$\begin{cases} F^{\varepsilon}\left(D^2 u_m^{\varepsilon}(x), Du_m^{\varepsilon}(x), u_m^{\varepsilon}(x), x\right) = 0\,, & x \in \Omega_m\,, \\ u_m^{\varepsilon}(x) = \phi_m(x)\,, & x \in \partial\Omega_m\,. \end{cases} \tag{2.8}$$

For this local problem, we define the following operators:

- $\mathcal{S}_m^{\varepsilon}$ is the solution operator that maps local boundary condition ϕ_m to the local solution u_m^{ε}:

  $$u_m^{\varepsilon} = \mathcal{S}_m^{\varepsilon}\phi_m\,.$$

  Denoting by d_m the number of grid points on the boundary $\partial\Omega_m$ and D_m the number of grid points on the subdomain Ω_m, then $\mathcal{S}_m^{\varepsilon}$ maps $\mathbb{R}^{d_m}$ to $\mathbb{R}^{D_m}$.

- $\mathcal{I}_{l,m}$ denotes the restriction (or trace-taking) operator that restricts the solution within Ω_m to its part that overlaps with the boundary of Ω_l, for all $l \in \mathcal{N}(m)$. That is,

  $$\mathcal{I}_{l,m} u_m^{\varepsilon} = u_m^{\varepsilon}|_{\partial\Omega_l \cap \Omega_m}\,.$$

  Denoting by $p_{l,m}$ the number of grid points in $\partial\Omega_l \cap \Omega_m$, then $\mathcal{I}_{l,m}$ maps $\mathbb{R}^{D_m}$ to

$\mathbb{R}^{p_{l,m}}$.

- $\mathcal{Q}^{\varepsilon}_{l,m}$ is the composition of $\mathcal{S}^{\varepsilon}_{m}$ and $\mathcal{I}_{l,m}$. It is a boundary-to-boundary operator that maps the local boundary condition ϕ_m to the restricted solution $u^{\varepsilon}_m|_{\partial\Omega_l\cap\Omega_m}$:

$$\mathcal{Q}^{\varepsilon}_{l,m}\phi_m = \mathcal{I}_{l,m}\mathcal{S}^{\varepsilon}_{m}\phi_m = u^{\varepsilon}_m|_{\partial\Omega_l\cap\Omega_m}\,.$$

  $\mathcal{Q}^{\varepsilon}_{l,m}$ maps $\mathbb{R}^{d_m}$ to $\mathbb{R}^{p_{l,m}}$.

- $\mathcal{Q}^{\varepsilon}_{m}$ denotes the collection of all segments of boundary conditions $\psi_{l,m}$ that is computed from the full-domain boundary condition ϕ_m:

$$\mathcal{Q}^{\varepsilon}_{m}\phi_m = \bigoplus_{l\in\mathscr{N}(m)} \psi_{l,m} = \bigoplus_{l\in\mathscr{N}(m)} \mathcal{Q}^{\varepsilon}_{l,m}\phi_m = \bigoplus_{l\in\mathscr{N}(m)} \mathcal{I}_{l,m}\mathcal{S}^{\varepsilon}_{m}\phi_m\,. \tag{2.9}$$

  Letting $p_m = \sum_{l\in\mathscr{N}(m)} p_{l,m}$, $\mathcal{Q}^{\varepsilon}_{m}$ maps $\mathbb{R}^{d_m}$ to $\mathbb{R}^{p_m}$.

The Schwarz procedure starts by making a guess of boundary condition on each Ω_m. At the nth iteration, (4.9) is solved for each subdomains Ω_m (possibly in parallel) and these solutions are used to define new boundary conditions for the neighboring subdomains Ω_l, $l\in\mathscr{N}(m)$. The boundary conditions for Ω_m at iteration $n+1$ are thus:

$$\phi^{(n+1)}_m = \begin{cases} \psi^{(n)}_{m,l} = \mathcal{I}_{m,l}\mathcal{S}^{\varepsilon}_{l}\phi^{(n)}_{l}\,, & \text{on } \partial\Omega_m\cap\Omega_l, \quad l\in\mathscr{N}(m)\,, \\ \phi|_{\partial\Omega_m\cap\partial\Omega}\,, & \text{on } \partial\Omega_m\cap\partial\Omega\,. \end{cases} \tag{2.10}$$

Note that the physical full-domain boundary condition is imposed on the points in $\partial\Omega_m\cap\partial\Omega$. Each iteration of the Schwarz procedure can be viewed as an application of the map $\mathcal{Q}^{\varepsilon}_{m,l}$. The procedure concludes by patching up the local solutions from the subdomains. The overall algorithm is summarized in Algorithm 1.

The convergence of classical Schwarz iteration is guaranteed for fully nonlinear elliptic equations; see, for example [163, 164, 107]. Since the computation of solution $u^{\varepsilon}_m = \mathcal{S}^{\varepsilon}_{m}\phi_m$ can be expensive due to the nonlinearity and oscillation of the medium at small scale ε,

the major computational cost for Schwarz iteration comes from the repeated evaluation of the boundary-to-boundary map $\mathcal{Q}^{\varepsilon}_{m,l}$, which requires solution of an elliptic PDE on each subdomain.

Algorithm 1 The Schwarz iteration for fully nonlinear elliptic equations (2.1).

Domain Decomposition:
    Decompose Ω into overlapping patches: $\Omega = \bigcup_{m\in J} \Omega_m$.
Given tolerance δ_0 and initial guesses $\phi_m^{(0)}$ of boundary conditions on each patch $m \in J$.
Schwarz iteration:
    Set $n = 0$ and res $= 1$.
    while res $\geq \delta_0$ **do**
        For $m \in J$, compute local solutions $u_m^{(n)} = \mathcal{S}_m \phi_m^{(n)}$;
        For $m \in J$ and $l \in \mathscr{N}(m)$, restrict the solutions $\psi_{m,l}^{(n)} = \mathcal{I}_{m,l} u_m^{(n)}$;
        For $m \in J$, update $\phi_m^{(n+1)}$ by (4.15);
        Set res $= \sum_m \|\phi_m^{(n+1)} - \phi_m^{(n)}\|_{L^2(\partial\Omega_m)}$ and $n \leftarrow n+1$.
    end while
return Global solution $u^{(n)} = \sum_{m\in J} \chi_m u_m^{(n)}$.

2.3 The Classical limit of the Schrödinger equation

In this section, we review the homogenization of wave-type PDEs, with a specific focus on the Wigner transform technique. This technique seamlessly connects wave-type PDEs and their high-frequency limit, or classical limit. We illustrate its application using the Schrödinger equation as an example.

2.3.1 Schrödinger equation

In this section, we present some preliminary results that show the classical limit of the Schrödinger equation in the classical limit.

For a nonrelativisitic single particle, the time-dependent Schrödinger equation in position basis writes as:

$$\begin{aligned} \mathrm{i}\varepsilon\partial_t\phi^{\varepsilon} = -\frac{1}{2}\varepsilon^2\Delta_x\phi^{\varepsilon} + V(x)\phi^{\varepsilon}\,, \quad x\in\mathbb{R}^d\,, \quad t>0\,, \\ \phi^{\varepsilon}(0,x) = \phi_{\mathrm{I}}^{\varepsilon}(x)\,, \quad x\in\mathbb{R}^d\,. \end{aligned} \tag{2.11}$$

This is derived assuming the Hamiltonian is $H = \frac{1}{2}|k|^2 + V(x)$, a summation of kinetic and potential energies of the particles constituting the system. In the equation, ϕ^ε is the wave function, $\varepsilon > 0$ is the rescaled Planck constant, and $V(x)$ is the potential term.

Some physical quantities can be calculated using ϕ^ε. For example, the particle density ρ^ε and current density J^ε are calculated by

$$\rho^\varepsilon(t,x) = |\phi^\varepsilon(t,x)|^2 \ , \quad J^\varepsilon(t,x) = \varepsilon \mathrm{Im}\left(\overline{\phi^\varepsilon(t,x)}\nabla_x \phi^\varepsilon(t,x)\right) \ .$$

These present the probability and the probability flux of the particle found in some spatial configuration at some instant of time, according to the Copenhagen interpretation. Both quantities are quadratic functionals of $\phi^\varepsilon(t)$, and it is straightforward to derive, from (2.11), the following conservation law:

$$\partial_t \rho^\varepsilon + \nabla_x \cdot J^\varepsilon = 0 \, .$$

A more general definition of physical observables can be given using phase space symbols and Weyl quantization [124]. To make it more explicit, let $a(x,k)$ be a symbol, then using Weyl quantization, we can define a pseudo-differential operator $a^W(x, \varepsilon D_x)$ whose action on $f(x)$ leads to:

$$(a^{\mathrm{W}}(x,\varepsilon D_x)f)(x) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^{2d}} a\left(\frac{x+y}{2}, \varepsilon k\right) f(y) e^{\mathrm{i}(x-y)k} \mathrm{d}y \mathrm{d}k \, , \tag{2.12}$$

where $\varepsilon D_x = -\mathrm{i}\varepsilon\nabla_x$. We then define the expectation value of the symbol a to be a quadratic functional of wave function $\phi^\varepsilon(t)$:

$$a[\phi^\varepsilon(t)] = \left\langle \phi^\varepsilon(t), a^{\mathrm{W}}(x,\varepsilon D_x)\phi^\varepsilon(t)\right\rangle_{L^2(\mathbb{R}^d)} \ ,$$

where $\langle\cdot,\cdot\rangle_{L^2(\mathbb{R}^d)}$ denotes the inner product on $L^2(\mathbb{R}^d)$.

The well-posedness theory of Schrödinger equation (2.11) is classical. For $V = V(x)$

being continuous and bounded, i.e., $V \in C_b(\mathbb{R}^d)$, the Hamiltonian operator $\hat{H}^\varepsilon$ is

$$\hat{H}^\varepsilon \phi^\varepsilon = -\frac{\varepsilon^2}{2}\Delta_x \phi^\varepsilon(x) + V(x)\phi^\varepsilon(x)\,. \tag{2.13}$$

It maps functions in $H^2(\mathbb{R}^d) \subset L^2(\mathbb{R}^d)$ to $L^2(\mathbb{R}^d)$, and is self-adjoint. By Stone's theorem, the operator $\frac{1}{\mathrm{i}\varepsilon}\hat{H}^\varepsilon$ generates a unitary, strongly continuous semi-group on $L^2(\mathbb{R}^d)$, which guarantees a unique solution to the Schrödinger equation (2.11). Moreover, the $L^2(\mathbb{R}^d)$ inner product is conserved in time:

$$\langle \phi_1^\varepsilon(t), \phi_2^\varepsilon(t)\rangle_{L^2(\mathbb{R}^d)} = \langle \phi_1^\varepsilon(0), \phi_2^\varepsilon(0)\rangle_{L^2(\mathbb{R}^d)}\,, \quad \forall t > 0\,, \tag{2.14}$$

for $\phi_i^\varepsilon(t), i = 1, 2$ both solve the Schrödinger equation (2.11).

2.3.2 Wigner transform and the classical limit

The Wigner transform is one of many approaches used to derive (semi-)classical limit of Schrödinger equations. The technique was explored in depth in [109]. Let $\phi_1^\varepsilon(t)$ and $\phi_2^\varepsilon(t)$ solve the Schrödinger equation, and we define the corresponding Wigner transform:

$$W^\varepsilon[\phi_1^\varepsilon, \phi_2^\varepsilon](t, x, k) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} e^{\mathrm{i}ky}\phi_1^\varepsilon\left(t, x - \frac{\varepsilon}{2}y\right)\overline{\phi_2^\varepsilon}\left(t, x + \frac{\varepsilon}{2}y\right)\mathrm{d}y\,. \tag{2.15}$$

Here $\overline{\phi^\varepsilon}$ is the complex conjugate of ϕ^ε. This definition is essentially the Fourier transform of the density matrix

$$\left\langle x - \frac{\varepsilon}{2}y\middle|\phi_1^\varepsilon\right\rangle\left\langle \phi_2^\varepsilon\middle| x + \frac{\varepsilon}{2}y\right\rangle = \phi_1^\varepsilon\left(t, x - \frac{\varepsilon}{2}y\right)\overline{\phi_2^\varepsilon}\left(t, x + \frac{\varepsilon}{2}y\right)$$

in the y variable.

We furthermore abbreviate $W^\varepsilon[\phi^\varepsilon, \phi^\varepsilon]$ to be $W^\varepsilon[\phi^\varepsilon]$. It is then straightforward to show that $W^\varepsilon[\phi^\varepsilon]$ is real-valued.

Note that the Wigner transform loses the phase information: Changing $\phi^\varepsilon(t)$ to $\phi^\varepsilon(t)e^{\mathrm{i}S(t)}$, one obtains the same corresponding Wigner function. Moreover, it is not

guaranteed that $W^\varepsilon[\phi^\varepsilon]$ is positive, and thus it does not serve directly as the particle density on the phase space. However, the quantum expectation of physical observables can be easily recovered using the Wigner function. Using the symbol defined in (2.12), it can be shown [124] that

$$a[\phi^\varepsilon(t)] = \left\langle \phi^\varepsilon(t), a^{\mathrm{W}}(x, \varepsilon D_x)\phi^\varepsilon(t)\right\rangle_{L^2(\mathbb{R}^d)} = \int_{\mathbb{R}^{2d}} a(x,k) W^\varepsilon[\phi^\varepsilon(t)]\mathrm{d}x\mathrm{d}k\,.$$

In particular, the first and second moments in k of $W^\varepsilon[\phi^\varepsilon]$ exactly recover the particle density $\rho^\varepsilon(t)$ and the current density $J^\varepsilon(t)$:

$$\rho^\varepsilon(t,x) = \int_{\mathbb{R}^d} W^\varepsilon[\phi^\varepsilon(t)](t,x,k)\mathrm{d}k\,, \quad J^\varepsilon(t,x) = \int_{\mathbb{R}^d} kW^\varepsilon[\phi^\varepsilon(t)](t,x,k)\mathrm{d}k\,.$$

We now derive the equation for $W^\varepsilon[\phi_1^\varepsilon, \phi_2^\varepsilon]$, as summarized in the following lemma.

Lemma 2.1. *Let $\phi_1^\varepsilon(t)$ and $\phi_2^\varepsilon(t)$ solve the Schrödinger equation* (2.11)*, and define*

$$f^\varepsilon(t,x,k) = W^\varepsilon[\phi_1^\varepsilon, \phi_2^\varepsilon](t,x,k)\,.$$

Then f^ε satisfies the following Wigner equation:

$$\begin{aligned} \partial_t f^\varepsilon + k\cdot\nabla_x f^\varepsilon = \mathcal{L}_V^\varepsilon[f^\varepsilon]\,, \quad (x,k)\in\mathbb{R}^{2d}\,, \quad t>0\,,\\ f^\varepsilon(0,x,k) = f_{\mathrm{I}}^\varepsilon(x,k)\,, \end{aligned} \tag{2.16}$$

with $f_{\mathrm{I}}^\varepsilon(x,k)$ being the Wigner transform of initial conditions $\phi_1^\varepsilon(0)$ and $\phi_2^\varepsilon(0)$, and the operator $\mathcal{L}_V^\varepsilon$ is defined as:

$$\mathcal{L}_V^\varepsilon[f^\varepsilon] = \mathrm{i}\frac{1}{(2\pi)^d}\int_{\mathbb{R}^{2d}} \delta^\varepsilon[V](x,y) f^\varepsilon(x,p) e^{\mathrm{i}y(k-p)}\mathrm{d}y\mathrm{d}p\,. \tag{2.17}$$

Here $\delta^\varepsilon[V](x,y) = \frac{1}{\varepsilon}\left[V\left(x+\frac{1}{2}\varepsilon y\right) - V\left(x-\frac{1}{2}\varepsilon y\right)\right]$. Equivalently, one can also write

$$\mathcal{L}_V^\varepsilon[f^\varepsilon] = \mathrm{i}\int_{\mathbb{R}^{2d}} e^{\mathrm{i}p(x-y)} V(y) D^\varepsilon f^\varepsilon(x,k,p)\mathrm{d}p\mathrm{d}y\,, \tag{2.18}$$

where the term $D^\varepsilon f^\varepsilon$ is defined by

$$D^\varepsilon f^\varepsilon(x,k,p) = \frac{1}{\varepsilon}\left[f^\varepsilon\left(x, k+\frac{1}{2}\varepsilon p\right) - f^\varepsilon\left(x, k-\frac{1}{2}\varepsilon p\right)\right]. \tag{2.19}$$

We note that $\mathcal{L}_V^\varepsilon$ is an operator that is anti-self-adjoint for all real-valued potential V. To see that, we first define

$$\mathcal{V}^\varepsilon(x,k) = \mathrm{i}\frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} \delta^\varepsilon[V](x,y)e^{\mathrm{i}yk}\mathrm{d}y\,. \tag{2.20}$$

This allows us to simplify (2.17) to a convolution form

$$\mathcal{L}_V^\varepsilon[f^\varepsilon] = \int_{\mathbb{R}^{2d}} \mathcal{V}^\varepsilon(x,k-p)f(x,p)\mathrm{d}p = \mathcal{V}^\varepsilon *_k f^\varepsilon\,.$$

Since $\mathcal{V}^\varepsilon(x,-k) = -\overline{\mathcal{V}^\varepsilon}(x,k)$, it is straightforward to see

$$\langle \mathcal{V}^\varepsilon *_k f_1^\varepsilon, f_2^\varepsilon\rangle_{L^2(\mathbb{R}^{2d})} = -\langle f_1^\varepsilon, \mathcal{V}^\varepsilon *_k f_2^\varepsilon\rangle_{L^2(\mathbb{R}^{2d})}\,,$$

meaning:

$$\langle \mathcal{L}_V^\varepsilon[f_1^\varepsilon], f_2^\varepsilon\rangle_{L^2(\mathbb{R}^{2d})} = -\langle f_1^\varepsilon, \mathcal{L}_V^\varepsilon[f_2^\varepsilon]\rangle_{L^2(\mathbb{R}^{2d})}\,. \tag{2.21}$$

To derive the Wigner equation (2.16), one only needs to plug in the Schrödinger equation for both ϕ_1^ε and ϕ_2^ε. The statement of the lemma is formal, but one can make it rigorous in $L^2(\mathbb{R}^d)$. We omit the derivation from this chapter, but refer interested readers to [109].

The nice format of the Wigner equation makes it easy to obtain the classical limit. Indeed, formally, as $\varepsilon \to 0$, $\delta^\varepsilon[V] \to y\cdot\nabla_x V$. Then according to the definition of the operator (2.17), we have

$$\mathcal{L}_V^\varepsilon[f^\varepsilon] \to \nabla_x V\cdot\nabla_k f^\varepsilon + O(\varepsilon^2)\,.$$

This means the asymptotic limit of (2.16), up to the truncation of $O(\varepsilon^2)$, is the Liouville

equation:

$$\partial_t f + k \cdot \nabla_x f - \nabla_x V \cdot \nabla_k f = 0 \,. \tag{2.22}$$

Following the characteristic of this equation we have:

$$\dot{x} = k \,, \quad \dot{k} = -\nabla_x V(x) \,. \tag{2.23}$$

This is exactly the same as the Newtonian law of motion generated by the Hamiltonian $H(x,k) = \frac{1}{2}|k|^2 + V(x)$.

This formal analysis can be made rigorous. Indeed in [109] the authors studied a general Hamiltonian system and derived the asymptotic limit for the Wigner equation. Let $\mathcal{S}'(\mathbb{R}^{2d})$ denotes the space of tempered distributions. In our special case, the rigorous theorem states as follows.

Theorem 2.2. *Suppose the potential $V(x)$ satisfies*

$$V(x) \in C^\infty(\mathbb{R}^d;\mathbb{R}) : \quad |\partial_x^\alpha V(x)| \leq C_\alpha \quad \forall \alpha \in \mathbb{N}^d \,, \tag{2.24}$$

then the Wigner transform $f^\varepsilon(t,x,k)$ of $\phi^\varepsilon(t)$, the solution to Schrödinger equation (2.11)*, converges, in $L^\infty(\mathbb{R},\mathcal{S}'(\mathbb{R}^{2d}))$ in the weak-$*$ sense, locally uniformly in t to the measure $f(t,x,k)$ that solves:*

$$\partial_t f + k \cdot \nabla_x f - \nabla_x V \cdot \nabla_k f = 0 \,, \quad f(0,x,k) = f_I(x,k) \,. \tag{2.25}$$

The initial data f_I is the weak-$*$ limit of Wigner transform of $\phi_{\mathrm{I}}^\varepsilon(t)$ in $\mathcal{S}'(\mathbb{R}^d)$.

Chapter 3

A Reduced Order Schwarz Method Based on Two-Layer Neural Networks

Neural networks are powerful tools for approximating high dimensional data that have been used in many contexts, including solution of partial differential equations. We describe a solver for multiscale fully nonlinear elliptic equations that makes use of domain decomposition, an accelerated Schwarz framework, and two-layer neural networks to approximate the boundary-to-boundary map for the subdomains, which is the key step in the Schwarz procedure. Conventionally, the boundary-to-boundary map requires solution of boundary-value elliptic problems on each subdomain. By leveraging the compressibility of multiscale problems, our approach trains the neural network offline to serve as a surrogate for the usual implementation of the boundary-to-boundary map. Our method is applied to a multiscale semilinear elliptic equation and a multiscale p-Laplace equation. In both cases we demonstrate significant improvement in efficiency as well as good accuracy and generalization performance.

3.1 Introduction

Approximation theory plays a key role in scientific computing, including in the design of numerical PDE solvers. This theory prescribes a certain form of ansatz to approximate a solution to the PDE, allowing derivation of an algebra problem whose solution yields the coefficients in the ansatz. Various methods are used to fine-tune the process of translation to an algebraic problem, but the accuracy of the calculated solution is essentially determined by the underlying approximation theory. New approximation methods have the potential to produce new strategies for numerical solution of PDEs.

During the past decade, driven by some remarkable successes in machine learning, neural networks (NNs) have become popular in many contexts. They are extremely powerful in such areas as computer vision, natural language processing, and games [154, 112]. What kinds of functions are well approximated by NNs, and what are the advantages of using NNs in the place of more traditional approximation methods? Some studies [37, 152, 92] have revealed that NNs can represent functions in high dimensional spaces very well. For Barron functions, in particular, unlike traditional approximation techniques that require a large number of parameters (exponential on the dimension), the number of parameter required for a NN to achieve a prescribed accuracy is rather limited. In this sense, NN approximation overcomes the "curse of dimensionality." This fact opens up many possibilities in scientific computing, where the discretization of high dimensional problems often plays a crucial role. One example is problems from uncertainty quantification, where many random variables are needed to represent a random field, with each random variable essentially adding an extra dimension to the PDE [214, 215, 110, 20]. Techniques that exploit intrinsic low-dimensional structures can be deployed on the resulting high-dimensional problem [105, 47, 38, 80, 127]. Another example comes from PDE problems in which the medium contains structures at multiple scales or is highly oscillatory, so that traditional discretization techniques require a large number of grid points to achieve a prescribed error tolerance. Efficient algorithms must then find ways to handle or compress the many degrees of freedom.

Despite the high dimensionality in these examples, successful algorithms have been developed, albeit specific to certain classes of problems. With the rise of NN approximations, with their advantages in high-dimensional regimes, it is reasonable to investigate whether strategies based on NNs can be developed that may even outperform classical strategies. In this chapter, we develop an approach that utilizes a two-layer NN to solve multiscale elliptic PDEs. We test our strategy on two nonlinear problems of this type.

The use of NN in numerical PDE solvers is no longer a new idea. Pioneering works that solving PDEs with NN often take the approach of using NN to approximate the *solutions*. Notable examples include the Physics Informed Neural Network [188], the Deep Ritz method [94], the Deep Backward SDE method [91], the Deep Galerkin method [196], the Weak Adversarial Network [218], a method based on the Feymann-Kac formula [46] and the Multi-scale Deep Neural Network [167, 158], to name a few. Another category of approaches uses NN to approximate the *solution map*. Pioneer works in this category include DeepONet [170], Fourier Neural Operator [161], PDE-Net [168], Bufferfly-Net [159, 216], methods based on hierarchical matrices [103, 102, 210], the Switch-Net [144] and methods based on modal space [213], to name a few. Due to the complicated and unconventional nature of approximation theory for NN, it is challenging to perform rigorous numerical analysis for the methods above, though solid evidence has been presented of the computational efficacy of these approaches.

The remainder of this chapter is organized as follows. In Section 3.2, we discuss our NN-based approach in detail and justify its use in this setting. We then present our reduced-order Schwarz method based on two-layer neural networks. Numerical evidence is reported in Section 3.3. Two comprehensive numerical experiments for the semilinear elliptic equation and the p-Laplace equation are discussed, and efficiency of the methods is evaluated. We make some concluding remarks in Section 3.4.

3.2 Reduced order Schwarz method based on neural networks

As we have seen from Section 2.2, the major numerical expense in the Schwarz iteration comes from the local PDE solves — one per subdomain per iteration. However, except at the final step where we assemble the global solution, our interest is not in the local solutions per se: It is in the boundary-to-boundary maps that share information between adjacent subdomains on each Schwarz iteration. If we can implement these maps *directly*, we can eliminate the need for local PDE solves. To this end, we propose an offline-online procedure. In the offline stage, we implement the boundary-to-boundary maps, and in the online stage, we call these maps repeatedly in the Schwarz framework. This approach is summarized in Algorithm 2. In this description, we replace the boundary-to-boundary map $\mathcal{Q}_m^\varepsilon$ by a surrogate $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$, which is neural network parametrized by weights θ_m, whose values are found by an offline training process.

Algorithm 2 The NN-Schwarz iteration for nonlinear elliptic equations (2.1).

Domain Decomposition:
  Decompose Ω into overlapping patches: $\Omega = \bigcup_{m\in J}\Omega_m$, and collect the indices for interior patches in $J_{\mathrm{i}} = \{m \in J : \partial\Omega_m \cap \partial\Omega = \emptyset\}$ and boundary patches in $J_{\mathrm{b}} = \{m \in J : \partial\Omega_m \cap \partial\Omega \neq \emptyset\}$. CHR: (There's a specific symbol for $\emptyset$.)[SCH: fixed]
Offline training:
  For each interior patch Ω_m, train the boundary-to-boundary map $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ parametrized by θ_m.
Schwarz iteration (Online):
  Given the tolerance δ_0 and the initial guess of boundary conditions $\phi_m^{(0)}$ on each patch $m \in J$.
  Set $n = 0$ and res $= 1$.
  while res $\geq \delta_0$ **do**
    For $m \in J_{\mathrm{i}}$, compute function $(\psi_{m,l}^{(n)})_{l\in\mathscr{N}(m)} = \mathcal{Q}_m^{\mathrm{NN}}(\theta_m)\phi_m^{(n)}$;
    For $m \in J_{\mathrm{b}}$, compute function $\psi_{l,m}^{(n)} = \mathcal{I}_{m,l}\mathcal{S}_m^\varepsilon\phi_m^{(n)}$ for $l \in \mathscr{N}(m)$;
    For $m \in J$, update $\phi_m^{(n+1)}$ by (4.15);
    Set res $= \sum_m \|\phi_m^{(n+1)} - \phi_m^{(n)}\|_{L^2(\partial\Omega_m)}$ and $n \leftarrow n+1$.
  end while
  For $m \in J$, compute function $u_m^{(n)} = \mathcal{S}_m^\varepsilon\phi_m^{(n)}$;
return Global solution $u^{(n)} = \sum_{m\in J}\chi_m u_m^{(n)}$.

Since the online stage is self-explanatory, we focus on the offline stage, and study how to obtain the approximation to $\mathcal{Q}_m^\varepsilon$.

We have two additional comments about our approach.

- Our algorithm uses the surrogate boundary-to-boundary map only for the interior patches $m \in J_{\mathrm{i}}$. For patches that are adjacent to the physical boundary, we perform the standard Schwarz iteration. This choice is mainly for convenience of coding. There are several other options. For example, one can choose to learn in the offline stage the surrogate boundary-to-boundary map for patches boundary patches J_{b} as well. However, in the training stage, one needs to impose a rather general class of functions to serve as the potential physical boundary condition. Whether this chosen class of functions represents well the boundary condition given in the online phase is a question for approximation theory. We omit a discussion here.

- If the PDE operator F^ε has no explicit dependence on x, then the boundary-to-boundary map is the same across all patches of the same size. In this case, training can be implemented in parallel, saving computational time.

3.2.1 Two observations

A rigorous approach to preparing the boundary-to-boundary map $\mathcal{Q}_m^\varepsilon$ in the offline stage is not straightforward. In the case of linear PDEs, it amounts to computing all Green's functions in the local subdomains and confining them on the adjacent subdomain boundaries for the map; see [62]. When the PDEs are nonlinear, there would seem to be no alternative to solving the local PDEs with all possible configurations of the boundary conditions, applying the appropriate restrictions, and storing the results. At the discrete level, $\mathcal{Q}_m^\varepsilon$ would be represented as a high-dimensional function mapping $\mathbb{R}^{d_m}$ to $\mathbb{R}^{p_m}$. To achieve a specified accuracy, both d_m and p_m need to scale as $O(\varepsilon^{-(d-1)})$. For brute-force training, at least $O(d_m) = O(\varepsilon^{-(d-1)})$ local PDE solves need to be performed to compute the required approximation to $\mathcal{Q}_m^\varepsilon$. This is a large amount of computation, and it offsets whatever gains accrue in the online stage from efficient deployment of the approximation

to $\mathcal{Q}_m^\varepsilon$.

To be cost-effective, a method of the form of Algorithm 2 must exploit additional properties, intrinsic to $\mathcal{Q}_m^\varepsilon$ and to the scheme for approximating this mapping. The first such property is a direct consequence of homogenization. As argued in Section 2.1, the solution of the effective equation (2.3) can preserve the ground truth well, with the effective equation independent of ε. Therefore, the map $\mathcal{Q}_m^\varepsilon$, though presented as a mapping from $\mathbb{R}^{d_m}$ to $\mathbb{R}^{p_m}$, is intrinsically of low dimension and can be compressed. To visualize this relation, we plot the relative singular values of the boundary-to-boundary operator $\mathcal{Q}_m^\varepsilon$ of a linear multiscale elliptic equation (see (3.17)) in Figure 3.1.

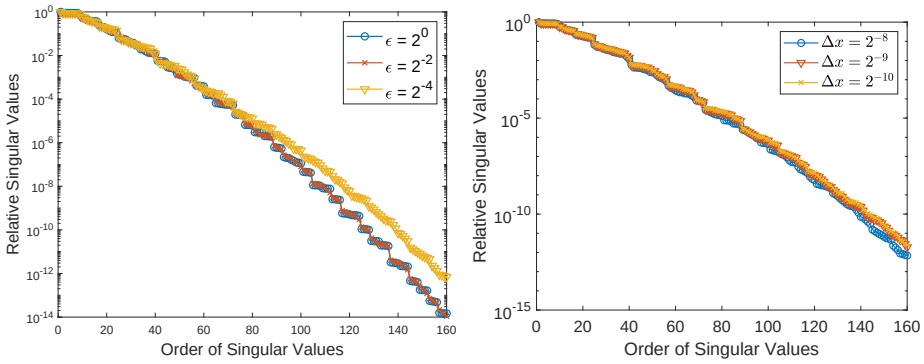


Figure 3.1: Singular values of the boundary-to-boundary operator $\mathcal{Q}_m^\varepsilon$ for the linear elliptic equation (3.17) with medium κ^ε defined in (3.15) for different values of ε and Δx on a local patch. Left plot: $\Delta x = 2^{-8}$. Right plot: $\varepsilon = 2^{-4}$. To ensure the regularity of the test function space, the discrete version of the boundary-to-boundary map is represented on basis functions composed of piecewise linear function with fixed step size 2^{-8}.

With the system being of intrinsically low dimension, we expect that a compression mechanism can be deployed. Even though the data itself is represented in high dimension, the number of parameters in the compressed representation should not grow too rapidly with the order of discretization. We seek an approximation strategy that can overcome the "curse of dimensionality." These considerations lead us to the use of neural network (NN). NN, unlike other approximation techniques, is powerful in learning functions supported in high dimensional space; the number of parameters that need to be tuned to fit data in a high dimensional space is typically relaxed from the dimension of the data.

Consider a fully connected feedforward neural network (fully connected NN) repre-

senting a function $f : \mathbb{R}^n \to \mathbb{R}^m$. A 2-layer fully connected NN with hidden-layer width h would thus be required to satisfy

$$f^{\mathrm{NN}}(x) = W_2\sigma(W_1x + b_1) + b_2\,, \quad x \in \mathbb{R}^n\,, \tag{3.1}$$

where $W_1 \in \mathbb{R}^{h\times n}$, $W_2 \in \mathbb{R}^{m\times h}$ are weight matrices and $b_1 \in \mathbb{R}^h$, $b_2 \in \mathbb{R}^m$ are biases. The activation function $\sigma : \mathbb{R} \to \mathbb{R}$ is applied component-wise to its argument. (The ReLU activation function $\sigma(x) = \max(x, 0)$ is especially popular.) This 2-layer fully connected NN already can represent high dimensional functions. A fundamental approximation result [152, 95, 37] is captured in the following theorem.

Theorem 3.1 (Barron's Theorem)**.** *Let $D \subset \mathbb{R}^n$ be a bounded domain. Suppose a generic function $f \in L^2(D)$ satisfies*

$$\Delta(f) = \int_{\mathbb{R}^n} \|\omega\|_1^2 |\hat{f}(\omega)| \mathrm{d}\omega < \infty\,, \tag{3.2}$$

where $\hat{f}$ is the Fourier transform of the zero extension of f to $L^2(\mathbb{R}^d)$. Then there exists a two-layer ReLU neural network f^{NN} with h hidden-layer neurons such that

$$\|f - f^{\mathrm{NN}}\|_{L^2(D)} \lesssim \frac{\Delta(f)}{\sqrt{h}}\,. \tag{3.3}$$

A natural high dimensional extension of the result is as follows.

Corollary 3.1. *Let $D \subset \mathbb{R}^n$ be a bounded domain. Suppose a generic function $f = [f_1\,, \ldots, f_m] : \mathbb{R}^n \to \mathbb{R}^m$ so that $f_i \in L^2(D)$ satisfies* (3.2)*, then there exists a two-layer ReLU neural network f^{NN} with h hidden-layer neurons such that*

$$\|f - f^{\mathrm{NN}}\|_{L^2(D)} \lesssim \sqrt{\sum_{i=1}^m \frac{\Delta^2(f_i)}{h/m}} \leq m\frac{\Delta(f)}{\sqrt{h}}\,. \tag{3.4}$$

where $\Delta(f) := \max_{i=1}^m \Delta(f_i)$.

A nice feature of this result is that the approximation error is mostly relaxed from the

dimension of the problem, making NN a good fit for our purposes. In our setting, it is the high-dimensional operator $\mathcal{Q}_m^\varepsilon$ that needs to be learned. Theorem 3.1 suggests that if fully connected NN is used as the representation, the number of neurons h required will not depend strongly on this dimension.

3.2.2 Offline training and the full algorithm

The two observations above suggest that using a neural-network approximation for the boundary-to-boundary operator can reduce computation costs and memory significantly. Following (3.1), we define the NN approximation $\mathcal{Q}_m^{\mathrm{NN}}$ to $\mathcal{Q}_m^\varepsilon$ as follows:

$$\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)\phi_m = W_{m,2}\sigma(W_{m,1}\phi_m + b_{m,1}) + b_{m,2}\,, \quad \text{where } \phi_m \in \mathbb{R}^{d_m}. \tag{3.5}$$

Here $\theta_m = \{W_{m,1}, W_{m,2}, b_{m,1}, b_{m,2}\}$ denotes all learnable parameters, with weight matrices $W_{m,1} \in \mathbb{R}^{h_m \times d_m}, W_{m,2} \in \mathbb{R}^{p_m \times h_m}$ and biases $b_{m,1} \in \mathbb{R}^{h_m}, b_{m,2} \in \mathbb{R}^{p_m}$. The number of neurons h_m is a tunable parameter that relates to the number of degrees of freedom in $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$. Theorem 3.1 and the homogenizability of the elliptic equation suggest that h_m can be chosen to satisfy a prescribed approximation error while being independent of both d_m and p_m, and thus of the small scale ε.

Given a fixed NN architecture and a data set, the identification of optimal $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ amounts to minimizing a loss function $\mathcal{L}(\theta_m)$ that measures the misfit between the data and the prediction. One needs to prepare a set of data $\mathscr{X}_m = \{\phi_{m,i}\}_{i=1}^N$ and corresponding outputs

$$\mathscr{Y}_m = \left\{\psi_{m,i} = \mathcal{Q}_m^\varepsilon \phi_{m,i} = (\psi_{l,m,i})_{l\in\mathscr{N}(m)} = \left(u_{m,i}^\varepsilon|_{\partial\Omega_l\cap\Omega_m}\right)_{l\in\mathscr{N}(m)}\right\}_{i=1}^N, \tag{3.6}$$

where $u_{m,i}^\varepsilon$ solves (2.8). The loss function to be minimized is

$$\mathcal{L}(\theta_m) := \frac{1}{N}\sum_{i=1}^N \ell\left(\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)\phi_{m,i}\,, \psi_{m,i}\right), \tag{3.7}$$

where ℓ evaluates the mismatch between the first and the second arguments. (This measure could be defined using the L_2 norm and / or the H^1 norm.) Gradient-based algorithms for minimizing (3.7) have the general form

$$\theta_m^{(t+1)} \leftarrow \theta_m^{(t)} - \eta_t G_t \left(\nabla_{\theta_m} \mathcal{L} \left(\theta_m^{(t)} \right), \ldots, \nabla_{\theta_m} \mathcal{L} \left(\theta_m^{(1)} \right) \right) , \tag{3.8}$$

where η_t is the learning rate and G_t is based on the all gradients seen so far. For example, for the Adam optimizer [146], the function G_t is a normalized exponentially decaying average of gradients:

$$G_t \left(a_t, \ldots, a_1 \right) \propto \left(1 - \beta_1^t \right)^{-1} \sum_{s=1}^{t} \beta_1^{t-s} \left(1 - \beta_1 \right) a_s \, , \tag{3.9}$$

for some parameter $\beta_1 \in (0, 1)$. The $\propto$ sign means G_t needs to be normalized so that $\|G_t\|_2 \sim 1$.

Like many optimization processes, the training and tuning of this NN depends on some prior knowledge. We propose a mechanism to select training data that represent well the information in $\mathcal{Q}_m^\varepsilon$. We also initialize the weights θ_m according to a reduced linear problem. These mechanisms are described in the following two sections; their effectiveness in numerical testing is demonstrated in Section 3.3.

Generating training data

To learn the parameters in the NN approximation to the boundary-to-boundary map, one needs to provide a training set of examples of the map. We generate such examples by adding a boundary margin of width Δx_{b} to each interior patch Ω_m to obtain an enlarged patch $\overline{\Omega}_m$, as shown in Figure 4.1. Samples are generated by choosing Dirichlet conditions for the enlarged patch, then solving the equation, and defining the map in terms of restrictions of both input and output conditions to the appropriate boundaries.

Specifically, following [71], we generate N i.i.d. samples of the boundary conditions $\overline{\phi}_m$ for the enlarged patch $\partial\overline{\Omega}_m$ according to $H^{1/2}(\partial\overline{\Omega})$ (See Appendix A.1), and solve the

following equations for $\overline{u}^{\varepsilon}_{m,i}(x)$:

$$\begin{cases} F^{\varepsilon}\left(D^2\overline{u}^{\varepsilon}_{m,i}(x), D\overline{u}^{\varepsilon}_{m,i}(x), \overline{u}^{\varepsilon}_{m,i}(x), x\right) = 0\,, & x \in \overline{\Omega}_m\,, \\ \overline{u}^{\varepsilon}_{m,i}(x) = \overline{\phi}_{m,i}(x)\,, & x \in \partial\overline{\Omega}_m\,. \end{cases} \tag{3.10}$$

The boundary-to-boundary map $\mathcal{Q}^{\varepsilon}_m$ maps each element of $\mathscr{X}_m = \{\phi_{m,i}\}_{i=1}^N$ to the corresponding element of $\mathscr{Y}_m = \{\psi_{m,i}\}_{i=1}^m$, where

$$\phi_{m,i} = \overline{u}^{\varepsilon}_{m,i}|_{\partial\Omega_m}\,, \quad \psi_{m,i} = (\psi_{l,m,i})_{l\in\mathscr{N}(m)} = \left(\overline{u}^{\varepsilon}_{m,i}|_{\partial\Omega_l\cap\Omega_m}\right)_{l\in\mathscr{N}(m)}\,. \tag{3.11}$$

This pair of sets — input set $\mathscr{X}_m$ and output set $\mathscr{Y}_m$ — serves as the training data. We have two comments regarding the training process.

- Various NN architectures could be considered. We use fully connected NN mostly because the initialization procedure requires singular value decomposition of the linearized counterpart of boundary-to-boundary map, and this kind of NN is a natural extension of the linear network that capture the SVD to the nonlinear regime. Another potentially good option is Convolutional Neural Network (CNN) whose structure can potentially alleviate computational difficulty as one refines the discretization.

- For generating training data, each patch is slightly enlarged before application of the PDE solver. Most homogenization results need a boundary layer correction adjacent to the physical boundary, so the low-rank property fails to hold near the boundary. The use of a small boundary buffer zone on each patch dampens the boundary layer effect and enables low-rank structure of the boundary-to-boundary map.

Initialization

The training problem of minimizing $\mathcal{L}(\theta_m)$ in (3.7) to obtain the NN approximate operator $\mathcal{Q}^{\mathrm{NN}}_m(\theta_m)$ is nonconvex, so a good initialization scheme can improve the performance of a gradient-based optimization scheme significantly. We can make use of knowledge about

the PDE to obtain good starting points. Our strategy is to assign good initial weights and biases for the neural network using a *linearization* of the fully nonlinear elliptic equation (2.1). Denoting by $\mathcal{Q}_m^{\mathrm{L}}$ the boundary-to-boundary operator of a linearized version of $\mathcal{Q}_m^{\varepsilon}$, to be made specific below for the numerical examples in Section 3.3, we initialize $\mathcal{Q}_m^{\mathrm{NN}}$ in a way that approximately captures $\mathcal{Q}_m^{\mathrm{L}}$. The linear boundary-to-boundary operator $\mathcal{Q}_m^{\mathrm{L}}$ has a matrix representation. Denoting by r_m the approximate rank (up to a preset error tolerance), we can write

$$\mathcal{Q}_m^{\mathrm{L}} \approx U_{m,r_m}\Lambda_{r_m}V_{m,r_m}^{\top} = \left(U_{m,r_m}\sqrt{\Lambda_{r_m}}\right)\left(V_{m,r_m}\sqrt{\Lambda_{r_m}}\right)^{\top}, \tag{3.12}$$

where $U_{m,r_m} \in \mathbb{R}^{p_m\times r_m}$ and $V_{m,r_m} \in \mathbb{R}^{d_m\times r_m}$ have orthonormal columns while $\Lambda_{r_m} \in \mathbb{R}^{r_m\times r_m}$ is diagonal. As argued in [62], due to the fact that the underlying equation is homogenizable, this rank r_m is much less than $\min\{d_m, p_m\}$, and is independent of p_m and d_m.

To start the iteration of $\mathcal{Q}_m^{\mathrm{NN}}$, we compare (3.5) with the form of (3.12). This suggests the following settings of parameters in (3.5): $b_{m,1} = b_{m,2} = 0$ and

$$\begin{aligned} W_{m,1} &= \left[V_{m,r_m}\sqrt{\Lambda_{r_m}}, -V_{m,r_m}\sqrt{\Lambda_{r_m}}\right]^{\top}, \\ W_{m,2} &= \left[U_{m,r_m}\sqrt{\Lambda_{r_m}}, -U_{m,r_m}\sqrt{\Lambda_{r_m}}\right]. \end{aligned} \tag{3.13}$$

Note that $h_m = 2r_m$. These configurations will be used as the initial iteration in (3.8).

We summarize our offline training method in Algorithm 3. Integrating into the full algorithm yields the reduced order neural network based Schwarz iteration method.

Algorithm 3 Offline training of $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$, as a surrogate of $\mathcal{Q}_m^{\varepsilon}$ on patch Ω_m.

1: Enlarge each interior patch Ω_m to obtain $\overline{\Omega}_m$;
2: Randomly generate samples $\{\overline{\phi}_{m,i}\}_{i=1}^N$ and solve (3.10) to obtain $\{\overline{u}_{m,i}^{\varepsilon}\}_{i=1}^N$.
3: Compute (4.10) to define $\{\mathscr{X}_m, \mathscr{Y}_m\} = \{\{\phi_{m,i}\}_{i=1}^N, \{(\psi_{l,m,i})_{l\in\mathscr{N}(m)}\}_{i=1}^N\}$;
4: Initialize θ_m in $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ by using the linearized boundary-to-boundary operator $\mathcal{Q}_m^{\mathrm{L}}$, as defined in (3.13);
5: Find the optimal coefficient θ_m^* in the neural network $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ by applying the gradient descent method (3.8) until convergence.

3.3 Numerical results

We present numerical examples using our proposed method to solve a multiscale semilinear elliptic equation and a multiscale p-Laplace equation. In both examples, we use domain $\Omega = [0,1]^2$. To form the partitioning, Ω is divided into $M_1 \times M_2$ equal non-overlapping rectangles, then each rectangle is enlarged by Δx_{o} on the sides that do not intersect with $\partial\Omega$, to create overlap. We thus have

$$\begin{aligned}\Omega_m = &\left[\max\left(\tfrac{m_1-1}{M_1} - \Delta x_{\mathrm{o}}, 0\right), \min\left(\tfrac{m_1}{M_1} + \Delta x_{\mathrm{o}}, 1\right)\right] \\ &\times \left[\max\left(\tfrac{m_2-1}{M_2} - \Delta x_{\mathrm{o}}, 0\right), \min\left(\tfrac{m_2}{M_2} + \Delta x_{\mathrm{o}}, 1\right)\right], \quad m = (m_1, m_2) \in J\,.\end{aligned}$$

The loss function is defined as in (3.7), with parameter $\mu = 10^{-3}$. For training to obtain $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$, we use PyTorch [183]. For both examples, each neural network is trained for 5,000 epochs using shuffled mini-batch gradient descent with a batch-size of 5% of the training set size. The Adam optimizer is used with default settings, and the learning rate decays with a decay-rate of 0.9 every 200 epochs. The codes accompanying this chapter are publicly available [68].

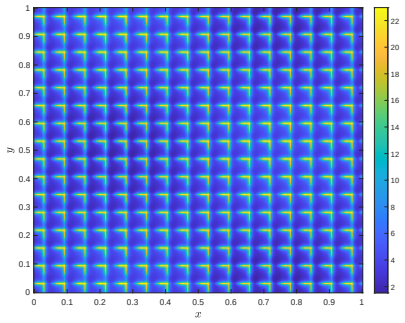


Figure 3.2: Medium κ for semilinear elliptic equation.

3.3.1 Semilinear elliptic equations

The first example is the semilinear elliptic equation

$$\begin{cases} -\nabla \cdot (\kappa^\varepsilon(x)\nabla u^\varepsilon(x)) + u^\varepsilon(x)^3 = 0, & x \in \Omega\,, \\ u^\varepsilon(x) = \phi(x), & x \in \partial\Omega\,, \end{cases} \tag{3.14}$$

with oscillatory medium $\kappa^\varepsilon(x) = \kappa^\varepsilon(x_1, x_2)$ defined by

$$\kappa^\varepsilon(x_1, x_2) = 2 + \sin(2\pi x_1)\cos(2\pi x_2) + \frac{2 + 1.8\sin(2\pi x_1/\varepsilon)}{2 + 1.8\cos(2\pi x_2/\varepsilon)} + \frac{2 + \sin(2\pi x_2/\varepsilon)}{2 + 1.8\cos(2\pi x_1/\varepsilon)}\,. \tag{3.15}$$

with $\varepsilon = 2^{-4}$. The medium is plotted in Figure 3.2.

The reference solution and the local PDE solves are computed using the standard finite-volume scheme with uniform grid with mesh size $\Delta x = 2^{-8} = \frac{1}{256}$ and Newton's method is used to solve the resulting algebraic problem. For our domain decomposition approach, we set $M_1 = M_2 = 4$ to define the patches Ω_m, with boundary margins $\Delta x_{\mathrm{o}} = 2^{-4} = \frac{1}{16}$ to form Ω_m. The input and output dimensions of Q_m^ε are thus $(d_m, p_m) = (388, 388)$.

To obtain the training data, each patch Ω_m is further enlarged to a buffered patch $\overline{\Omega}_m$ by adding a margin of $\Delta x_{\mathrm{b}} = 2^{-4} = \frac{1}{16}$ to Ω_m. On each patch $\overline{\Omega}_m$, $10,000$ samples are generated with random boundary conditions defined by $R_m = 1000$ and $D = 3$. To train the NN, we use the loss function (3.7) with

$$\begin{aligned} \ell(\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)\phi_m - \psi_m) =& \|\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)\phi_m - \psi_m\|^2 \\ &+ \mu \sum_{l \in \mathscr{N}(m)} \|D_h \mathcal{Q}_{l,m}^{\mathrm{NN}}(\theta_m)\phi_m - D_h\psi_{l,m}\|^2\,, \end{aligned} \tag{3.16}$$

where $\mathcal{Q}_{l,m}^{\mathrm{NN}}(\theta_m)$ and $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ are NN approximation of $\mathcal{Q}_{l,m}(\theta_m)$ and $\mathcal{Q}_{l,m}(\theta_m)$, as defined in (2.7) and (2.9), respectively; and D_h is the discrete version of the derivative operator with step size h. The second term measures mismatch in the derivative so as to enforce the regularity.

To initialize the neural networks, we take $\mathcal{Q}_m^{\mathrm{L}}$ to be the boundary-to-boundary operator

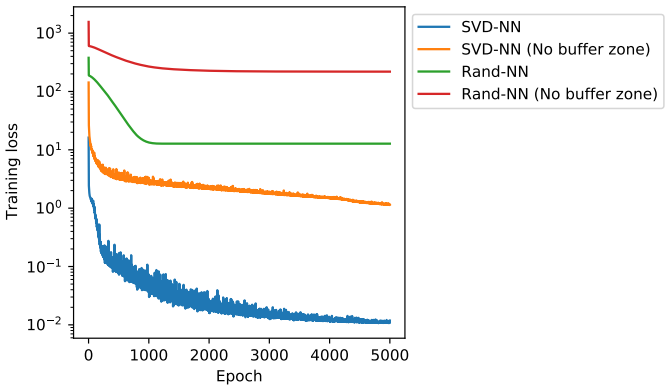


Figure 3.3: Training loss for loss function $\mathcal{L}$ (3.16) for patch (2,2). For the variants that use random initializations, we use the PyTorch default, which generate the weights and biases in each layer uniformly from $(-\sqrt{d_{\text{input}}}, \sqrt{d_{\text{input}}})$, where d_{input} is the input dimension of the layer.

of the following linear elliptic equation

$$-\nabla \cdot (\kappa^{\varepsilon}(x)\nabla u^{\varepsilon}(x)) = 0, \quad x \in \Omega\,. \tag{3.17}$$

We truncate the rank representation of $\mathcal{Q}_m^{\text{L}}$ at rank $r_m = 40$ to preserve all singular values bigger than a tolerance $\delta_1 = 10^{-2}$ so that the width of the hidden layer is $h_m = 80$.

Offline training

We show the improvements in the offline process for training $\mathcal{Q}_m^{\text{NN}}$ due to the two strategies described in Subsection 3.2.2: the use of enlarged patches, and initialization using SVD of a matrix representation of a linearized equation. Figure 3.3 plots number of epochs in the offline training vs the training loss function $\mathcal{L}$ (3.16) associated with $\mathcal{Q}_m^{\text{NN}}$ for the patch $m = (2, 2)$ in four different settings: SVD-initialization on training data with buffer zone, SVD-initialization on training data without buffer zone, and the counterpart without SVD-initialization. The same NN model is used in all four settings. It is immediate that the training process has a much faster decay in error if buffer zone is adopted, and that the SVD initialization gives a much smaller error than random initialization.

To show the generalization performance of the resulting trained NN, we generate a test

data set from the same distribution as the buffered training data set with 1,000 samples, for the same patch $m = (2, 2)$. Since the NNs trained using non-buffered data produce larger error, we only test the NNs trained with buffered data. The test errors (3.16) in the training process for different models are plotted in Figure 3.4. Again, the use of buffered data along with SVD-initialization yields the best performance.

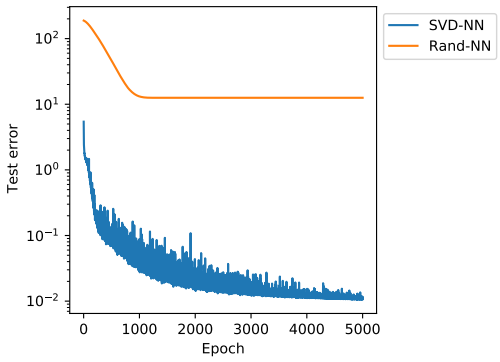


Figure 3.4: Testing error during the training for patch (2,2).

To demonstrate generalization performance, we plot the predicted outputs for two typical examples in the test set in Figure 3.5. For comparison, we also plot the outputs produced by randomly initialized neural network and the linear operator $\mathcal{Q}_m^{\mathrm{L}}$. It can be seen that the low-rank SVD-initialized neural network has the best performance among all the initialization methods.

We note too that the neural network models initialized by the SVD of linear PDEs tend to be more interpretable. Figure 3.6 shows the final weight matrices for models initialized by different methods. It can be seen that SVD-initialized model yields weight matrices with recognizable structure: the parameters for higher modes are near zero, and only the top 25 modes in the positive and negative halves are nontrivial. By comparison, the trained weight matrices using randomly initialized parameters do not show any pattern or structure.

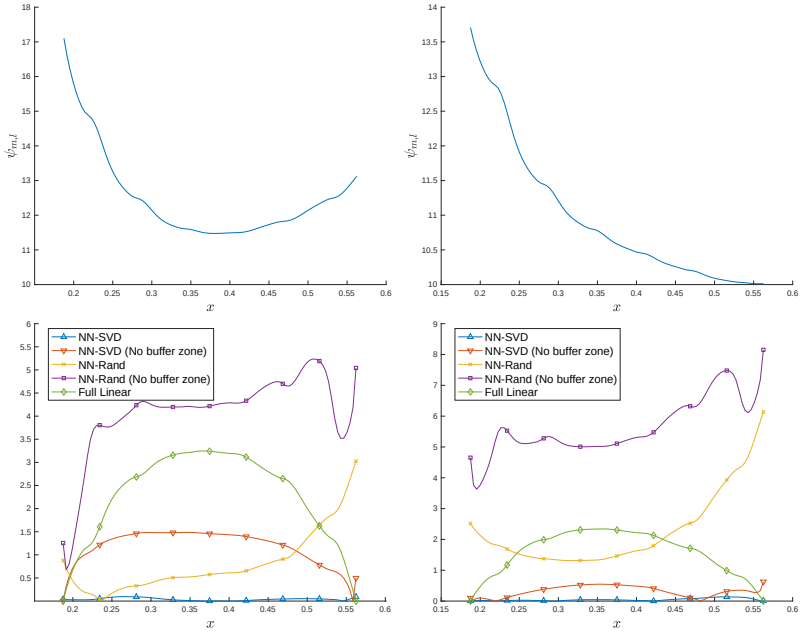


Figure 3.5: The top row shows the ground truths $\psi_{l,m}$ ($m = (2,2)$, $l = (2,1)$) of two samples in the test set. The bottom row shows the error $|\psi_{l,m} - \widetilde{\psi}_{l,m}|$, where $\widetilde{\psi}_{l,m}$ are computed by the low-rank SVD initialized $\mathcal{Q}_m^{\mathrm{NN}}$ (with and without buffer-zone), randomly initialized $\mathcal{Q}_m^{\mathrm{NN}}$ (with and without buffer-zone), and the linear operator $\mathcal{Q}_m^{\mathrm{L}}$.

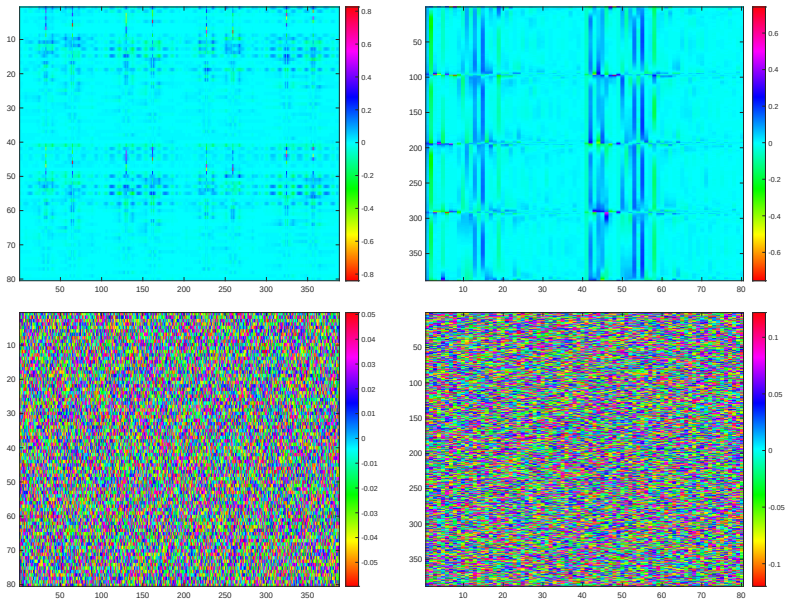


Figure 3.6: The first row shows the final weight matrices W_1 (left), W_2 (right) obtained the for SVD-initialized model on patch $m = (2,2)$. The second row shows the final weight matrices W_1 (left), W_2 (right) for randomly initialized model on patch $m = (2,2)$. In both cases, training data is obtained by enlarging the patch.

No.	Boundary condition
1	$\phi(x,0) = 40\,,\ \phi(x,1) = 40$ $\phi(0,y) = 40\,,\ \phi(1,y) = 40$
2	$\phi(x,0) = 50 - 50\sin(2\pi x)\,,\ \phi(x,1) = 50 + 50\sin(2\pi x)$ $\phi(0,y) = 50 + 50\sin(2\pi y)\,,\ \phi(1,y) = 50 - 50\sin(2\pi y)$
3	$\phi(x,0) = 10\,,\ \phi(x,1) = 35$ $\phi(0,y) = 10 + 25y\,,\ \phi(1,y) = 10 + 25y$

Table 3.1: Boundary conditions used in the global test.

Online phase: Schwarz iteration

We show results obtained by using the NN approximation $\mathcal{Q}_m^{\mathrm{NN}}(\theta_m)$ of the boundary-to-boundary map inside the Schwarz iteration. Table 3.1 shows the boundary conditions used for the three problems we tested. (The same medium (3.15) is used in all cases.) We use $\delta_0 = 10^{-4}$ for the tolerance in Algorithm 2, and use the full accuracy local solvers as in the generation of training data set. In Figure 3.7, we plot the ground truth solutions for different boundary conditions and the absolute error of u^{NN} obtained by neural network-based Schwarz iteration. (Note that the scaling of the y-axis in the latter is different from the former.) The relative errors obtained for the four variants of NN approximation along with the linear approximation $\mathcal{Q}_m^{\mathrm{L}}$ to the boundary-to-boundary map can be found in Tables 3.2 and 3.3. Note that the smallest errors are attained by the variant that uses the SVD initialization and buffered patches. To demonstrate the efficiency of our method, we compare the CPU time of neural network based-Schwarz method and the classical Schwarz method, using the same tolerance $\delta_0 = 10^{-4}$ for the latter. The NNs we used for the test is trained by SVD initialization, and its training data is generated with buffer zone. Since NN-produced local boundary-to-boundary map is only an approximation to the ground truth, for a fair comparison, we also run the reference local solution with a relaxed accuracy requirement. The CPU time, number of iteration and error comparison can be found in Table 3.4. In all three test cases, the NN approximate executes faster than the conventional local solution technique as a means of implementing the boundary-to-boundary map, while producing H^1 errors of the same order.

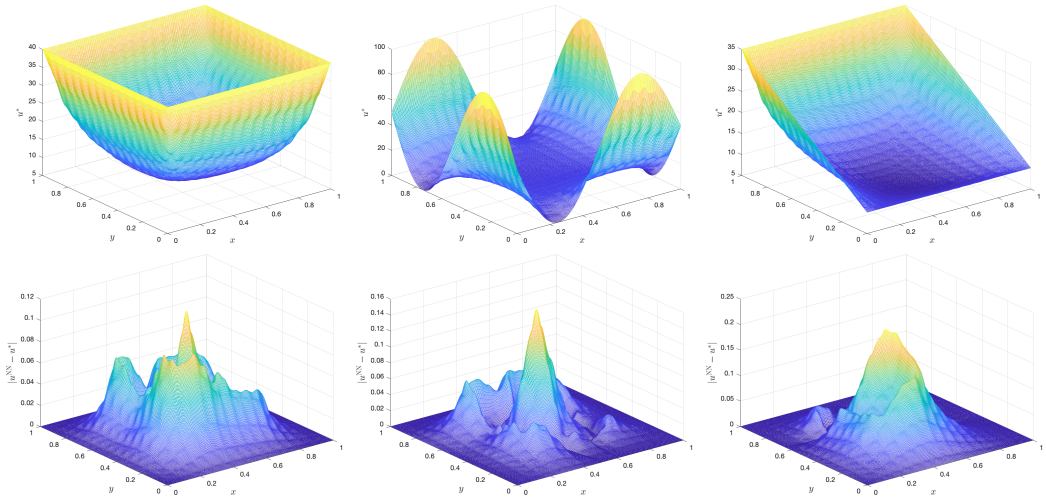


Figure 3.7: The first row shows the ground truth solutions u^* for boundary conditions 1 to 3 from left to right. The second row shows the absolute error $|u^{\mathrm{NN}} - u^*|$ for boundary conditions 1 to 3 from left to right. (Note the much smaller vertical scale used in the second row.)

Problem Number	1			2		
Relative Error	L^2	H^1	L^∞	L^2	H^1	L^∞
SVD-NN	**0.0013**	**0.0028**	**0.0029**	**0.0010**	**0.0010**	**0.0016**
SVD-NN (No buffer zone)	0.0042	0.0091	0.0078	0.0029	0.0030	0.0039
Rand-NN	0.0425	0.0445	0.0907	0.0379	0.0188	0.0370
Rand-NN (No buffer zone)	0.0882	0.0965	0.1555	0.0773	0.0400	0.0629
Linear	0.0606	0.0644	0.1066	0.0505	0.0252	0.0415

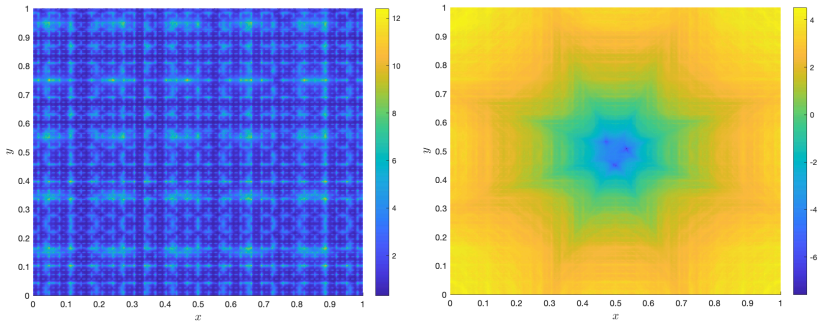
Table 3.2: Relative error for global solutions by different methods.

Problem Number	3		
Relative Error	L^2	H^1	L^∞
SVD-NN	**0.0035**	**0.0059**	**0.0058**
SVD-NN (No buffer zone)	0.0235	0.0341	0.0346
Rand-NN	0.1029	0.1293	0.1333
Rand-NN (No buffer zone)	0.1739	0.2277	0.2078
Linear	0.0614	0.0729	0.0776

Table 3.3: Relative error for global solutions by different methods. (Continued)

Problem Number	1		2		3	
Method	NN	Classical	NN	Classical	NN	Classical
CPU time	**12.4**	17.2	**13.9**	18.5	**13.4**	19.5
Iteration	30	29	30	29	34	35
H^1 Error	0.0035	0.0024	0.0012	0.0007	0.0062	0.0022

Table 3.4: CPU time (s), number of iterations and the H^1 error of the classical Schwarz iteration and the neural network accelerated Schwarz iteration.



(a) κ (b) $\log_{10}(\kappa|\nabla u^\varepsilon|^{p-2})$

Figure 3.8: Medium κ and $\kappa|\nabla u^\varepsilon|^{p-2}$ for p-Laplace equation. The solution u^ε is computed by boundary condition 1 (See Table 3.5).

3.3.2 p-Laplace equations

The second example concerns the multiscale p-Laplace elliptic equation [13, 108, 186, 76, 166] defined as follows:

$$\begin{cases} -\nabla \cdot (\kappa^\varepsilon(x)|\nabla u^\varepsilon|^{p-2}\nabla u^\varepsilon(x)) = 0, & x \in \Omega\,, \\ u^\varepsilon(x) = \phi(x), & x \in \partial\Omega\,, \end{cases} \tag{3.18}$$

where we use $p = 6$ in this section, and the oscillatory medium is

$$\begin{aligned} \kappa^\varepsilon(x,y) = \frac{1}{6}\Bigg(&\frac{1.1+\sin(2\pi x/\varepsilon_1)}{1.1+\sin(2\pi y/\varepsilon_1)} + \frac{1.1+\sin(2\pi y/\varepsilon_2)}{1.1+\cos(2\pi x/\varepsilon_2)} + \frac{1.1+\cos(2\pi x/\varepsilon_3)}{1.1+\sin(2\pi y/\varepsilon_3)} \\ &+ \frac{1.1+\sin(2\pi y/\varepsilon_4)}{1.1+\cos(2\pi x/\varepsilon_4)} + \frac{1.1+\cos(2\pi x/\varepsilon_5)}{1.1+\sin(2\pi y/\varepsilon_5)} + \sin(4x^2y^2) + 1\Bigg)\,, \end{aligned} \tag{3.19}$$

with $\varepsilon_1 = 1/5, \varepsilon_2 = 1/13, \varepsilon_3 = 1/17, \varepsilon_4 = 1/31, \varepsilon_5 = 1/65$. See Figure 3.8 for an illustration of the medium.

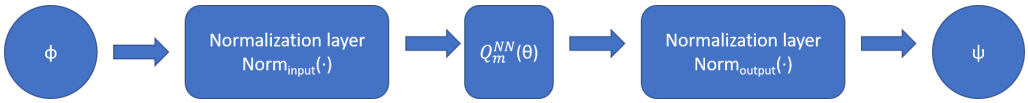


Figure 3.9: Neural network architecture for the boundary-to-boundary map in the p-Laplace equation.

Noting that the differential equation in (3.18) is invariant when a constant is added or when multiplied by a constant, we use a normalization layer to improve the accuracy and robustness of the two-layer neural network. Given a input boundary condition $\phi_m \in \mathbb{R}^{d_m}$, we define the input normalization layer by

$$\mathrm{Norm}_{\mathrm{input}}(\phi_m) = \frac{\phi_m - \widetilde{\phi}_m}{\max\{\|\phi_m\|_2, \varepsilon_1\}} \tag{3.20}$$

where $\widetilde{\phi}_m := 1/d_m \sum_{i=1}^{d_m} (\phi_m)_i$ is the mean, and the norm is defined by $\|\phi_m\|_2^2 = \Delta x \sum_{i=1}^{d_m} (\phi_m)_i^2$. We use $\varepsilon_1 = 10^{-8}$ for the regularization constant. The output normalization layer is defined by

$$\mathrm{Norm}_{\mathrm{output}}(\psi_m) = \max\{\|\phi_m\|_2, \varepsilon_1\}\psi_m + \widetilde{\phi}_m \,. \tag{3.21}$$

The overall architecture is illustrated in Figure 3.9.

To compute both the reference solution and the patchwise solutions (3.10), we formulate the discretization using the standard piecewise linear finite element with uniform triangular grid, and solve with a preconditioned gradient descent method [132], where the line search parameter is computed by the Matlab function fminunc. The mesh size is $\Delta x = 2^{-8} = 1/256$.

For the domain decomposition, we set $M = 8$ with $\Delta x_{\mathrm{o}} = .03125$ to form Ω_m. The resulting input dimension is $d_m = 196$ and the output dimension is $p_m = 196$. Training data is produced on enlarged patches $\overline{\Omega}_m$ with $\Delta x_{\mathrm{b}} = .09375$. On each patch, $1,000$ samples are generated with random distribution parameters $R_m = 10$ and $D = 3$. To initialize the neural networks, we take $\mathcal{Q}_m^{\mathrm{L}}$ to be the boundary-to-boundary operator of the linear elliptic equation $-\nabla \cdot (\kappa^{\varepsilon}(x)\nabla u(x)) = 0$. We truncate the rank presentation of $\mathcal{Q}_m^{\mathrm{L}}$ at rank $r_m = 36$, to preserve all singular values greater than a tolerance $\delta_1 = 10^{-2}$ so

that the width of the hidden layer is $h_m = 72$.

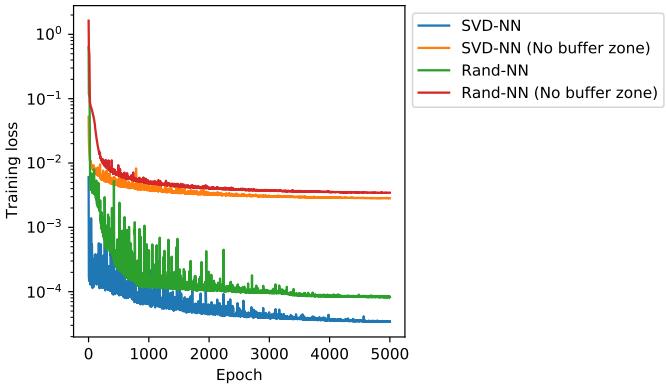


Figure 3.10: Training loss using loss function $\mathcal{L}$ (3.16) for patch $m = (2, 2)$. We use the default random initialization method in PyTorch, which generate the weights and biases in each layer uniformly from $(-\sqrt{d_{\text{input}}}, \sqrt{d_{\text{input}}})$ with d_{input} being the input dimension of the layer.

Offline training

Here we show the improvements in the training process of $\mathcal{Q}_m^{\text{NN}}$ by using the sampling and initialization strategies in Subsection 3.2.2. Figure 3.10 shows the training loss vs epochs for learning $\mathcal{Q}_m^{\text{NN}}$ for the patch $m = (2, 2)$ using 1,000 samples. The four variants are the same as in Figure 3.3. As for the previous example, the most effective training loss is for the variant in which samples are computed from buffered patches, using a reduced SVD initialization based on the linear approximate operator.

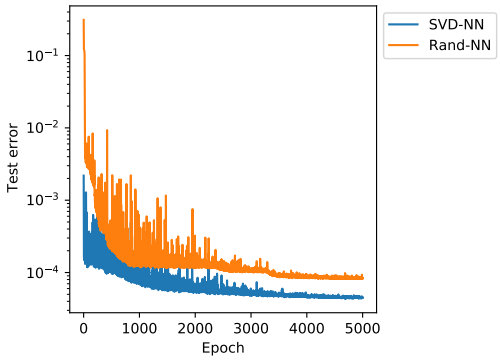


Figure 3.11: Testing error during the training for patch (2,2).

We generate a test data set from the same distribution as the buffered training data set with 100 samples for patch $m = (2, 2)$. The test errors (3.16) in the training process for different models are plotted in Figure 3.11. As for the training loss, the variant with buffered patches and SVD initialization gives the best results.

To demonstrate generalization performance on this example, we plot the predicted outputs for two typical examples in the test set in Figure 3.12. For comparison, we also plot the outputs produced by randomly initialized neural network and the linear operator $\mathcal{Q}_m^{\mathrm{L}}$. The low-rank SVD-initialized neural network shows best reconstruction performance.

Figure 3.13 show the final weight matrices for models initialized by different methods. All weights have non-trivial values, suggesting that the NN has appropriate dimensions for approximating $\mathcal{Q}_m^{\varepsilon}$. Although the structure of the W_2 matrix looks roughly similar for each case, the W_1 matrices are quite different in character, with the randomly initialized version at bottom left having no obvious structure.

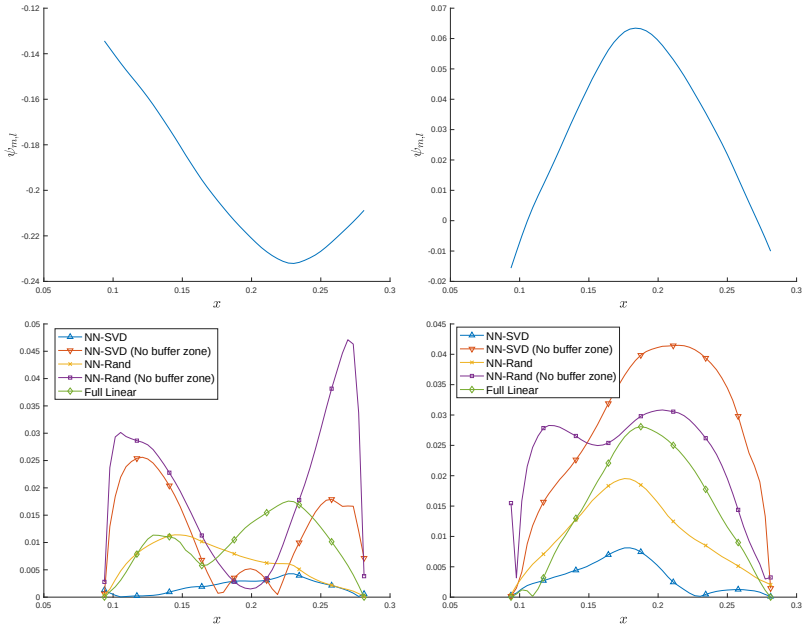


Figure 3.12: The top row shows the ground truths $\psi_{l,m}$ ($m = (2, 2)$, $l = (2, 1)$) of two samples in the test set. The bottom row shows the error $|\psi_{l,m} - \widetilde{\psi}_{l,m}|$, where $\widetilde{\psi}_{l,m}$ are computed by the low-rank SVD initialized $\mathcal{Q}_m^{\mathrm{NN}}$ (with and without buffer zone), randomly initialized $\mathcal{Q}_m^{\mathrm{NN}}$ (with and without buffer zone), and the linear operator $\mathcal{Q}_m^{\mathrm{L}}$.

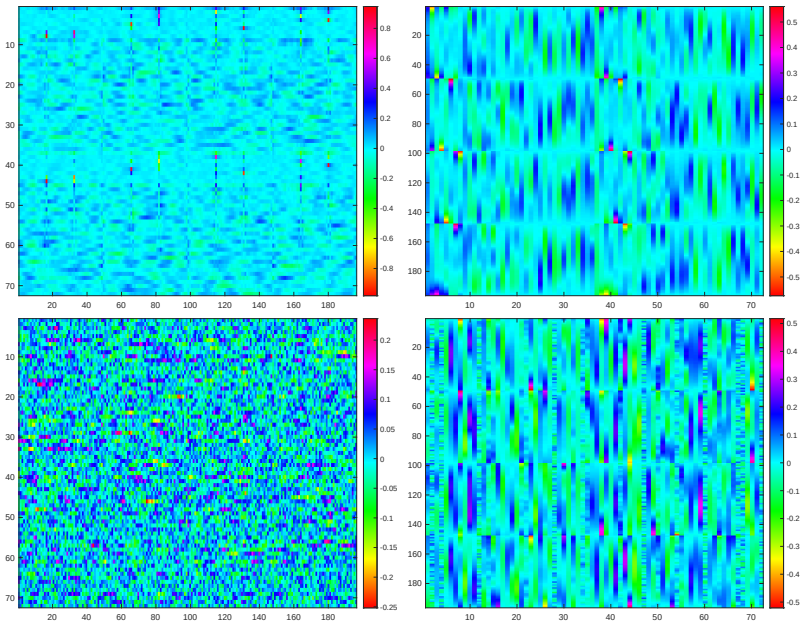


Figure 3.13: The first row shows the final weight matrices W_1 (left), W_2 (right) for SVD-initialized model on patch $m = (2, 2)$. The second row shows the weight matrices W_1 (left), W_2 (right) for randomly initialized model on patch $m = (2, 2)$. In both cases, training data is obtained by enlarging the patch.

Schwarz iteration: Online solutions

No.	Boundary condition
1	$\phi(x,0) = -\sin(2\pi x)$, $\phi(x,1) = \sin(2\pi x)$ $\phi(0,y) = \sin(2\pi y)$, $\phi(1,y) = -\sin(2\pi y)$
2	$\phi(x,0) = -\sin(4\pi x)$, $\phi(x,1) = \sin(4\pi x)$ $\phi(0,y) = \sin(4\pi y)$, $\phi(1,y) = -\sin(4\pi y)$
3	$\phi(x,0) = -1$, $\phi(x,1) = 1$ $\phi(0,y) = 2y^2 - 1$, $\phi(1,y) = 2y^2 - 1$

Table 3.5: Boundary conditions for p-Laplace equation (3.18) used in the global test.

Next, we apply the neural networks to the Schwarz iteration and show the global test performance. In Table 3.5 we list the boundary conditions for three problems used in the test. We use tolerance $\delta_0 = 10^{-4}$ in Algorithm 2 and use the full accuracy local solvers as in the generation of training data set. Figure 3.14 shows ground-truth solutions for different boundary conditions and the absolute error of u^{NN} obtained by neural network-based Schwarz iteration (plotted on a different scale). Error norms for the different methods can

be found in Table 3.6.

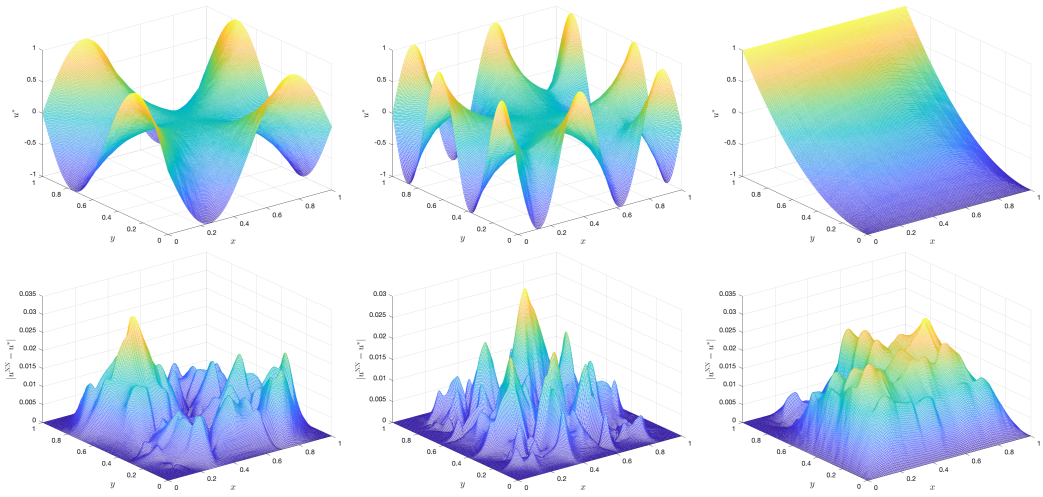


Figure 3.14: The first row shows the ground truth solution u^* for p-Laplace equation (3.18) for boundary condition 1 to 3 from left to right. The second row shows the absolute error $|u^{\mathrm{NN}} - u^*|$ for boundary condition 1 to 3 from left to right.

To demonstrate the efficiency of our method, we compare the CPU time of neural network based-Schwarz method and the classical Schwarz method with tolerance $\delta_0 = 10^{-4}$ in Algorithm 4. The NNs are trained using SVD initialization, with training data generated with buffer zones on the patches. The local solvers in the reference solution are chosen so that the local accuracy is at the same level as the NN-approximation, making for a fair comparison. The CPU time, number of iteration and error comparison can be found in Table 3.8. Compared with the classical Schwarz iteration, the reduced method updates local iterations much faster, while producing H^1 errors of the same order.

No. BC	1			2		
Relative Error	L^2	H^1	L^∞	L^2	H^1	L^∞
SVD-NN	**0.0199**	**0.0314**	**0.0324**	**0.0171**	**0.0250**	**0.0290**
SVD-NN (No buffer zone)	0.0935	0.1793	0.1398	0.0874	0.1052	0.1346
Rand-NN	0.0280	0.0400	0.0480	0.0260	0.0331	0.0367
Rand-NN (No buffer zone)	0.1062	0.1793	0.1412	0.0696	0.1023	0.1119
Linear	0.0623	0.1178	0.0909	0.0606	0.0990	0.0751

Table 3.6: Relative error for p-Laplace equation (3.18) by different methods.

No. BC	3		
Relative Error	L^2	H^1	L^∞
SVD-NN	**0.0215**	**0.0443**	**0.0311**
SVD-NN (No buffer zone)	0.1204	0.3331	0.2173
Rand-NN	0.0241	0.0578	0.0411
Rand-NN (No buffer zone)	0.2748	0.4380	0.3947
Linear	0.1390	0.1861	0.1624

Table 3.7: Relative error for p-Laplace equation (3.18) by different methods. (Continued)

Problem Number	1		2		3	
Method	NN	Classical	NN	Classical	NN	Classical
CPU time	**35.0**	87.8	**27.8**	68.3	**117.7**	302.2
Iteration	52	54	37	38	151	146
H^1 Error	0.0392	0.0231	0.0363	0.0256	0.0457	0.0124

Table 3.8: CPU time (s), number of iterations and the H^1 error of the classical Schwarz iteration and the neural network accelerated Schwarz iteration for p-Laplace equation (3.18).

3.4 Conclusion

We have presented a reduced-order neural network-based Schwarz method for multiscale nonlinear elliptic PDEs. In each iteration, the Schwarz method requires evaluation of a boundary-to-boundary map for each of the subdomains (patches). This map has high dimensional input and output spaces but is compressible due to the existence of a homogenization limit. A neural network can approximate high-dimensional maps using a number of parameters relaxed significantly from the dimension of data, and thus is a perfect fit to learn the boundary-to-boundary operator. Our method trains two-layer neural networks (with many fewer parameters than the input and output dimensions) to learn the boundary-to-boundary operators in an offline stage. In an online stage, the neural networks serve as surrogates of local solvers in the Schwarz iteration, leading to significant speedup over classical approaches. Our approach is illustrated with two examples: a semilinear elliptic equation and a p-Laplace equation.

Chapter 4

A Manifold Learning Approach for Numerical Homogenization

In this chapter, we describe an efficient domain decomposition-based framework for nonlinear multiscale PDE problems. The framework is inspired by manifold learning techniques and exploits the tangent spaces spanned by the nearest neighbors to compress local solution manifolds. In particular, our framework is applied to a semilinear elliptic equation with oscillatory media and a nonlinear radiative transfer equation. This new method does not rely on detailed analytical understanding of the multiscale PDEs, such as their asymptotic limits, and thus is more versatile for general multiscale problems.

4.1 Introduction

Homogenization is a body of theory and methods to study differential, or differential-integro equations with rapidly oscillating coefficients. It traces back to the famous work of Bensoussan-Lions-Papanicolaou [44], and builds on several other important developments [128, 90, 16, 89, 114, 35, 115, 1]. Generally speaking, the goal of homogenization is to derive asymptotic limiting equations as accurate surrogates of the original equations that do not have scale separations. The core technique is asymptotic analysis.

4.1.1 Goal

There are a number of famous examples that use homogenization techniques, such as elliptic equations with rapidly oscillating media [44], Schrödinger equation with small rescaled Planck constant [109], the neutron transport equation with small Knudsen number [45, 118], compressible Euler equation with small Mach number [149, 148, 194], and Boltzmann-type equations in the fluid regime [32]. All these examples have the form

$$\mathcal{N}^\varepsilon u^\varepsilon = f\,, \tag{4.1}$$

where $\mathcal{N}^\varepsilon$ is a partial differential operator that depends explicitly on the small parameter ε. The term f on the right-hand side represents the external information—the source terms, the boundary conditions, the initial conditions, and so on—which has no dependence on ε. Due to the ε-dependence of $\mathcal{N}^\varepsilon$, the PDE is rather stiff: the solutions either exhibit high oscillations (such as the Schrödinger equation with small value of the rescaled Planck constant, or the elliptic equation with rough media), or present boundary/initial layers within which solutions change rapidly (such as the Knudsen layer in kinetic systems). The oscillations and layers themselves usually do not carry any interesting physical information; one is more interested in extracting physically meaningful quantities from the solutions directly, with these details omitted. Thus, it is important to evaluate the limiting behavior of (4.1) as $\varepsilon \to 0$. There are two contrasting approaches in the literature that enable this task: One is analytical and the other is numerical.

The analytical approach seeks the asymptotic limit of the PDE (4.1), defined as follows:

$$\mathcal{N}^* u^* = f\,. \tag{4.2}$$

The term "asymptotic limit" refers to the fact that for any reasonable f, in a certain space with a certain metric, we have

$$\|u^\varepsilon - u^*\| \to 0 \quad \text{as } \varepsilon \to 0. \tag{4.3}$$

A classical way to derive this limit is to perform Hilbert expansion in terms of ε. Here, we define the ansatz

$$u^\varepsilon = u_0 + \varepsilon u_1 + \varepsilon^2 u_2 + \dots$$

and substitute into (4.1), then balance the two sides in terms of ε. Typically, at some level of the expansion, a closure is performed to derive the effective operator $\mathcal{N}^*$. This framework is highly effective and general; we will give explicit examples in later sections.

On the numerical side, we look for cheap solvers that compute the asymptotic limits. A typical requirement for classical numerical solvers to be accurate is that the discretization has to resolve the smallness of ε. This can lead to high numerical and memory cost, sometimes beyond reasonable computational resources. The focus of "numerical homogenization" or "asymptotic preserving" is thus to design schemes that capture asymptotic limits of the solutions with relaxed (and thus more efficient) discretization requirements. One technique is to explore analytical results and translate them to the discrete setting: The asymptotic limiting equations are derived first, and then a "macro solver" for the limiting equation and a "micro solver" that solves the original equation, are combined in some way. This strategy has been applied to deal with the Boltzmann-type equations, the Schrödinger equation, and the elliptic equations with highly oscillatory media, under the name of designing "asymptotic preserving" schemes, finding semi-classical limits, and performing "numerical homogenization." There is a significant drawback of this approach: The design of the numerical method is based completely on analytical understanding, so numerical development necessarily lags analytical progress. This fact significantly limits the role of multiscale computation.

This observation motivates the question that we address in this chapter. Given a system of the form (4.1), knowing it has an asymptotic limit (4.2) but not knowing the specific form of this limit, can we design an efficient, accurate solver?

4.1.2 Approach

In this chapter, we propose a numerical approach based on "compression." Classical methods require the use of $N_\varepsilon \sim \frac{1}{\varepsilon^\alpha}$ grid points to achieve accuracy and stability in solving (4.1), for some power $\alpha > 0$. Note that N_ε blows up to infinity as $\varepsilon \to 0$. By contrast, the limiting equation (4.2) is independent of ε, so we typically require only N_* grid points (a number that is independent of ε) to solve this system. Thus, the information carried in N_ε degrees of freedom can potentially be "compressed" into N_* degrees of freedom, provided that we can tolerate an asymptotic error of order ε in the solution (see (4.3)).

How can we design an approach to solving (4.1) that exploits compression? Our roadmap consists three steps: (a) identify the solution set that can be compressed; (b) compress the set into a smaller effective solution set; (c) for a given new data point $f(x)$, single out the solution from the effective set. We call the first two steps the offline stage, and the last step the online stage.

For linear equations, this roadmap has been followed by several authors in [52, 65, 64, 63]. When the setup is linear, the solution set is a space, and thus information is entirely coded in representative basis functions. These basis functions can be found in the offline stage, and a Galerkin formulation can then be used to identity the linear combination of the basis for a given $f(x)$ in the online stage. To find the representative basis functions, one can utilize the random sampling technique developed for finding low rank structures of matrices in [121], where the authors proved that a few random samples are able to reconstruct the low-rank column space with high probability; see [65].

In this chapter, we develop the roadmap in the nonlinear setting. The extension is not straightforward. Since the solution set is not a space in the nonlinear setup, the notion of "basis function" does not even exist. Instead, we seek an N_*-dimensional approximating manifold in an N_ε-dimensional space. For every given $f(x)$, there is a corresponding numerical solution u^ε to the original equation (4.1) in the N_ε space. Within ε distance there exists its homogenized solution u^* to the limiting equation (4.2). Since u^* relies

on only N_* degrees of freedom, as $f(x)$ varies, the variations of u^* form a manifold of dimension at most N_*.

By using this argument, we formulate the homogenization problem (in the nonlinear setting) into a manifold-learning problem: Suppose we can generate a few configurations of $f(x)$ and compute the associated numerical solutions, can we learn to represent the solution manifold? Further, given a completely new configuration of $f(x)$, can we quickly identify the corresponding solution? These two questions are addressed in the offline and online stages, respectively.

Many different approaches have been proposed for manifold learning based on observed point clouds. They typically look for key features that the points share, either locally (as in local linear embedding (LLE) [192], multi-scale SVD [9, 165], local tangent space alignment [219]), or globally (as in the use of heat kernels [81, 41]). The strategy we propose here is not a direct application of any one of these ideas, but it uses elements of the the local linear embedding and multi-scale SVD approaches. Specifically, we seek local linear approximations to the solution map, and cover the solution manifold with a number of these tangent space "patches."

We define the solution map as follows:

$$\mathcal{S}^\varepsilon : \quad f \in \mathcal{X} \to u^\varepsilon \in \mathcal{Y}\,. \tag{4.4}$$

It maps the source term and initial/boundary conditions captured in $f(x)$ to the solution of the equation (4.1). To find the solution manifold, we randomly sample a large number of configurations f_i in $\mathcal{X}$, and compute the solution $u_i^\varepsilon = \mathcal{S}^\varepsilon f_i \in \mathcal{Y}$ associated with each of these configurations. These solutions form a point cloud in a high dimensional space $\mathcal{Y}$. We subdivide the set of configurations $\{f_i\}$ into a number of small neighborhoods, and we look for the tangential approximation to the mapping (4.4) on each of these neighborhoods. Given a configuration f, we identify the neighborhood to which it belongs, and interpolate linearly to obtain the corresponding solution.

We summarize our online-offline strategy as follows. (Some modifications described in

Section 4.2 will reduce the cost of implementation.)

Offline: Randomly sample $f_i(x)$, $i = 1, \ldots, N$, and find solutions $u_i^\varepsilon = \mathcal{S}^\varepsilon f_i$;

Online: Given $f(x)$:

Step 1: Identify the k-nearest neighbors of $f(x)$, call them $f_{i_j}, j = 1, 2, \ldots, k$, with f_{i_1} being the nearest neighbor;

Step 2: Compute

$$\mathcal{S}^\varepsilon \phi \approx u_{i_1}^\varepsilon + \mathsf{U} \cdot c\,, \quad \text{with} \quad \mathsf{U} = \begin{bmatrix} | & & | \\ u_{i_2}^\varepsilon - u_{i_1}^\varepsilon & \cdots & u_{i_k}^\varepsilon - u_{i_1}^\varepsilon \\ | & & | \end{bmatrix},$$

where c is a set of coefficient that fits $f - f_{i_1}$ with a linear combination of $f_{i_j} - f_{i_1}$, for $j = 2, 3, \ldots, k$.

In **Step 2** we used the fact that the solution manifold if of low dimensional locally. To make the strategy mathematically precise, we need to address several questions, including the following.

- How should we sample $f_i(x)$ during the offline step?
- What metric should we use to quantify distance?
- Since computing each solution map $u_i^\varepsilon = \mathcal{S}^\varepsilon f_i$ is expensive, is there anyway to reduce the cost further?

We discuss these questions in the following sections. We stress that the manifold learning technique that we investigate in this chapter works best when the intrinsic dimensionality of the problem is significantly smaller than the typical required degrees of freedom, and this holds true for all homogenizable problems where the discretization of the limiting equation eliminates the ε dependence. For problems without ε dependence, and the dimension of the numerical solution is only moderately large, the approach that we take is not expected to reduce cost.

4.1.3 The layout of this chapter

We discuss the general recipe of the algorithm in Section 4.2, then show how the approach can be applied to two examples (a semi-linear elliptic equation and a nonlinear radiative transfer equation coupled with a temperature term) in Section 4.3 and Section 4.4, respectively. In both sections, we review the relevant homogenization theory for the equations, study the low rank structure of the tangential solution spaces, and present numerical evidence for the efficacy of our approach.

4.2 Framework

Our approach is a domain decomposition algorithm that makes use of Schwarz iteration.

After decomposing the domain into multiple overlapping patches, the Schwarz method solves the PDE in each patch, conditioned on agreement of solutions in the overlapping regions, which are boundary regions for the adjacent patches. At the initial step, these boundary conditions are unknown, so some initial guess is made. Subsequently, solution of PDEs on each patch alternates with updates of the solution on the overlapping regions, until convergence is obtained with respect to certain criteria. The cost of the entire process is determined by the number of iterations and the cost of the local solves, noting that, as with any domain decomposition method, the local solves can be performed in parallel. The approach is efficient when the local solves can be performed much more efficiently on the available computing resources than a solver that does not decompose the domain. The optimal domain partitioning depends on the conditioning of the problem and is often specific to the problem under study. Comprehensive descriptions of the Schwarz method appear in [197, 203].

This basic Schwarz iteration does not fully address the issue of ε-dependence that we discussed in Section 4.1, since local solvers still necessarily depend on ε. As a step toward making use of compression, we take the viewpoint that the purpose of the local solution step is to implement a boundary-to-boundary map, taking one part of the boundary conditions on a patch and using the solution of the resulting PDE to update the boundary

conditions for its neighboring patches. We propose to learn the boundary-to-boundary maps in an "offline" stage, by running the local solvers as many times as are needed to attain the desired accuracy in this map. This offline stage comes with a high overhead cost, but the computation is done only once, and we hope that the cost of the online stage is greatly reduced by having the boundary-to-boundary maps available. Note that this "offline" learning process is distinct from the **Offline** stage discussed in Section 1. With the application of domain decomposition, it is the local behavior that needs to be learned, instead of the full u^ε.

In the linear setting, building the boundary-to-boundary maps is quite straightforward. It amounts roughly to finding all discrete Green's functions, with the degree of freedom being determined by the number of grid points on the patch boundary, with one Green's function per grid point. In the nonlinear setting, the boundary-to-boundary map is nonlinear, so we can no longer build a linear basis, and we turn to manifold learning approach to approximate the map. Specifically, in the offline stage, we would sample randomly some configurations and find the corresponding image under the map. The resulting point cloud in high dimensional space can be viewed as samples of the manifold, which we can then learn by means of local approximate tangential planes. In the online stage, these tangential planes are used as surrogates to local boundary-to-boundary maps.

Before presenting details of the offline and online stage computations, we specify the setup and notation. We consider the following nonlinear PDE with Dirichlet boundary conditions in a domain $\Omega \subset \mathbb{R}^2$:

$$\begin{cases} \mathcal{N}^\varepsilon u^\varepsilon = 0\,, & \text{in } \Omega \\ u^\varepsilon = \phi, & \text{on } \partial\Omega, \end{cases} \tag{4.5}$$

where, as usual, ε indicates the small scale of the problem. For simplicity, we will assume throughout a square geometry $\Omega = [0, L]^2$. The domain Ω is decomposed into overlapping

rectangular patches defined by

$$\Omega = \bigcup_{m \in J} \Omega_m, \quad \text{with} \quad \Omega_m = (x_{m_1}^{(1)}, x_{m_1}^{(2)}) \times (y_{m_2}^{(1)}, y_{m_2}^{(2)}) \,, \tag{4.6}$$

where $m = (m_1, m_2)$ is a multi-index and J is the collection of the indices

$$J = \{m = (m_1, m_2) : \, m_1 = 1, \ldots, M_1, \, m_2 = 1, \ldots, M_2\} \,.$$

This setup is illustrated in Figure 2.1. For each patch we define the associated partition-of-unity function χ_m, which has $\chi_m(x) \geq 0$ and

$$\chi_m(x) = 0 \quad \text{on } x \in \Omega \setminus \Omega_m \,, \quad \sum_m \chi_m(x) = 1, \quad \forall x \in \Omega \,. \tag{4.7}$$

We set $\partial\Omega_m$ to be the boundary of patch Ω_m and denote by $\mathscr{N}(m)$ the collection of indices of the neighbors of Ω_m. In this particular 2D case, we have

$$\mathscr{N}(m) = \{(m_1 \pm 1, m_2)\} \cup \{(m_1, m_2 \pm 1)\} \subset J \,. \tag{4.8}$$

Assume that the equation (4.5) is well-posed, meaning that given ϕ in some function space $\mathcal{X}$, there exists a unique solution u^ε in another function space $\mathcal{Y}$. Assume further that the local nonlinear equation on patch Ω_m defined by

$$\begin{cases} \mathcal{N}^\varepsilon u_m^\varepsilon = 0 \,, & \text{in } \Omega_m \,, \\ u_m^\varepsilon = \phi_m, & \text{on } \partial\Omega_m \,, \end{cases}$$

is well-posed, given local boundary condition ϕ_m in some function space $\mathcal{X}_m$, and that the solution u_m^ε lives in space $\mathcal{Y}_m$. We further define the following operators.

- $\mathcal{S}_m^\varepsilon$ denotes the solution operator that maps local boundary condition ϕ_m to the local solution u_m^ε:

$$\mathcal{S}_m^\varepsilon : \mathcal{X}_m \to \mathcal{Y}_m, \quad \mathcal{S}_m^\varepsilon \phi_m = u_m^\varepsilon \,.$$

- $\mathcal{I}_m^l$ denotes the trace operator for all $l \in \mathcal{N}(m)$:

$$\mathcal{I}_m^l u_l^\varepsilon = u_l^\varepsilon|_{\partial\Omega_m \cap \Omega_l}, \quad l \in \mathcal{N}(m)\,,$$

  which takes the value of u_l^ε restricted on the boundary $\partial\Omega_m \cap \Omega_l$. Here we assume that the space $\mathcal{Y}_l$ allows for trace.

- $\mathcal{P}_m$ denotes the boundary update operator, mapping $\bigoplus_{l\in\mathcal{N}(m)} \mathcal{X}_l$ to $\mathcal{X}_m$

$$\mathcal{P}_m(\phi_l,\, l \in \mathcal{N}(m)) = \begin{cases} \mathcal{I}_m^l \mathcal{S}_l^\varepsilon \phi_l, & \text{on } \partial\Omega_m \cap \Omega_l, \quad l \in \mathcal{N}(m)\,, \\ \phi|_{\partial\Omega_m \cap \partial\Omega}, & \text{on } \partial\Omega_m \cap \partial\Omega\,. \end{cases}$$

  Note that on the points in $\partial\Omega_m \cap \partial\Omega$, the boundary condition from the whole domain Ω is imposed.

The offline and online stage of the algorithm are essentially to construct and to evaluate $\mathcal{P}_m$, as we now show.

4.2.1 Offline Stage

The goal of the offline stage is to construct a dictionary to approximate $\mathcal{P}_m$ for every $m \in J$. To eliminate any boundary layer effect, we enlarge each local patch slightly by adding a margin around its edges (except for the edges that correspond to part of the boundary of the whole domain). The enlarged domains are denoted by $\widetilde{\Omega}_m$ and illustrated in Figure 4.1.

We denote by $\widetilde{\mathcal{X}}_m$ the space of boundary conditions on $\partial\widetilde{\Omega}_m$ equipped with norm $\|\cdot\|$, and define a ball in $\widetilde{\mathcal{X}}_m$ as follows:

$$B(R_m;\, \widetilde{\mathcal{X}}_m) = \{\widetilde{\phi} \in \widetilde{\mathcal{X}}_m : \|\widetilde{\phi}\| \le R_m\}\,.$$

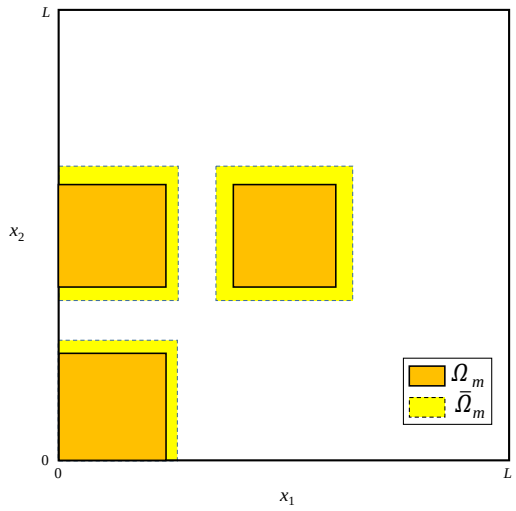


Figure 4.1: The plot demonstrates the use of local enlargement to damp boundary effects.

First, we draw N samples randomly from the ball, as follows:

$$\widetilde{\phi}_{m,i} \in B(R_m;\, \widetilde{\mathcal{X}}_m)\,, \quad i = 1, \ldots, N\,.$$

(The specific measure used in drawing depends on the particular problem being considered; we will make it more precise in the examples below.) For these samples we obtain local solutions $\widetilde{u}_{m,i}$ from the following PDEs:

$$\begin{cases} \mathcal{N}^\varepsilon \widetilde{u}^\varepsilon_{m,i} = 0\,, & \text{in } \widetilde{\Omega}_m \\ \widetilde{u}^\varepsilon_{m,i} = \widetilde{\phi}_{m,i}\,, & \text{on } \partial\widetilde{\Omega}_m \end{cases} \tag{4.9}$$

We build a dictionary from these solutions by confining them in the interior Ω_m and the boundary $\partial\Omega_m$:

$$\mathscr{I}_m = \{\psi_{m,i} = \widetilde{u}_{m,i}|_{\Omega_m}\}_{i=1}^N, \quad \mathscr{B}_m = \{\phi_{m,i} = \widetilde{u}_{m,i}|_{\partial\Omega_m}\}_{i=1}^N\,. \tag{4.10}$$

Since the problems that we consider are homogenizable, meaning that the solution manifold is of low dimensional, the value of N can be relatively small.

Remark 4.1. *Two remarks are in order.*

- *How to sample? That is, how to find a measure μ_m on $\widetilde{\mathcal{X}}_m$ for drawing samples? To make the setting more precise, we discretize the space $\widetilde{\mathcal{X}}_m$ to get $\widetilde{\mathcal{X}}_m^h$ equipped with norm $\|\cdot\|_h$, and define a measure μ_m^h on the ball $B(R_m; \widetilde{\mathcal{X}}_m^h)$. Denoting the dimension of $\widetilde{\mathcal{X}}_m$ by p, we sample the magnitude and the angle separately, that is, we take the measure as a product $\mu_m^h = \mu_{r,m} \otimes \mu_{S,m}$ with $\mu_{r,m}$ being the radial part on $(0, R_m)$ and $\mu_{S,m}$ being the measure on the unit sphere $S^{p-1} = \{\phi \in \widetilde{\mathcal{X}}_m^h : \|\phi\|_h = 1\} \subset \mathbb{R}^p$. The angular measure $\mu_{S,m}$ is chosen to be the uniform and the radial part $\mu_{r,m}$ has a density function $f(r) = \frac{D+1}{R_m^D} r^D$. The number D here plays the role of effective dimension; it should depend on the expected dimension of solution manifold. Note that if we take $D = p-1$, the measure μ_m^h is exactly the uniform measure on the full ball $B(R_m; \widetilde{\mathcal{X}}_m^h) \subset \mathbb{R}^p$. The question of selecting D in a rigorous way is left to future research. (See Appendix A.1 and Appendix A.2 for further details on this issue.)*

- *The physical boundary. To respect the boundary condition on $\partial\Omega$, the boundary patches Ω_m that touch the physical boundary need to be treated differently. For each sample $\widetilde{\phi}_{m,i}$, the physical boundary condition is enforced on the set $\partial\Omega_m \cap \partial\Omega$. Random sampling is done only on the remaining part of the patch boundary, that is, $\partial\Omega_m \setminus \partial\Omega$. See Appendix A.1 and Appendix A.2 for details.*

4.2.2 Online Stage

The online stage finds a particular solution u for given boundary data ϕ, based on information accumulated in the offline stage. This process is carried out through a Schwarz iteration to update local boundary conditions on each patch.

Denote by $\phi^{(n)} = [\dots, \phi_m^{(n)}, \dots]$ the collection of local boundary conditions at the nth iteration, with m being the patch index. At each iteration, we need to obtain $\phi_m^{(n+1)} = \mathcal{P}_m \phi^{(n)}$. For each $m \in J$, let $\phi_{m,i_q^{(n)}}$ be the q-th L^2-nearest neighbor of $\phi_m^{(n)}$ in $\mathscr{B}_m$, $q = 1, 2, \dots, k$. These neighbors, supported on $\partial\Omega_m$ lie (approximately) on a local tangential

plane centered at $\phi_{m,i_1^{(n)}}$:

$$\Phi_m^{(n)} = \begin{bmatrix} | & & | \\ \phi_{m,i_2^{(n)}} - \phi_{m,i_1^{(n)}} & \dots & \phi_{m,i_k^{(n)}} - \phi_{m,i_1^{(n)}} \\ | & & | \end{bmatrix} . \tag{4.11}$$

Associated with this plane, we also formulate the solution space centered around $\psi_{m,i_1^{(n)}}$:

$$\Psi_m^{(n)} = \begin{bmatrix} | & & | \\ \psi_{m,i_2^{(n)}} - \psi_{m,i_1^{(n)}} & \dots & \psi_{m,i_k^{(n)}} - \psi_{m,i_1^{(n)}} \\ | & & | \end{bmatrix} . \tag{4.12}$$

Locally, the map between these two planes is approximately linear, and thus to find $\phi_m^{(n+1)} = \mathcal{P}_m \phi^{(n)}$, we look for a linear interpolation of $\phi_m^{(n)}$ on $\Phi_m^{(n)}$, and map this interpolation to $\Psi_m^{(n)}$. More precisely, we look for $c_m^{(n)}$ that solves the least-squares problem

$$c_m^{(n)} = \text{argmin}_{v_m \in \mathbb{R}^{k-1}} \left\| \phi_m^{(n)} - \phi_{m,i_1^{(n)}} - \Phi_m^{(n)} v_m \right\|_{L^2(\partial\Omega_m)} , \tag{4.13}$$

and define the approximate solution to be:

$$u_m^{(n)} = \mathcal{S}_m^\varepsilon \phi_m^{(n)} \approx \psi_{m,i_1^{(n)}} + \Psi_m^{(n)} c_m^{(n)} . \tag{4.14}$$

To summarize: the map $\mathcal{P}_m \phi^{(n)}$ is a composition of $\mathcal{P}_m(\phi_l^{(n)},\, l \in \mathscr{N}(m))$ with $l \in \mathscr{N}(m)$, where

$$\phi_m^{(n+1)} = \mathcal{P}_m(\phi_l^{(n)},\, l \in \mathscr{N}(m)) = \mathcal{I}_m^l \mathcal{S}^\varepsilon{}_l \phi_l, \quad \text{on } \partial\Omega_m \cap \Omega_l . \tag{4.15}$$

Once a preset error tolerance is achieved, at some step n (usually because the local

boundary condition barely changes), the global solution is patched up as follows:

$$u^{(n)} = \sum_{m \in J} \chi_m u_m^{(n)}, \tag{4.16}$$

where $u_m^{(n)}$ is the local solution (4.14) and $\chi_m : \Omega \to \mathbb{R}$ is the smooth partition of unity associated with the partition.

We summarize the procedure in Algorithm 4.

Remark 4.2. *The Johnson-Lindenstrauss lemma [139] indicates that the search for d-dimensional k nearest neighbors in a data set of size N, with distance error δ, can be done in query time $O\Big(kd\frac{\log N}{\delta^2}\Big)$ and storage cost $N^{O(\log(1/\delta)/\delta^2)} + O\Big(d\big(N + \frac{\log N}{\delta^2}\big)\Big)$ [134, 12]. In addition, a cost of $O(k^2 d)$ is incurred at each iteration, due to L^2 minimization for each patch via QR factorization. In our setting, d is equal to the degrees of freedom on the boundary $\partial\Omega_m$.*

Remark 4.3. *To avoid notational complexity, the discussion above does not consider the physical boundary $\partial\Omega$. If a patch contains part of $\partial\Omega$, then that particular section of the patch is not updated. The true boundary condition ϕ is enforced in every iteration. The derivation is straightforward and is omitted from the discussion.*

4.3 Example 1: Semilinear elliptic equations with highly oscillatory media

In this section, we apply the methodology described above to solve semilinear elliptic equations. Semilinear elliptic equations with multiscale structures arise in a variety of situations, for instance, in nonlinear diffusion generated by nonlinear sources [142], and in the gravitational equilibrium of stars [162, 59]. As fundamental models in many areas of physics and engineering, the equations have received considerable attention.

Algorithm 4 Multiscale solver for nonlinear homogenizable equations (4.5).

Given the radius R_m, the number of nearest neighbors k, the tolerance δ and the initial guess of boundary conditions $\phi_m^{(0)}$ on each patch $m \in J$.
Domain Decomposition:
    Decompose Ω into overlapping patches: $\Omega = \bigcup_{m\in J} \Omega_m$, and enlarge each patch to obtain $\widetilde{\Omega}_m$.
Offline Stage: Prepare local dictionaries on interior patches Ω_m.
    Step 1: For each $m \in J_i$, generate N samples $\widetilde{\phi}_{m,i}$ from $B(R_m; \widetilde{\mathcal{X}}_m)$;
    Step 2: For all i, call function

$$\widetilde{u}_{m,i} = \text{LocPDESol}(\widetilde{\Omega}_m, \widetilde{\phi}_{m,i})\,;$$

    Step 3: Collect local dictionaries according to (4.10) for $\mathscr{B}_m$ and $\mathscr{I}_m$.
Online Stage: Schwarz iteration.
    while $\sum_m \|\phi_m^{(n)} - \phi_m^{(n-1)}\|_{L^2(\partial\Omega_m)} \geq \delta$ **do**
        for $m \in J$ **do**
            Search for k-nearest neighbors of $\phi_m^{(n)}$ in $\mathscr{B}_m$;
            Solve $c_m^{(n)}$ from the least-squares problem (4.13);
            Update $\phi_m^{(n+1)}$ by (4.15).
        end for
        $n \leftarrow n + 1$
    end while
return Global solution $u^{(n)}$ defined by (4.16).

function LocPDESol(Local domain Ω_m, Boundary condition ϕ_m)
    Perform the standard finite difference or finite element methods to solve the local nonlinear equation (4.9);
return Local solution u_m
end function

We consider the equation

$$\begin{cases} -\nabla_x \cdot \left(A\left(x, \frac{x}{\varepsilon}\right) \nabla_x u^\varepsilon\right) + f(u^\varepsilon) = 0, & x \in \Omega\,, \\ u^\varepsilon(x) = \phi(x), & x \in \partial\Omega\,. \end{cases} \tag{4.17}$$

The physical domain is $\Omega \subset \mathbb{R}^d$ with $d \geq 1$, and Dirichlet boundary condition is given as $\phi(x)$. The permeability $A(x,y) = (a_{ij}(x,y))_{d\times d} : \Omega \times \mathbb{R}^d \to \mathbb{R}^{d\times d}$ depends on both the slow variable x and the fast variable $y = x/\varepsilon$ and is highly oscillatory. The function $f : \mathbb{R} \to \mathbb{R}$ describes the nonlinear source term. The solution u^ε presents one component in a chemical reaction or one species of a biological system.

The well-posedness of equation (4.17) is classical. We assume that the permeability A is a symmetric matrix with L^∞-coefficients satisfying the standard coercivity condition, and that the nonlinear function f is locally Lipschitz continuous and increasing. Then, assuming the boundary $\partial\Omega$ is smooth enough, given boundary condition $\phi \in H^{1/2}(\partial\Omega) \cap L^\infty(\partial\Omega)$, the problem (4.17) has a unique H^1-solution satisfying the maximum principle. We refer to [111, 75] for details.

4.3.1 Homogenization limit

The semilinear elliptic equation (4.17) has a homogenization limit as $\varepsilon \to 0$. Supposing that $A(x,y)$ is smooth and periodic in y with period $I = [0,1]^d$, then as $\varepsilon \to 0$, the solution u^ε converges to a limit u^* that satisfies the same class of semilinear elliptic equations with an ε-independent effective permeability $A^*(x) = (a^*_{ij}(x))_{d\times d}$:

$$\begin{cases} -\nabla_x \cdot (A^*(x)\nabla_x u^*) + f(u^*) = 0, & x \in \Omega\,, \\ u^*(x) = \phi(x), & x \in \partial\Omega\,. \end{cases} \tag{4.18}$$

This equation (in particular, the effective permeability $A^*(x)$) can be derived by expanding the equation (4.17) into different orders of ε. Rigorous proofs are given in [44, 43, 182]. We cite the following theorem as a reference:

Theorem 4.1 (Section 16.3 in Chapter 1 of [44]; see also [8]). *Assume the boundary $\partial\Omega$ is smooth. Given $\phi(x) \in H^{1/2}(\partial\Omega) \cap L^\infty(\partial\Omega)$, let u^ε be the unique solution to the semilinear elliptic equation (4.17) in $H^1(\Omega) \cap L^\infty(\Omega)$. Assume that the permeability $A(x,y)$ is periodic in y with period $I = [0,1]^d$ and that $A(x,\cdot) \in C^1(I)$. Then the solution u^ε converges weakly in $H^1(\Omega)$ as $\varepsilon \to 0$ to u^* (the solution to (4.18)), where the permeability $A^*(x) = (a^*_{ij}(x))_{d\times d}$ is defined by*

$$a^*_{ij}(x) = \int_I \sum_{k,l} a_{kl}(x,y)(\delta_{ki} + \partial_{y_k}\chi_i)(\delta_{lj} + \partial_{y_l}\chi_j)\mathrm{d}y\,. \tag{4.19}$$

Here, for each fixed coordinate $j = 1, 2, \ldots, d$, the function $\chi_j(x,y)$ is the solution of the following cell problem with periodic boundary condition on I:

$$\nabla_y \cdot (A(x,y)\nabla_y(\chi_j(x,y) + y_j)) = 0\,. \tag{4.20}$$

To solve (4.17), the discretization has to resolve ε, but in the limit (4.18), the discretization is independent of ε. This suggests significant opportunities for cost savings: The information contained in $O(1/\varepsilon)$ degrees of freedom can be expressed with $O(1)$ degrees of freedom.

4.3.2 Low dimensionality of the tangent space

We now study the structure of the tangent space of the solution manifold, verifying in particular the low dimension assumption. We choose some point $\overline{u}^\varepsilon$ on the solution manifold and then randomly pick a neighboring solution point u^ε. These two points are solutions to (4.17) computed from distinct nearby boundary configurations $\overline{\phi}$ and ϕ, that is,

$$\overline{u}^\varepsilon|_{\partial\Omega} = \overline{\phi}\,, \quad u^\varepsilon|_{\partial\Omega} = \phi\,, \quad \text{with} \quad \|\overline{\phi} - \phi\|_{L^\infty(\partial\Omega)} = \mathcal{O}(\delta)\,. \tag{4.21}$$

By varying ϕ around $\overline{\phi}$, one can build a small point cloud around $\overline{u}^\varepsilon$. Denoting $\delta u^\varepsilon := u^\varepsilon - \overline{u}^\varepsilon$, we have immediately that

$$\begin{cases} -\nabla_x \cdot \left(A\left(x, \frac{x}{\varepsilon}\right) \nabla_x \delta u^\varepsilon\right) + f(\overline{u}^\varepsilon + \delta u^\varepsilon) - f(\overline{u}^\varepsilon) = 0, & x \in \Omega\,, \\ \delta u^\varepsilon(x) = \phi(x) - \overline{\phi}(x), & x \in \partial\Omega\,. \end{cases} \tag{4.22}$$

In the small-δ regime, this collection of solution differences δu^ε spans the tangent plane. We claim this tangent plane is low dimensional, so that it inherits the homogenization effect of the original equation. We have the following result.

Theorem 4.2. *Let δu^ε solve* (4.22)*. Assume $A(x,y) = (a_{ij}(x,y))_{d\times d}$ is periodic in y with period $I = [0,1]^d$. The equation has homogenization limit when $\varepsilon \to 0$, meaning there exists a limiting permeability $A^*(x) = (a^*_{ij}(x))_{d\times d}$, determined by $A(x,y)$ via equation* (4.19) *and* (4.20)*, so that $\delta u^\varepsilon \to \delta u^*$ and δu^* solves:*

$$\begin{cases} -\nabla_x \cdot (A^*(x)\nabla_x \delta u^*) + f(\overline{u}^* + \delta u^*) - f(\overline{u}^*) = 0, & x \in \Omega\,, \\ \delta u^*(x) = \phi(x) - \overline{\phi}(x), & x \in \partial\Omega\,, \end{cases} \tag{4.23}$$

where $\overline{u}^*$ solves:

$$\begin{cases} -\nabla_x \cdot (A^*(x)\nabla_x \overline{u}^*) + f(\overline{u}^*) = 0, & x \in \Omega\,, \\ u^*(x) = \overline{\phi}(x), & x \in \partial\Omega\,. \end{cases} \tag{4.24}$$

Further, for small δ, equation (4.23)*, in the leading order of δ, becomes:*

$$-\nabla_x \cdot (A^*(x)\nabla_x \delta u^*) + f'(\overline{u}^*(x))\delta u^* = 0\,. \tag{4.25}$$

Proof. By applying Theorem 4.1 to the equation for $\overline{u}^\varepsilon$, which is

$$\begin{cases} -\nabla_x \cdot \left(A\left(x, \frac{x}{\varepsilon}\right) \nabla_x \delta \overline{u}^\varepsilon\right) + f(\overline{u}^\varepsilon) = 0, & x \in \Omega\,, \\ \delta \overline{u}^\varepsilon(x) = \overline{\phi}(x), & x \in \partial\Omega\,, \end{cases}$$

we have by comparing with equation (4.17) for u^ε that $\overline{u}^\varepsilon$ converges weakly to $\overline{u}^*$, which

solves (4.24), and that u^ε converges weakly to u^*, which solves (4.18). From the definition $\delta u^\varepsilon = u^\varepsilon - \overline{u}^\varepsilon$, we find that δu^ε converges to δu^*, which solves (4.23). □

This theorem suggests that for the discretized equation, because of the existence of the homogenized limit, the tangent plane of the discrete solution is approximately low-rank. The space spanned by $\{\delta u^\varepsilon\}$ can be approximately spanned by $\{\delta u^*\}$, which solves the limiting equation (4.23) without dependence on small scales.

4.3.3 Implementation

We apply Algorithm 4 to equation (4.17) with $f(u) = u^3$ and $\Omega = [0, L]^2 \subset \mathbb{R}^2$, that is,

$$\begin{cases} -\nabla_x \cdot \left(a\left(x, \frac{x}{\varepsilon}\right) \nabla_x u\right) + u^3 = 0\,, & x \in \Omega = [0, L]^2, \\ u(x) = \phi(x), & x \in \partial\Omega. \end{cases} \tag{4.26}$$

We use the domain decomposition strategy of Section 4.2 to solve this system. Since Ω is convex and the coefficient $a(x, x/\varepsilon)$ belongs to $L^\infty(\Omega)$, we can show using the monotone method [10, 75] that the equation is well-posed, having a unique solution if we set

$$\mathcal{X} = H^{1/2}(\partial\Omega) \cap L^\infty(\partial\Omega), \quad \mathcal{Y} = H^1(\Omega) \cap L^\infty(\Omega)\,.$$

In the offline stage, we generate N samples for each enlarged patch $\widetilde{\Omega}_m$, as follows:

$$\widetilde{\phi}_{m,i} \in B(R_m;\, \widetilde{\mathcal{X}}_m)\,, \quad i = 1, \ldots, N\,.$$

(The measure we use for sampling is discussed in Appendix A.1.) We equip the ball with $H^{1/2}$-norm:

$$B(R_m;\, \widetilde{\mathcal{X}}_m) = \{\widetilde{\phi} \in \widetilde{\mathcal{X}}_m : \|\widetilde{\phi}\|_{H^{1/2}(\partial\Omega_m)} \le R_m\}\,. \tag{4.27}$$

We compute the $H^{1/2}(\partial\Omega)$-norm numerically using the Gagliardo seminorm [87]:

$$\|\phi\|_{H^{1/2}(\partial\Omega)} = \sqrt{\int_{\partial\Omega} |\phi(x)|^2 \mathrm{d}x + \iint_{\partial\Omega\times\partial\Omega} \frac{|\phi(x)-\phi(y)|^2}{|x-y|^2} \mathrm{d}x\mathrm{d}y}\,.$$

For these boundary configurations, we solve the equation

$$\begin{cases} -\nabla_x \cdot \left(a\left(x, \frac{x}{\varepsilon}\right) \nabla_x \widetilde{u}_{m,i}\right) + \widetilde{u}_{m,i}^3 = 0\,, & x \in \widetilde{\Omega}_m\,, \\ \widetilde{u}_{m,i}(x) = \widetilde{\phi}_{m,i}(x)\,, & x \in \partial\widetilde{\Omega}_m\,, \end{cases} \tag{4.28}$$

and build two sets of dictionaries by confining the solutions in the interior Ω_m and the boundary $\partial\Omega_m$, as follows:

$$\mathscr{I}_m = \{\psi_{m,i} = \widetilde{u}_{m,i}|_{\Omega_m}\}_{i=1}^N, \quad \mathscr{B}_m = \{\phi_{m,i} = \widetilde{u}_{m,i}|_{\partial\Omega_m}\}_{i=1}^N\,. \tag{4.29}$$

In the online stage, local boundary conditions are updated according to (4.14) at each iteration, with coefficients computed from (4.13). The local tangent space is found by searching for the k nearest neighbors in the dictionary $\mathscr{B}_m$, mapped to the dictionary $\mathscr{I}_m$ (see (4.29)). We use the L^2 norm to measure the distance between the newly generated solutions and the older solution set.

Once a preset error tolerance is achieved (at step n, say), the global solution is patched up from the local pieces, as follows:

$$u^{(n)} = \sum_{m\in J} \chi_m u_m^{(n)}\,, \tag{4.30}$$

where $u_m^{(n)}$ is the local solution on Ω_m at the n-th step and $\chi_m : \Omega \to \mathbb{R}$ is a smooth partition of unity.

4.3.4 Numerical Tests

We present numerical results for (4.26) in this subsection. We use $L = 1$, yielding the domain $\Omega = [0,1]^2$, and define the oscillatory media as follows:

$$a(x,y,x/\varepsilon,y/\varepsilon) = 2 + \sin(2\pi x)\cos(2\pi y) + \frac{2 + 1.8\sin(2\pi x/\varepsilon)}{2 + 1.8\cos(2\pi y/\varepsilon)} + \frac{2 + \sin(2\pi y/\varepsilon)}{2 + 1.8\cos(2\pi x/\varepsilon)}\,.$$

The boundary condition is

$$\begin{aligned} &\phi(x,0) = -\sin(2\pi x)\,, \quad \phi(x,1) = \sin(2\pi x)\,, \\ &\phi(0,y) = \sin(2\pi y)\,, \qquad \phi(1,y) = -\sin(2\pi y)\,. \end{aligned}$$

To form the partitioning, the whole domain Ω is divided equally into 4×4 non-overlapping squares, and then each square is enlarged by $\Delta x_{\rm o} = .0625$ on the sides that do not intersect with $\partial\Omega$, to create overlap. We thus have $M_1 = M_2 = 4$, with Ω_m for $m = (m_1, m_2)$, $m_1 = 1,2,3,4$ and $m_2 = 1,2,3,4$, defined by

$$\begin{aligned} \Omega_m = &\left[\max\left(\tfrac{m_1-1}{M_1} - \Delta x_{\rm o}, 0\right), \min\left(\tfrac{m_1}{M_1} + \Delta x_{\rm o}, 1\right)\right] \\ &\times \left[\max\left(\tfrac{m_2-1}{M_2} - \Delta x_{\rm o}, 0\right), \min\left(\tfrac{m_2}{M_2} + \Delta x_{\rm o}, 1\right)\right], \quad m = (m_1,m_2) \in J\,. \end{aligned}$$

Denote $\Omega_m = [x_m^{(1)}, x_m^{(2)}] \times [y_m^{(1)}, y_m^{(2)}]$. The partition of unity function χ_m is defined by normalizing the bump functions on the overlapping domains. More precisely, we first define a bump function $f_m : \Omega \to \mathbb{R}$ supported on Ω_m as follows:

$$f_m(x,y) = \begin{cases} \exp\left(-\frac{1}{1-|x-x_m|/\alpha_m} - \frac{1}{1-|y-y_m|/\beta_m}\right), & (x,y) \in \Omega_m \\ 0\,, \quad \text{Otherwise} \end{cases},$$

where $x_m = \frac{x_m^{(2)} - x_m^{(1)}}{2}$, $y_m = \frac{y_m^{(2)} - y_m^{(1)}}{2}$, $\alpha_m = \frac{x_m^{(1)} + x_m^{(2)}}{2}$ and $\beta_m = \frac{y_m^{(1)} + y_m^{(2)}}{2}$. The partition of

unity $\chi_m : \Omega \to \mathbb{R}$ is then obtained by

$$\chi_m(x,y) = \frac{f_m(x,y)}{\sum_{m\in J} f_m(x,y)}\,.$$

A standard finite-volume scheme with uniform grid is used for discretization, the corresponding nonlinear discrete system being solved by Newton's method. The reference solutions are computed on the fine mesh with $h = 2^{-12} = \frac{1}{4096}$. Unless otherwise specified, other computations are performed with mesh size $h = 2^{-9} = \frac{1}{512}$. Denoting the numerical solution by $u_{ij} \approx u(x_i, y_j)$, we use the classical discrete L^2 norm

$$\|u\|_{L^2} = h\sqrt{\sum_{i,j=0}^{p} |u_{ij}|^2}\,,$$

and the energy norm

$$\|u\|_{\mathcal{E}} = h\sqrt{\sum_{j=0}^{p}\sum_{i=0}^{p-1} a_{i+1/2,j}\left|\frac{u_{i+1,j}-u_{ij}}{h}\right|^2 + \sum_{i=0}^{p}\sum_{j=0}^{p-1} a_{i,j+1/2}\left|\frac{u_{i,j+1}-u_{ij}}{h}\right|^2}\,,$$

and define the relative errors accordingly by

$$\text{relative } L^2 \text{ error} = \frac{\|u_{\text{ref}} - u_{\text{approx}}\|_{L^2}}{\|u_{\text{ref}}\|_{L^2}}\,, \quad \text{relative energy error} = \frac{\|u_{\text{ref}} - u_{\text{approx}}\|_{\mathcal{E}}}{\|u_{\text{ref}}\|_{\mathcal{E}}}.$$

We first describe numerical experience with the offline stage. Each interior patch Ω_m is enlarged by a margin Δx_{b} to damp the boundary effects. The resulting buffered patch $\widetilde{\Omega}_m$ is concentric with Ω_m; see Figure 4.2. In the plots shown below, we study the patch indexed by $m = (2, 2)$.

To build the local dictionary, we generate 64 samples randomly in $B(R_{2,2}, \widetilde{X}_{2,2})$, where $R_{2,2} = 20$. (The sampling scheme is discussed in Appendix A.1.) We compute the local solutions with these boundary conditions on $\widetilde{\Omega}_{2,2}$, for several choices of buffer size Δx_{b}, and subtract the solutions from the reference solution, confined to $\Omega_{2,2}$. This procedure forms the tangent space centered around the reference solution in this particular patch. In

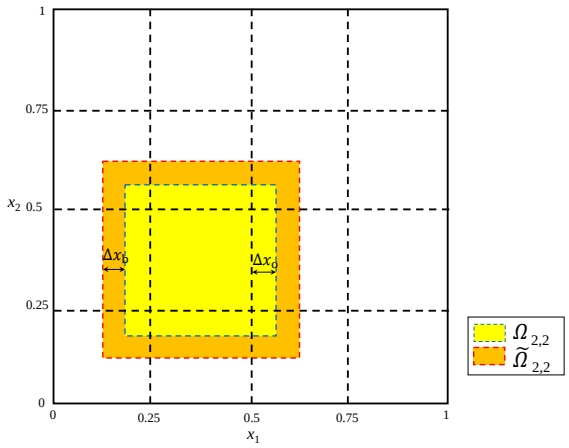


Figure 4.2: Buffered domain decomposition.

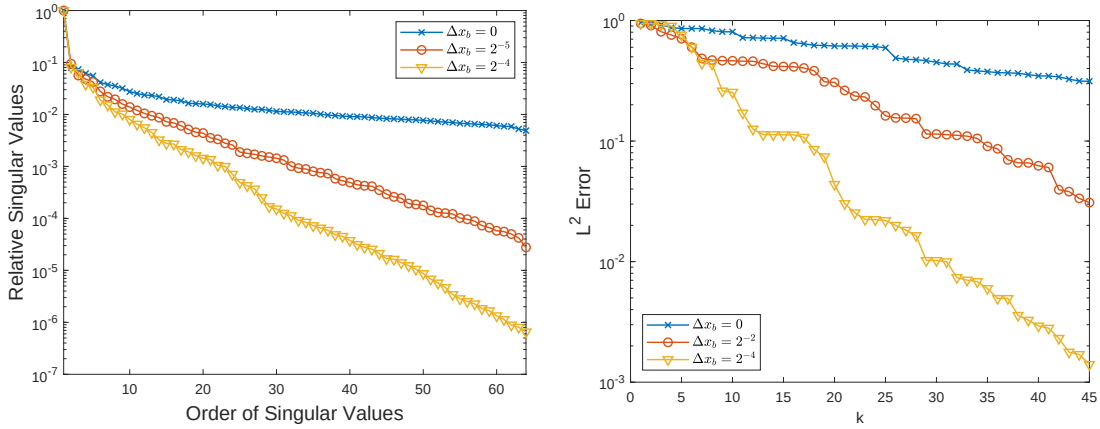
Figure 4.3a we plot the singular value decay of this tangent space, for $\varepsilon = 2^{-4}$. It is clear that the singular values decay exponentially, with a larger buffer margin Δx_{b} leading to a faster decay rate. This observation suggests that the tangent space is approximately low dimensional. We then project the reference solution onto the space spanned by its closest neighbors. As the number of neighbors increases, the relative error decays exponentially, as seen in Figure 4.3b. When the buffer margin is $\Delta x_{\mathrm{b}} = 2^{-4}$, we achieve 99% accuracy with 30 neighbors. By comparison, the degrees of freedom for this patch is determined by the total number of grid points on the boundary of this patch — 768, in this particular case.

In the online stage, we set the stopping criterion to be

$$\sum_m \|\phi_m^{(n)} - \phi_m^{(n-1)}\|_{L^2(\partial\Omega_m)} < 10^{-5}\,,$$

where the upper index (n) indicates the evaluation of the solution in the n-th iteration on $\mathcal{X}_m$, which is the boundary of Ω_m. The initial guess for all local boundary condition is chosen (trivially) to be $\phi_m^{(0)}|_{\partial\Omega_m \setminus \partial\Omega} = 0$ and $\phi_m^{(0)}|_{\partial\Omega_m \cap \partial\Omega} = \phi|_{\partial\Omega_m \cap \partial\Omega}$.

In Figure 4.4, we compare the numerical solutions using the space spanned by $k = 5$ and $k = 40$ nearest neighbors. The buffer margin is $\Delta x_{\mathrm{b}} = 2^{-4}$, and we set $\varepsilon = 2^{-4}$. We also document the error behavior as a function of k, ε, and Δx_{b}. In Figure 4.5, we plot



(a) (b)

Figure 4.3: (a) The singular value decay of the tangent space (centered around the reference solution) on patch $\Omega_{2,2}$, for different values of the buffer margin Δx_{b}. (b) The relative error of the projection of the reference solution onto the space spanned by the nearest k neighbors on $\Omega_{2,2}$. The distance is measured in $L^2(\Omega_{2,2})$. $\varepsilon = 2^{-4}$ in both plots.

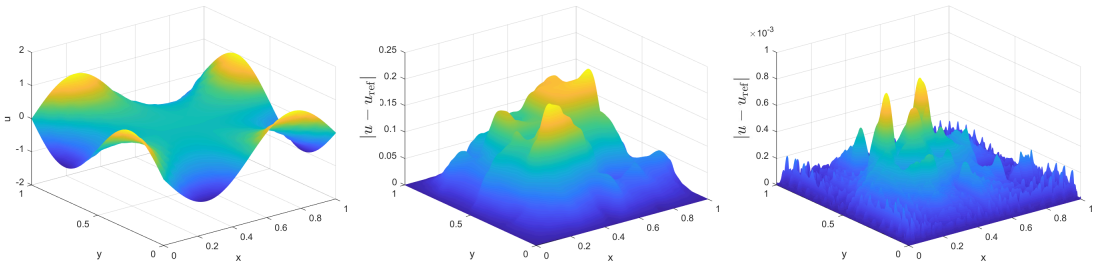


Figure 4.4: Computed solutions. Left panel shows the reference solution obtained with fine grids of width $h = 2^{-12}$. Middle and right panels show the numerical error $|u - u_{\mathrm{ref}}|$ obtained with $k = 5$ and $k = 30$, respectively.

the error decay as a function of k (the number of neighbors used in the online stage) for different values of ε and Δx_{b}. The decay is independent of ε, indicating the rank structure is not influenced by small scales in the equation. As the number of neighbors k increases, the global relative L^2 and energy error decays exponentially provided a buffer zone is present. When $\Delta x_{\mathrm{b}} = 0$ (no buffer), the boundary layer effect is strong, and convergence is not obtained, meaning that the local solution cannot be well approximated from the dictionary.

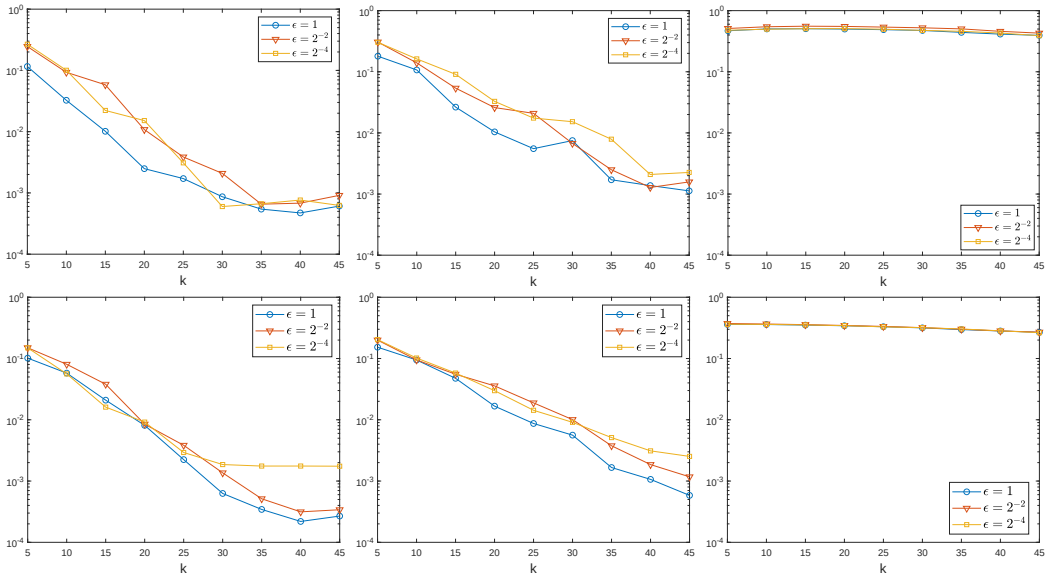


Figure 4.5: The top row of plots shows the global L^2 error as a function of k with different ε and buffer zone size Δx_{b}. The bottom row of plots shows the global energy error. The three columns of plots represent $\Delta x_{\mathrm{b}} = 2^{-4}, 2^{-5}, 0$, respectively.

We show CPU times in Table 4.1, comparing the reduced model for different values of k with the classical Schwarz iteration, for $\varepsilon = 2^{-4}$ and $\Delta x_{\mathrm{b}} = 2^{-4}$. The same stopping criterion is used for all variants. The online stage of each reduced model is significantly faster than the classical Schwarz iteration. Even with $k = 40$ neighbors involved in the local solution reconstruction, our method requires 1.12s, compared to 187.8s required by the classical Schwarz method. While the offline preparation is expensive in general, it is still cheaper in this example than the classical Schwarz iteration for solving a single problem. Because the dictionary can be reused, our method has a strong advantage in situations

	CPU Time (s) ($\varepsilon = 2^{-4}$)	
	offline	online
Reduced model $k = 5$	135.6914	0.173712
Reduced model $k = 10$		0.305707
Reduced model $k = 20$		0.462857
Reduced model $k = 30$		0.696785
Reduced model $k = 40$		1.124082
Classical Schwarz	—	187.7705

Table 4.1: CPU time comparison between our reduced method with $k = 5, 10, 20, 30, 40$ and classical Schwarz method.

where many solutions corresponding to different boundary conditions are needed. This is a typical situation in inverse problems, where to determine the unknown media, many boundary configurations are imposed and numerical solutions are computed to compare with measurements [60].

4.4 Example 2: Nonlinear radiative transfer equation

Here we study the application of Algorithm 4 to a nonlinear radiative transfer equation. Radiative transfer is the physical phenomenon of energy transfer in the form of electromagnetic radiation, and the radiative transfer equations describe the absorption or scattering of radiation as it propagates through a medium. The equations are important in optics, astrophysics, atmospheric science, remote sensing [172], and other applications.

We denote by $I^\varepsilon(x, v)$ the distribution function of photon particles at location x moving with velocity v in the physical domain $\mathcal{D} \subset \mathbb{R}^3$ and the velocity domain $\mathcal{V} = \mathbb{S}^2$. Also denote by $T^\varepsilon(x)$ the temperature profile across domain $\mathcal{D}$. We consider a nonlinear system of equations that couples the photon particle distribution with the temperature profile. The steady state equations are

$$\begin{cases} \varepsilon v \cdot \nabla_x I^\varepsilon = B(T^\varepsilon) - I^\varepsilon, \ \text{ for } (x, v) \in \mathcal{K} = \mathcal{D} \times \mathcal{V}, \\ \varepsilon^2 \Delta_x T^\varepsilon = B(T^\varepsilon) - \langle I^\varepsilon \rangle, \ \text{ for } x \in \mathcal{D}\,, \end{cases} \tag{4.31}$$

with the velocity-averaged intensity given by

$$\langle I\rangle(x) = \int_{\mathcal{V}} I(x,v)\mathrm{d}\mu(v)\,. \tag{4.32}$$

Here, $\mu(v)$ is a normalized uniform measure on $\mathcal{V}$ and $B(T)$ is a nonlinear function of T, typically defined as

$$B(T) = \sigma T^4, \tag{4.33}$$

where σ is a scattering coefficient [151, 207]. The parameter ε is called the Knudsen number, standing for the ratio of the mean free path and the typical domain length. When the medium is highly scattering and optically thick, the mean free path is small, with $\varepsilon \ll 1$. The scattering coefficient σ is independent of ε.

We consider a slab geometry. Assuming the y and z directions to be homogeneous, then since $v = (\cos\theta, \sin\theta\sin\varphi, \sin\theta\cos\varphi)$, the v_x component becomes $\cos\theta \in [-1,1]$. The problem is simplified to:

$$\begin{cases} \varepsilon v\partial_x I^\varepsilon = B(T^\varepsilon) - I^\varepsilon \\ \varepsilon^2\partial_x^2 T^\varepsilon = B(T^\varepsilon) - \langle I^\varepsilon\rangle \end{cases}, \quad (x,v) \in \mathcal{K} = [a,b]\times[-1,1]\,, \tag{4.34}$$

with $\langle I\rangle(x) = \frac{1}{2}\int_{-1}^{1} I(x,v)\mathrm{d}v$.

We provide incoming boundary conditions that specify the distribution of photons entering the domain. The boundary condition itself has no ε dependence; we have

$$I^\varepsilon(x,v) = I_b(x,v) \quad \text{on } \Gamma_-\,, \quad T^\varepsilon(x) = T_b(x) \quad \text{on } \partial\mathcal{D}\,. \tag{4.35}$$

Here $\Gamma_\pm$ collect the coordinates at the boundary with velocity pointing into or out of the domain:

$$\Gamma_\pm = \{(x,v) : x \in \partial\mathcal{D}\,, \pm v\cdot n_x > 0\}\,,$$

and n_x denotes the unit outer normal vector at $x \in \partial\Omega$.

4.4.1 Homogenization limit

The equations (4.31) have a homogenization limit. As $\varepsilon \to 0$, the right hand side of the equations dominates, and by balancing the scales we obtain

$$I^\varepsilon \sim \langle I^\varepsilon \rangle \sim \sigma (T^\varepsilon)^4 \sim I^* \sim \sigma (T^*)^4 \,.$$

To find the equation satisfied by T^*, we expand the equations (4.31) up to second order in ε. Rigorous results are shown in [156, 150, 33, 36]. We cite the following theorem captures the results needed here.

Theorem 4.3 (Modification of Theorem 3.2 in [150])**.** *Let $\mathcal{D} \subset \mathbb{R}^3$ be bounded and $\partial\mathcal{D}$ be smooth. Assuming that the boundary conditions* (4.35) *are positive and that $T_b \in H^{1/2}(\partial\mathcal{D}) \cap L^\infty(\partial\mathcal{D})$ and $I_b \in L^\infty(\Gamma_-)$, then the nonlinear radiative transfer equation* (4.31) *has a unique positive solution $(I^\varepsilon, T^\varepsilon) \in L^\infty(\mathcal{K}) \times L^\infty(\mathcal{D})$. If we assume further that $(I_b, T_b) \geq \gamma > 0$ and $I_b = B(T_b)$ a.e. on Γ_-, then the solution in the limit as $\varepsilon \to 0$ converges weakly to $(B(T^*), T^*)$, where the limiting temperature T^* is the unique positive solution to the following PDE:*

$$\Delta_x (T^* + B(T^*)/3) = 0 \,, \quad \textit{for} \quad x \in \mathcal{D} \,, \tag{4.36}$$

equipped with Dirichlet boundary data $T^*|_{\partial\mathcal{D}} = T_b$. The convergence of T^ε is in $H^1(\mathcal{D})$ weak and the convergence of I^ε is in $L^\infty(\mathcal{K})$ weak-$*$.

Remark 4.4. *Without appropriate boundary conditions $I_b = B(T_b)$, boundary layers of width $O(\varepsilon)$ may emerge as $\varepsilon \to 0$. It is conjectured in [150] that the boundary layers in the neighborhood of each point $\hat{x} \in \partial\mathcal{D}$ can be characterized by the following one-dimensional*

Milne problem for $y \in [0, \infty)$:

$$
\begin{aligned}
-(v \cdot n_{\hat{x}})\partial_y \hat{I} &= B(\hat{T}) - \hat{I}\,, \\
\partial_y^2 \hat{T} &= B(\hat{T}) - \langle \hat{I} \rangle\,, \\
\hat{T}(0) = T_b(\hat{x}), \quad \hat{I}(0, v) &= I_b(\hat{x}, v), \quad \textit{for } v \cdot n_{\hat{x}} < 0\,,
\end{aligned}
$$

where $y = \frac{(x-\hat{x})\cdot n_{\hat{x}}}{\varepsilon}$ represents a rescaling of the layer. The solutions that are bounded at infinity are used to form the Dirichlet boundary conditions for (4.36)*: At the limit as $y \to \infty$, $B(\hat{T}) = \langle \hat{I} \rangle = \hat{I}$, and one uses $T(\hat{x}) = \hat{T}(\infty)$.*

According to Theorem 4.3, in the zero limit of ε, I^ε loses its velocity dependence and is proportional to $(T^\varepsilon)^4$ that satisfies a semi-linear elliptic equation. Since the information in the velocity domain is lost, we expect low dimensionality of the (discretized) solution set. For the slab problem for RTE (4.34), the number of grid points needed for a satisfactory numerical result is $N_x N_v$, with both N_x and N_v scaling as $O(\frac{1}{\varepsilon})$ for numerical accuracy. Thus, for every given configuration of boundary conditions, the numerical solution is one data point in an $N_x N_v$-dimensional space — a space of very high dimension. However, when ε is small, the solutions are approximately given by the limiting elliptic equation (4.36) and the number of grid points needed is a number N_x^* that has no dependence on ε. This implies that the point clouds in the $O(1/\varepsilon^2)$-dimensional space can be essentially represented using $O(1)$ degrees of freedom: The solution manifold is approximately low dimensional. (Savings are even greater for problems with higher physical / velocity dimension.)

The use of a limiting equation to speed up the computation of kinetic equations is not new. For Boltzmann-type equations (for which RTE serves as a typical example), one is interested in designing algorithms that automatically reconstruct the limiting solutions with low computational cost. The algorithms that achieve this property are called "asymptotic-preserving" (AP) methods [84, 138], because the asymptotic limits are preserved automatically. There are many successful examples of AP schemes, but most of

them depend strongly on the analytical understanding of the limiting equation. The solver of the limiting equation is built into the Boltzmann solver, to drag the numerical solution to its macroscopic description [155, 88, 86, 131]. Such a design scheme limits the application of AP methods significantly. Many kinetic equations have unknown limiting behavior, making the use of AP designs impossible. By contrast, Algorithm 4 does not rely on any explicit information of the limiting equation, and is able to deal with general kinetic equations with small scales.

4.4.2 Low dimensionality of the tangent space

As for the example of Section 4.3, we start by studying some basic properties of the local solution manifold and its tangential plane.

We first randomly pick a point $(\overline{I}^{\varepsilon}, \overline{T}^{\varepsilon})$ on the solution manifold around which to perform tangential approximation. Nearby points $(I^{\varepsilon}, T^{\varepsilon})$ are obtained by solutions to the RTE (4.34) with respect to perturbed boundary conditions. The boundary conditions for $(\overline{I}^{\varepsilon}, \overline{T}^{\varepsilon})$ and $(I^{\varepsilon}, T^{\varepsilon})$, respectively, are

$$(\overline{I}^{\varepsilon}|_{\Gamma_-}, \overline{T}^{\varepsilon}|_{\partial\mathcal{D}}) = (\overline{I}_b, \overline{T}_b), \quad (I^{\varepsilon}|_{\Gamma_-}, T^{\varepsilon}|_{\partial\mathcal{D}}) = (I_b, T_b), \tag{4.37}$$

and we assume close proximity, in the sense that

$$\|\overline{I}_b - I_b\|_{L^2(\Gamma_{m,-})} + \|\overline{T}_b - T_b\|_2 = O(\delta). \tag{4.38}$$

Using the notation $\delta I^{\varepsilon} := I^{\varepsilon} - \overline{I}^{\varepsilon}$ and $\delta T^{\varepsilon} := T^{\varepsilon} - \overline{T}^{\varepsilon}$ for the difference of the two solutions, we find that this difference satisfies the equations

$$\begin{cases} \varepsilon v \partial_x \delta I^{\varepsilon} = B(\overline{T}^{\varepsilon} + \delta T^{\varepsilon}) - B(\overline{T}^{\varepsilon}) - \delta I^{\varepsilon}, \\ \varepsilon^2 \partial_x^2 \delta T^{\varepsilon} = B(\overline{T}^{\varepsilon} + \delta T^{\varepsilon}) - B(\overline{T}^{\varepsilon}) - \langle \delta I^{\varepsilon} \rangle, \end{cases} \tag{4.39}$$

with boundary conditions:

$$\delta I^\varepsilon|_{\Gamma_-} = \overline{I}_b - I_b\,, \quad \delta T^\varepsilon|_{\partial\mathcal{D}} = \overline{T}_b - T_b\,.$$

By varying I_b and T_b (subject to (4.38)), we obtain a list of solutions $(\delta I^\varepsilon\,, \delta T^\varepsilon)$ that spans the tangent plane of the solution manifold surrounding $(\overline{I}^\varepsilon\,, \overline{T}^\varepsilon)$. It will be shown below that this plane is low dimensional. We have the following result.

Theorem 4.4. *Let $(\delta I^\varepsilon, \delta T^\varepsilon)$ solve (4.39). As $\varepsilon \to 0$, we have $(\delta I^\varepsilon, \delta T^\varepsilon) \to (\delta I^*, \delta T^*)$ so that $\delta I^* = \langle \delta I^* \rangle = B(\overline{T}^* + \delta T^*) - B(\overline{T}^*)$ and δT^* solves:*

$$\partial_x^2 \left[\delta T^* + \tfrac{1}{3} B(\overline{T}^* + \delta T^*) - \tfrac{1}{3} B(\overline{T}^*) \right] = 0\,. \tag{4.40}$$

Here the reference state $\overline{T}^*$ solves:

$$\partial_x^2 \left[\overline{T}^* + \tfrac{1}{3} B(\overline{T}^*) \right] = 0\,. \tag{4.41}$$

Both equations are equipped with appropriate Dirichlet type boundary conditions. Furthermore, for small δ, the leading order equation is

$$\Delta_x \left[\left(1 + \tfrac{1}{3} B'(\overline{T}^*) \right) \delta T^* \right] = 0\,. \tag{4.42}$$

Proof. Apply Theorem 4.3 (in one dimension) to the equation for $(\overline{I}^\varepsilon\,, \overline{T}^\varepsilon)$ to obtain

$$\begin{cases} \varepsilon v \partial_x \overline{I}^\varepsilon = B(\overline{I}^\varepsilon) - \overline{I}^\varepsilon \\ \varepsilon^2 \partial_x^2 \overline{T}^\varepsilon = B(\overline{I}^\varepsilon) - \langle \overline{I}^\varepsilon \rangle, \end{cases}$$

and the equation (4.34) for $(I^\varepsilon\,, T^\varepsilon)$. Together, these equations show that $(\overline{I}^\varepsilon\,, \overline{T}^\varepsilon)$ converges weakly to $(\overline{I}^*\,, \overline{T}^*)$ that solves (4.41), and also that $(I^\varepsilon\,, T^\varepsilon)$ converges weakly to $(I^*\,, T^*)$ that solves (4.36). Taking the difference for $(\overline{I}^\varepsilon\,, \overline{T}^\varepsilon)$ and $(I^\varepsilon\,, T^\varepsilon)$ we find that $(\delta I^\varepsilon, \delta T^\varepsilon)$ converges to $(\delta I^*, \delta T^*)$, which solves (4.40). □

In one dimension, the elliptic problem only has two degrees of freedom, determined by the two Dirichlet boundary conditions. This suggests that in the limit as $\varepsilon \to 0$, for relatively small δ, the tangent plane spanned by $(\delta I^\varepsilon, \delta T^\varepsilon)$ is asymptotically two-dimensional, and is parameterized by the two boundary conditions for δT^ε. (A similar reduction holds in higher dimensions, but we leave the implementation to future work.)

4.4.3 Implementation of the algorithm

In RTE, the domain setup needs some extra care, and we need to re-perform partitioning. The physical boundaries are no longer the boundaries on which the Dirichlet conditions are imposed, and the general framework in Algorithm 4.2 for PDE with Dirichlet boundary condition on the physical boundaries has to be changed accordingly. For the $(1+1)$D case, we set

$$\mathcal{K} = \mathcal{D} \times \mathcal{V} = [0, L] \times [-1, 1]\,; \text{ then } \Gamma_- = \{(0, v) : v > 0\} \cup \{(1, v) : v < 0\}\,,$$

with boundary conditions

$$I^\varepsilon|_{\Gamma_-} = g = (g^{(1)}(0, \cdot), g^{(2)}(L, \cdot)), \quad T^\varepsilon(0) = \theta^{(1)}, \quad T^\varepsilon(L) = \theta^{(2)}\,,$$

where $g^{(1)}$ is supported only on $v > 0$ while $g^{(2)}$ is supported only on $v < 0$. For notational simplicity, we write

$$u := (I^\varepsilon, T^\varepsilon)\,, \quad u|_{\Gamma_-} := \phi = (g^{(1)}(0, \cdot), g^{(2)}(L, \cdot), \theta^{(1)}, \theta^{(2)})\,.$$

To partition the domain, we divide $\mathcal{K}$ into M overlapping patches:

$$\mathcal{K} = \bigcup_{m=1}^{M} \mathcal{K}_m, \quad \text{with} \quad \mathcal{K}_m = \mathcal{D}_m \times \mathcal{V} = [t_m, s_m] \times [-1, 1], \tag{4.43}$$

where t_m and s_m are left and right boundaries for the m-th patch, satisfying

$$0 = t_1 < t_2 < s_1 < t_3 < \cdots < s_{M-2} < t_M < s_{M-1} < s_M = L\,.$$

The size of the mth patch in x direction is denoted as $d_m = t_m - s_m$. For each patch, we define the local incoming boundary coordinates as follows:

$$\Gamma_{m,-} = \{(t_m, v) : v > 0\} \cup \{(s_m, v) : v < 0\}\,. \tag{4.44}$$

See Figure 4.6 for an illustration of the configuration.

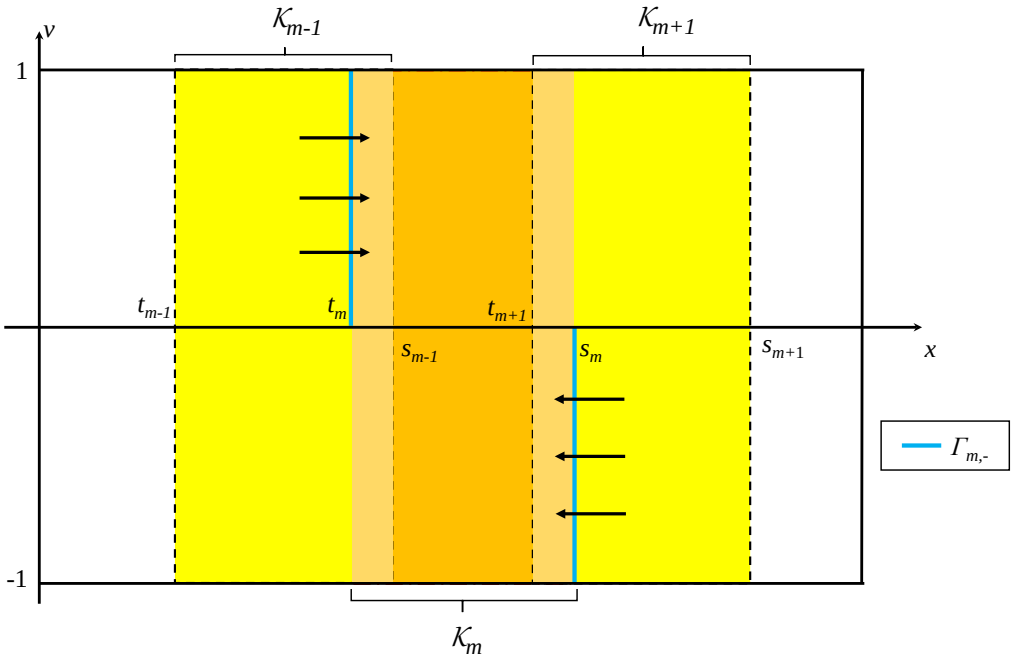


Figure 4.6: Domain decomposition for nonlinear RTE and the incoming boundary of the local patch.

In this particular setup, according to [156], if ϕ is in the space

$$\mathcal{X} = L^2(\Gamma_-) \times \mathbb{R}^2_+ = \left\{ \left(g, \theta^{(1)}, \theta^{(2)} \right) \mid g \in L^2(\Gamma_-);\ \theta^{(1)}, \theta^{(2)} \geq 0 \right\}\,,$$

then there exists a unique positive solution in the space

$$\mathcal{Y} = H_2^1(\mathcal{K}) \times H^1(\mathcal{D}) = \left\{ (I,T) \mid I \in H_2^1(\mathcal{K}), T \in H^1(\mathcal{D}) \right\} ,$$

where $H_2^1(\mathcal{K})$ is the space of functions for which the following norm is finite:

$$\|I\|_{H_2^1(\mathcal{K})} = \|I\|_{L^2(\mathcal{K})} + \|v\partial_x I\|_{L^2(\mathcal{K})} .$$

Note that the trace operators $T_\pm u = u|_{\Gamma_\pm}$ are well-defined maps from $H_2^1(\mathcal{K})$ to $L^2(\Gamma_\pm)$ (see, for example [6]).

To proceed, we define several operators. We denote spaces associated with each patch m as follows:

$$\mathcal{X}_m := L^2(\Gamma_{m,-}) \times \mathbb{R}_+^2 = \left\{ \left(g, \theta^{(1)}, \theta^{(2)} \right) \mid g \in L^2(\Gamma_{m,-}),\ \theta^{(1)}, \theta^{(2)} \geq 0 \right\} ,$$
$$\mathcal{Y}_m := H_2^1(\mathcal{K}_m) \times H^1(\mathcal{D}_m) = \left\{ (I,T) \mid I \in H_2^1(\mathcal{K}_m),\ T \in H^1(\mathcal{D}_m) \right\} .$$

Then we have the following operator definitions for each patch m. (For simplicity of notation, we set $\sigma \equiv 1$ in the definition (4.33) of $B(T)$.)

- The solution operator $\mathcal{S}_m : \mathcal{X}_m \to \mathcal{Y}_m$ satisfies $\mathcal{S}_m \phi_m = u_m$, where $u_m = (I_m^\varepsilon, T_m^\varepsilon)$ solves the RTE on patch $\mathcal{K}_m$ with boundary condition $\phi_m = (g_m, \theta_m^{(1)}, \theta_m^{(1)})$:

$$\begin{cases} \varepsilon v \partial_x I_m^\varepsilon &= (T_m^\varepsilon)^4 - I_m^\varepsilon \\ \varepsilon^2 \partial_x^2 T_m^\varepsilon &= (T_m^\varepsilon)^4 - \langle I_m^\varepsilon \rangle \end{cases} , \ (x,v) \in \mathcal{K}_m ,$$

  with $T_m^\varepsilon(t_m) = \theta_m^{(1)}$, $T_m^\varepsilon(s_m) = \theta_m^{(2)}$, and

$$I_m^\varepsilon|_{\Gamma_{m,-}} = g_m(x,v) = (g_m^{(1)}(x,v), g_m^{(2)}(x,v)) .$$

- The restriction operator $\mathcal{I}_{m\pm1}^m$ from patch $\mathcal{K}_m$ to the boundaries of adjacent patches,

namely, $\mathcal{K}_m \cap \Gamma_{m\pm1,-}$ and $\mathcal{D}_m \cap \partial\mathcal{D}_{m\pm1,-}$, is defined as follows:

$$\mathcal{I}^m_{m+1} u_m = (I^\varepsilon_m|_{\mathcal{K}_m\cap\Gamma_{m+1,-}}, T^\varepsilon_m|_{\mathcal{D}_m\cap\partial\mathcal{D}_{m+1,-}}), \quad m = 1,\dots,M-1\,,$$
$$\mathcal{I}^m_{m-1} u_m = (I^\varepsilon_m|_{\mathcal{K}_m\cap\Gamma_{m-1,-}}, T^\varepsilon_m|_{\mathcal{D}_m\cap\partial\mathcal{D}_{m-1,-}}), \quad m = 2,\dots,M\,.$$

- The boundary update operator $\mathcal{P}_m : \mathcal{X}_{m-1} \oplus \mathcal{X}_{m+1} \to \mathcal{X}_m$ is defined for $m \neq 1$ and $m \neq M$ by

$$\mathcal{P}_m(\phi_{m-1}, \phi_{m+1}) = (\mathcal{I}^{m-1}_m \mathcal{S}_{m-1}\phi_{m-1}, \mathcal{I}^{m+1}_m \mathcal{S}_{m+1}\phi_{m+1}). \tag{4.45}$$

  For the two "end" patches $\mathcal{K}_1$ and $\mathcal{K}_M$ that intersect with physical boundary Γ_-, boundary conditions are updated only in the interior of the domain:

$$\mathcal{P}_1 : \mathcal{X}\times\mathcal{X}_2 \to \mathcal{X}_1, \qquad \mathcal{P}_1(\phi,\phi_2) = (\phi|_{\Gamma_-\cap\Gamma_{1,-}}, \mathcal{I}^2_1\mathcal{S}_2\phi_2)\,,$$
$$\mathcal{P}_M : \mathcal{X}_{M-1}\times\mathcal{X} \to \mathcal{X}_M, \quad \mathcal{P}_M(\phi_{M-1},\phi) = (\mathcal{I}^{M-1}_M\mathcal{S}_{M-1}\phi_{M-1}, \phi|_{\Gamma_-\cap\Gamma_{M,-}})\,.$$

As suggested by Algorithm 4, in the offline stage, we construct local dictionaries on interior patches from a few random samples, enlarging each interior patch slightly to eliminate the boundary layer effect. Define $\widetilde{\mathcal{K}}_m$ and $\widetilde{\mathcal{D}}_m$ such that

$$\mathcal{K}_m \subset \widetilde{\mathcal{K}}_m = \widetilde{\mathcal{D}}_m \times \mathcal{V}\,,$$

where $\mathcal{D}_m \subset \widetilde{\mathcal{D}}_m \subset \mathcal{D}$ expands the boundary of $\mathcal{D}_m$ to both sides by a margin of Δx_{b}. Denoting by $\widetilde{\Gamma}_{m,-}$ the boundary coordinates corresponding to $\widetilde{\mathcal{D}}_m$, we let $\widetilde{\mathcal{X}}_m = L^2(\widetilde{\Gamma}_{m,-})\times\mathbb{R}^2$ capture the boundary conditions on $\partial\widetilde{\mathcal{D}}_m$.

We draw N samples $\widetilde{\phi}_{m,i}$, $i = 1,2,\dots,N$, randomly from the set

$$B_+(R_m;\widetilde{\mathcal{X}}_m) := \{\widetilde{\phi} = (\widetilde{I}_B, \widetilde{T}_B) \in \widetilde{\mathcal{X}}_m : \|\widetilde{\phi}\|_{\widetilde{\mathcal{X}}_m} \le R_m,\ \widetilde{I}_B \ge 0,\ \widetilde{T}_B \ge 0\}\,.$$

(The sampling procedure is discussed in Appendix A.2.) The local solutions $\widetilde{u}_{m,i} =$

$(\widetilde{I}^{\varepsilon}_{m,i}, \widetilde{T}^{\varepsilon}_{m,i})$ solve

$$\begin{cases} \varepsilon v \partial_x \widetilde{I}^{\varepsilon}_{m,i} = (\widetilde{T}^{\varepsilon}_{m,i})^4 - \widetilde{I}^{\varepsilon}_{m,i} \\ \varepsilon^2 \partial_x^2 \widetilde{T}^{\varepsilon}_{m,i} = (\widetilde{T}^{\varepsilon}_{m,i})^4 - \langle \widetilde{I}^{\varepsilon}_{m,i} \rangle \end{cases} \quad (x,v) \in \widetilde{\mathcal{K}}_m \,, \tag{4.46}$$

$$(\widetilde{I}^{\varepsilon}_{m,i}|_{\widetilde{\Gamma}_{m,-}}, \widetilde{T}^{\varepsilon}_{m,i}|_{\partial \widetilde{\mathcal{D}}_m}) = \widetilde{\phi}_{m,i} \,, \quad i = 1, 2, \ldots, N.$$

The solutions to these equations, confined to the original patch $\mathcal{K}_m$ and its boundary Γ_m, are used to construct two dictionaries:

$$\mathscr{I}_m = \{\psi_{m,i}\}_{i=1}^N, \quad \mathscr{B}_m = \{\phi_{m,i}\}_{i=1}^N. \tag{4.47}$$

where

$$\psi_{m,i} = (\widetilde{I}^{\varepsilon}_{m,i}|_{\mathcal{K}_m}, \widetilde{T}^{\varepsilon}_{m,i}|_{\mathcal{D}_m}) \,, \quad \phi_{m,i} = (\widetilde{I}^{\varepsilon}_{m,i}|_{\Gamma_{m,-}}, \widetilde{T}^{\varepsilon}_{m,i}|_{\partial \mathcal{D}_m}) \,.$$

In the online stage, at each iteration, we seek neighbors to interpolate for local solutions. We use the L^2 norm to measure the distance between the newly generated solutions and the older solution set. Denote by $\phi_m^{(n)}$ the solution at the n-th iteration in patch $\mathcal{K}_m$, and define by

$$\{\phi_{m,i_q^{(n)}} \,, \quad q = 1, 2, \ldots, k\}$$

its k nearest neighbors in $\mathscr{B}_m$, for some chosen positive integer k, with the indices $i_q^{(n)}$ being ordered so that $\phi_{m,i_1^{(n)}}$ is the nearest neighbor. Then we define the local tangential approximation $\mathcal{S}_m \phi_m^{(n)}$ by:

$$u_m^{(n)} = \mathcal{S}_m \phi_m^{(n)} = \psi_{m,i_1^{(n)}} + \Psi_m^{(n)} c_m^{(n)} \,, \tag{4.48}$$

where $\Psi_m^{(n)}$ and $c_m^{(n)}$ are defined as in (4.12) and (4.13). The local solution is then updated as follows:

$$\phi_m^{(n+1)} = \mathcal{P}_m(\phi_{m-1}^{(n)}, \phi_{m+1}^{(n)}) = (\mathcal{I}_m^{m-1} \mathcal{S}_{m-1} \phi_{m-1}^{(n)}, \mathcal{I}_m^{m+1} \mathcal{S}_{m+1} \phi_{m+1}^{(n)}) \,. \tag{4.49}$$

For $m = 1$ and $m = M$, to avoid updating the physical boundary, we set

$$\mathcal{P}_1(\phi, \phi_2^{(n)}) = (\phi|_{\Gamma_- \cap \Gamma_{1,-}}, \mathcal{I}_1^2 \mathcal{S}_2 \phi_2^{(n)}) \,,$$
$$\mathcal{P}_M(\phi_{M-1}^{(n)}, \phi) = (\mathcal{I}_M^{M-1} \mathcal{S}_{M-1} \phi_{M-1}^{(n)}, \phi|_{\Gamma_- \cap \Gamma_{M,-}}) \,.$$

Once the convergence is achieved (at iteration n, say), we assemble the final solution as

$$u_{\text{final}} = u^{(n)} = \sum_{m=1}^{M} \chi_m u_m^{(n)} \,, \tag{4.50}$$

with $\chi_m : \Omega \to \mathbb{R}$ being the smooth partition of unity associated with the partition of $\mathcal{K}$.

4.4.4 Numerical Tests

In the numerical tests, we take the domain to be

$$\mathcal{K} = \mathcal{D} \times \mathcal{V} = [0, L] \times [-1, 1] = [0, 3] \times [-1, 1].$$

To form the patch $\mathcal{K}_m = \mathcal{D}_m \times \mathcal{V}$, the domain $\mathcal{D}$ is divided into $M = 7$ non-overlapping patches whose widths are $d_1 = d_7 = \frac{L}{2(M-1)} = 0.25$ and $d_i = \frac{L}{M-1} = 0.5$, $i = 2, \ldots, M-1$. Each patch is then enlarged by $\Delta x_{\text{o}} = .125$ to both sides (except the ones adjacent to the physical boundary, which are enlarged only on the "internal" sides), so we have

$$\mathcal{D}_m = \left(\tfrac{L(2m-1)}{2(M-1)} - \Delta x_{\text{o}}, \tfrac{L(2m-1)}{M} + \Delta x_{\text{o}} \right) , \quad m = 2, \ldots, M-1 \,,$$
$$\mathcal{D}_1 = \left(0, \tfrac{L}{2(M-1)} + \Delta x_{\text{o}} \right) , \quad \mathcal{D}_M = \left(L - \tfrac{3}{2(M-1)} - \Delta x_{\text{o}}, 3 \right) .$$

The region of overlap between adjacent patches $\mathcal{K}_m$ has size $2\Delta x_{\text{o}} \times [-1, 1]$. The partition of unity functions over each patch $\mathcal{K}_m$ are obtained using the method of Section 4.3.4

Denote the spatial grid points by $0 = x_0 < x_1 < \cdots < x_{N_x - 1} < x_{N_x} = L$, which is a uniform grid with step size $\Delta x = \frac{L}{N_x}$. The velocity grid points are denoted by $-1 < v_1 < v_2 < \cdots < v_{N_v - 1} < v_{N_v} < 1$ for some even value of N_v. We use the Gauss-Legendre quadrature points for the v_i. The numerical solutions are denoted by

$I^{ij} \approx I(x_i, v_j)$ and $T^i \approx T(x_i)$. To quantify the numerical error, we denote the discrete L^2 norm of $u = ([I^{ij}], [T^i])$ by

$$\|u\|_2^2 = \sum_{j=1}^{N_v} w_j \frac{\Delta x}{2} |I^{0j}|^2 + \sum_{j=1}^{N_v} w_j \frac{\Delta x}{2} |I^{iN_x}|^2 + \sum_{i=1}^{N_x-1} \sum_{j=1}^{N_v} w_j \Delta x |I^{ij}|^2 + \frac{\Delta x}{2} |T^0|^2 + \frac{\Delta x}{2} |T^{N_x}|^2 + \sum_{i=1}^{N_x-1} \Delta x |T^i|^2 ,$$

where w_j is the Gauss-Legendre weight, and the relative error u_{ref} between a reference solution and an approximate solution u_{approx} is defined by

$$\text{relative } L^2 \text{ error} = \frac{\|u_{\text{ref}} - u_{\text{approx}}\|_2}{\|u_{\text{ref}}\|_2} .$$

We solve the PDE using finite differences. The intensity equation is discretized in space by a classical second-order exponential finite difference scheme [133, 191], and the temperature equation is approximated by the standard three-point scheme. The resulting nonlinear system is then solved by fixed point iteration [150, 156], where in each evaluation of the fixed point map, the monotone iterative method is exploited to solve the semilinear elliptic equation. For computations with $\varepsilon = 2^{-4}$ and $\varepsilon = 2^{-6}$, we further use Anderson acceleration to boost the convergence of fixed point iteration [11, 104, 208].

We use extremely fine discretization with $\Delta x = 2^{-14} = \frac{1}{16384}$ and $N_v = 2^{10} = 1024$. The discretization is fine enough for us to view it as the reference solution. All the other computations are done with coarser mesh $\Delta x = 2^{-11} = \frac{1}{2048}$ and $N_v = 2^7 = 128$.

The boundary condition $\phi = (g^{(1)}, g^{(2)}, \theta^{(1)}, \theta^{(2)})$ is defined as follows:

$$g^{(1)}(0, v > 0) = 3 + \sin(2\pi v) , \quad g^{(2)}(L = 3, v < 0) = 2 + \sin(2\pi v) ,$$
$$\theta(0) = \theta^{(1)} = 2 , \quad \theta(L) = \theta^{(2)} = 3 .$$

The enlarged patches needed in the offline stage, denoted by $\widetilde{\mathcal{K}}_m$, are obtained by

enlarging each respective $\mathcal{K}_m$ by the quantity Δx_{b}. The configuration of the domain and the partition are seen in Figure 4.7, where $\Delta x_{\mathrm{b}} = .125$.

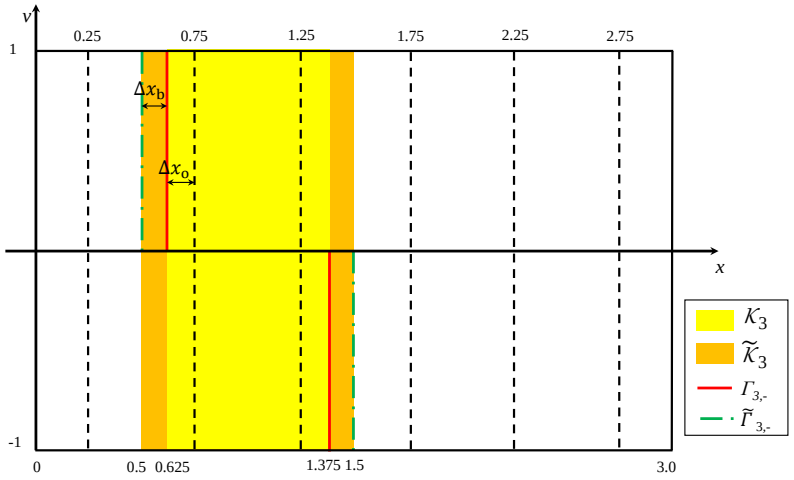


Figure 4.7: Configuration of patches (including enlarged patches) in the decomposed domain

On the buffered interior patch $\widetilde{\mathcal{K}}_m$, we sample $N = 64$ configuration of boundary conditions in $B_+(R_m; \widetilde{\mathcal{X}}_m)$. On the discrete level, this process finds 64 boundary conditions $\widetilde{\phi}$ so that

$$\|\widetilde{\phi}\|^2 = \sum_{j=1}^{\frac{N_v}{2}} w_j |\widetilde{g}^{(2)}(s, v_j)|^2 + \sum_{j=\frac{N_v}{2}+1}^{N_v} w_j |\widetilde{g}^{(1)}(t, v_j)|^2 + |\widetilde{\theta}^{(1)}|^2 + |\widetilde{\theta}^{(2)}|^2 < R_m \,.$$

We set $R_m = 25$ in our experiments.

To demonstrate the linearity of the updating map $\mathcal{P}_m$, we choose the patch $\mathcal{K}_3 = [0.625, 1.375] \times [-1, 1]$, which overlaps $\mathcal{K}_2$ at $[0.625, 0.875] \times [-1, 1]$. For $\Delta x_{\mathrm{b}} = 2^{-3}$ and $\varepsilon = 2^{-6}$, we compute local solutions on the buffered domain $\widetilde{\mathcal{K}}_3$ with 64 different configurations, and evaluate T at 0.625 and 1.375 (the two ending points of $\mathcal{K}_3$) and at 0.875 (the point that intersects with $\partial\mathcal{K}_2$). In Figure 4.8, we plot $T(0.875)$ as a function of $T(0.625)$ and $T(1.375)$. We observe that it is a slowly varying two-dimensional manifold and is locally almost linear. Thus, $T(0.875)$ can be determined uniquely by the pair of values $(T(0.625), T(1.375))$. Further, we plot $\frac{\langle |I(x,\cdot) - \langle I \rangle(x)|^2 \rangle}{\langle I \rangle(x)^2}$ and $\frac{\langle I \rangle(x) - T^4(x)}{T^4(x)}$ at $x = 0.875$, showing that the relative variation is nearly zero. This means that I is essentially constant

at $x = 0.875$, with $I = T^4$. These calculations suggest that the entire solution on this patch is uniquely determined by $T(0.625)$ and $T(1.375)$, implying that the local degrees of freedom for the solution in the entire patch is only two, so that the local solution manifold is approximately two-dimensional.

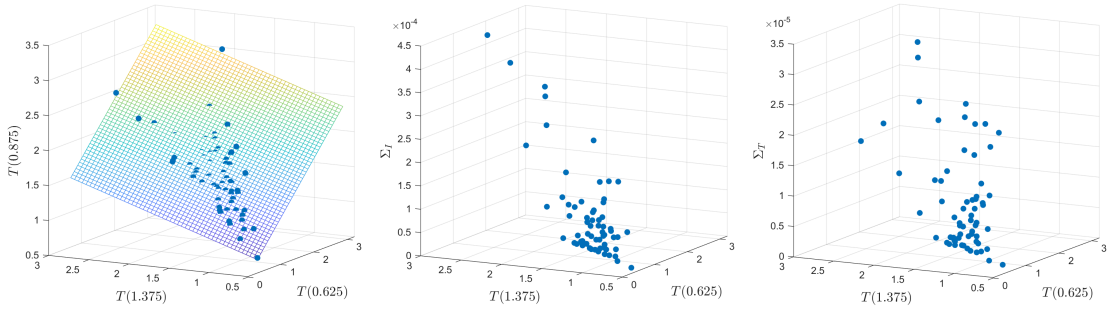
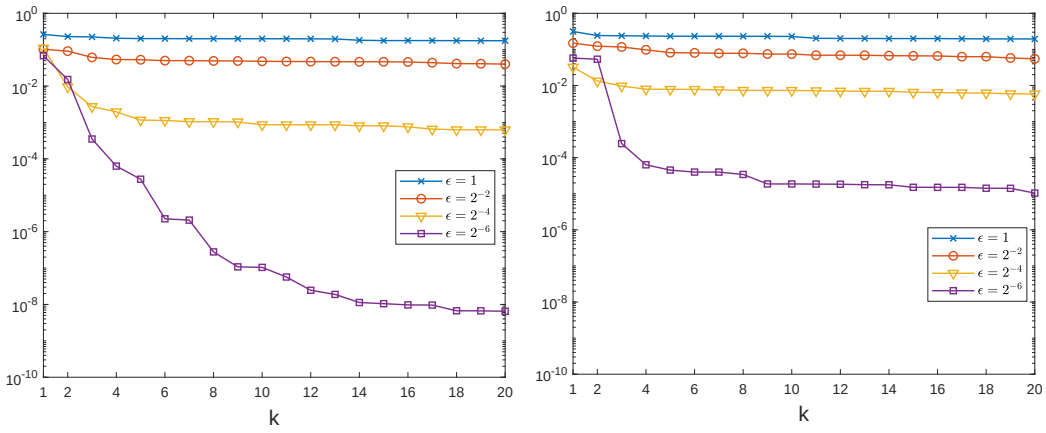


Figure 4.8: The plot on the left shows the point cloud $(T(0.625), T(1.375), T(0.875))$ and its fitting plane. We observe that the manifold is approximately two-dimensional, so that $T(0.875)$ can be uniquely determined by $(T(0.625), T(1.375))$. The middle and right panels show the quantities $\Sigma_I = \frac{\langle |I(x,\cdot) - \langle I \rangle(x)|^2 \rangle}{\langle I \rangle(x)^2}$ and $\Sigma_T = \frac{\langle I \rangle(x) - T(x)^4}{T(x)^4}$ at $x = 0.875$, respectively, showing that the solution is nearly constant, with $I = T^4$.

To verify that the local dictionary represents the solution manifold adequately, we confine the reference solution in patch $\mathcal{K}_2$ and project it onto the space spanned by its nearest k modes in the local dictionary. We evaluate the resulting relative error as a function of k, plotting the result in Figure 4.9. For $\varepsilon = 2^{-6}$ and $\Delta x_{\mathrm{b}} = .125$, we observe a sharp decay of error when $k \geq 3$, meaning that the local reference solution can be represented to acceptable accuracy by two local dictionary modes, and suggesting once again that the local solution manifold is two-dimensional.

The sample number N and the radius R_m are two crucial parameters that affect the effectiveness of the method. We check how the approximation capability of the local dictionary depends on the two parameters over the local patch $\mathcal{K}_2$. In Figure 4.10a, we show the projection error as N increases for different R_m. The error of the dictionary saturates as N increases, and it can be used as a criterion to decide the size of the local dictionary. In Figure 4.10b, we show the relative projection error of the reference solution onto the local tangent space using dictionaries with different R_m. It can be seen that the



(a) $\Delta x_{\mathrm{b}} = 2^{-2}$ (b) $\Delta x_{\mathrm{b}} = 2^{-3}$

Figure 4.9: The relative error of the L^2 projection of the reference solution onto the space spanned by the nearest k modes on the patch $\mathcal{K}_2$.

radius R_m must be large enough to obtain a good local basis.

In the online computation, we set the stopping criterion to be

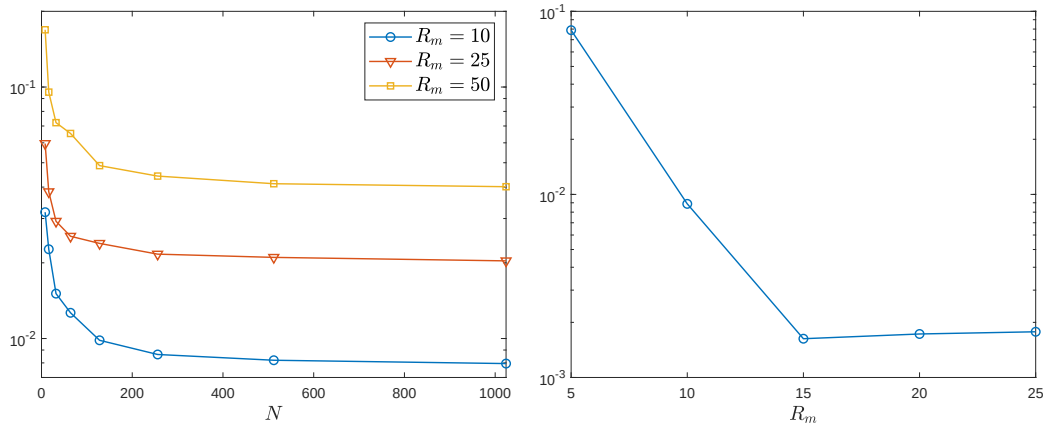
$$\sum_m \|\phi_m^{(n)} - \phi_m^{(n-1)}\| < 10^{-3}\,,$$

where $\phi_m^{(n)}$ is the boundary condition on the patch $\mathcal{K}_m$ at the n-th iteration. We take the initial boundary condition on each patch to be trivial, setting $\phi_m^{(0)}|_{\Gamma_{m,-}\setminus\Gamma_-} = 0$, except on the real physical boundary condition, where it is set to the prescribed Dirichlet conditions.

In Figure 4.11, we compare the reference solution with our numerical solution computed using $k = 5$ and buffer zone $\Delta x_{\mathrm{b}} = 2^{-3}$. When $\varepsilon = 1$, the equation is far away from its homogenization limit, and the numerical solution is far from the reference, but for $\varepsilon = 2^{-6}$ the numerical solution is captured rather well using just $k = 5$ neighbors.

In Figure 4.12 we document the relative error for various values of k and Δx_{b}. When ε is small, and for buffer width Δx_{b} sufficiently large, we need only $k = 2$ neighbors to produce a solution of acceptable accuracy. Without the buffer zone to damp the boundary layer effect, however, the low dimensionality of the solution manifold cannot be captured, even for small ε.

We also compare the cost of our reduced method with the classical Schwarz iteration.



(a) (b)

Figure 4.10: The plot on the left shows the average error of the L^2 projection of 100 test samples onto the space spanned by the nearest 5 modes. The test samples are generated from the same distribution as the dictionary. The plot on the right shows the relative error of the L^2 projection of the reference solution onto the space spanned by the nearest 5 modes on patch $\mathcal{K}_2$. The number of samples is $N = 64$ for all R_m.

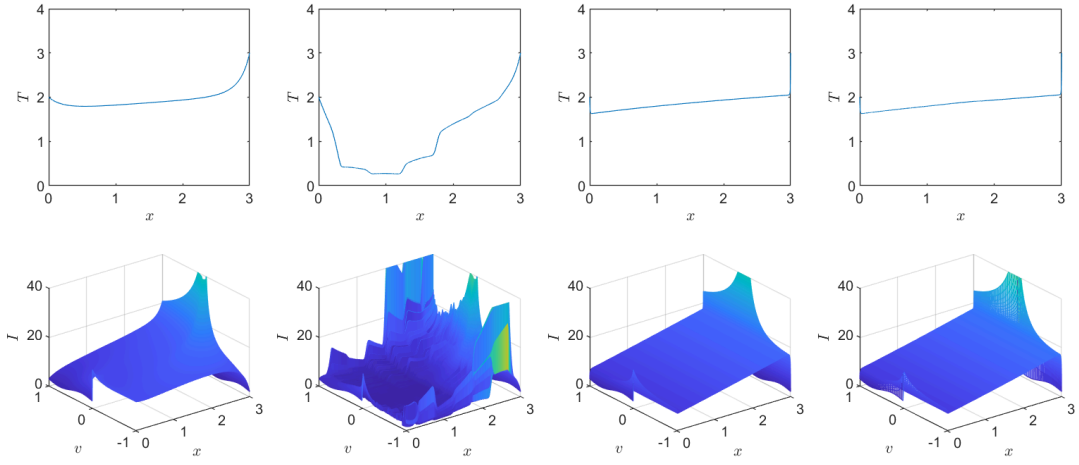
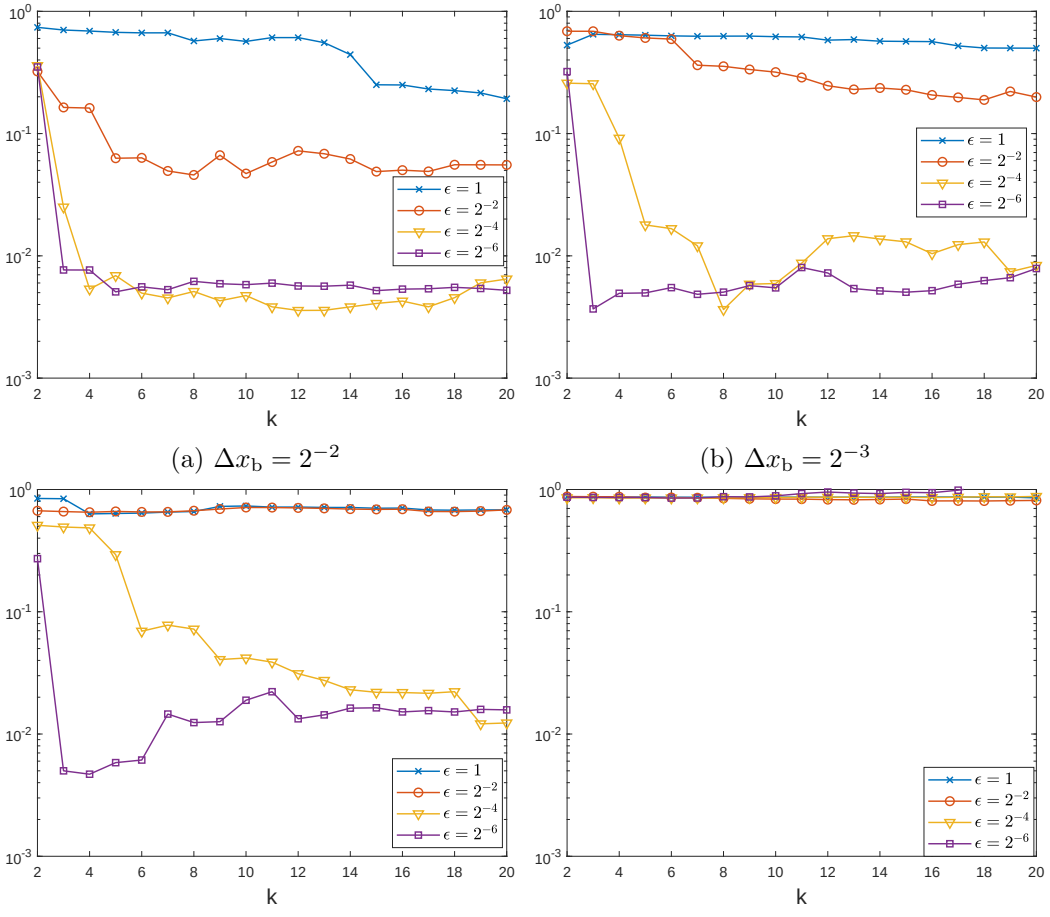


Figure 4.11: The first two columns of plots show the reference solution and numerical solution for $\varepsilon = 1$, and the last two columns compare the solutions for $\varepsilon = 2^{-6}$.



(a) $\Delta x_{\mathrm{b}} = 2^{-2}$

(b) $\Delta x_{\mathrm{b}} = 2^{-3}$

(c) $\Delta x_{\mathrm{b}} = 2^{-4}$

(d) $\Delta x_{\mathrm{b}} = 0$

Figure 4.12: The relative L^2 error in one trial as a function of k, for various values of Δx_{b} and ε.

	CPU Time (s)			
	$\varepsilon = 2^{-4}$		$\varepsilon = 2^{-6}$	
	offline	online	offline	online
Reduced model $k = 3$	394.3911	0.181324	904.7498	0.215390
Reduced model $k = 5$		0.301761		0.222538
Reduced model $k = 10$		0.379348		0.282070
Reduced model $k = 15$		0.548689		0.346633
Reduced model $k = 20$		0.586276		0.532603
Classical Schwarz	—	458.0987	—	2183.7079

Table 4.2: CPU time comparison between reduced model method with $k = 3, 5, 10, 15, 20$ (size of each local dictionary $N = 64$).

CPU times for both methods are summarized in Table 4.2 for $\varepsilon = 2^{-4}$ and $\varepsilon = 2^{-6}$, with buffer size $\Delta x_{\mathrm{b}} = .125$. The online cost of the reduced method is about 1000 times cheaper than the classical Schwarz iteration when $\varepsilon = 2^{-4}$ and 4000 times cheaper when $\varepsilon = 2^{-6}$. Even considering the large overhead cost in the offline stage, the reduced order method is still cheaper than Schwarz iteration.

Finally, we reiterate that due to the nonlinear nature of the equations, the concept of "basis function" is not well-defined. The reduced model method for linear equations was proposed in [62, 63], where random sampling is used to construct the boundary-to-boundary map $\mathcal{P}$, by following the idea of randomized SVD [121]. If we translate this approach to nonlinear homogenization, using Green's functions in a brute-force manner, the numerical results are poor. By the "Green's functions," we mean the solution to the equation with delta boundary conditions (counterparts of Green's functions in the linear setting). The numerical results are presented in Figure 4.13, that compares the ground-truth solution with the Green's function interpolation.

4.5 Conclusion

Multiscale physical phenomena are often described by PDEs that contain small parameters. It is generally expensive to capture small-scale effects using numerical solvers. There is a vast literature on improving numerical performance of PDE solvers in this context,

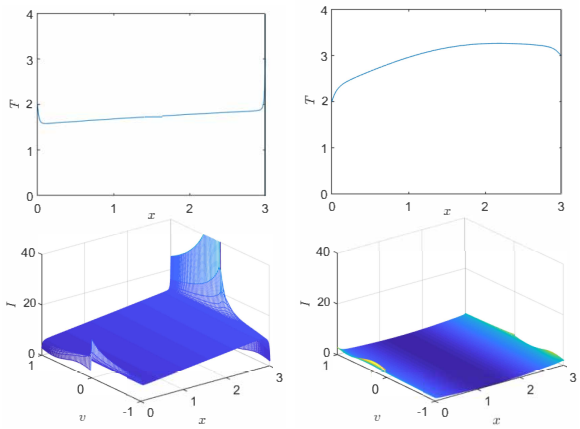


Figure 4.13: The left column of plots is the solution with $\varepsilon = 2^{-4}$, and the right column is the solution from a linear combination of the full set of "Green's functions". The top row shows the solution T and the bottom row shows the solution I.

but most algorithms are equation-specific, requiring analytical understanding to be built into algorithm design.

We have described numerical methods that can capture the homogenization limit of nonlinear PDEs with small scales automatically, without analytical prior knowledge. This work can be seen as a nonlinear extension of the earlier work [65] for linear PDEs. Elements of our algorithm include domain decomposition framework and Schwarz iteration. The method is decomposed into offline and online stages, where in the offline stage, random sampling is employed to learn the low-rank structure of the solution manifold, while in the online stage, the reduced manifolds serve as surrogates of local solvers in the Schwarz iteration. Since the manifolds are prepared offline and are of low dimension, the method exhibits significant speedup over naive approaches, as we demonstrate using computational results on the semilinear elliptic equation.

Chapter 5

Semiclassical limit of a time-dependent inverse problem for the Schrödinger equation

It is a classical derivation that the Wigner equation, derived from the Schrödinger equation that contains the quantum information, converges to the Liouville equation when the rescaled Planck constant $\varepsilon \to 0$. Since the latter presents the Newton's second law, the process is typically termed the (semi-)classical limit. In this chapter, we study the classical limit of an inverse problem for the Schrödinger equation. More specifically, we show that using the initial condition and final state of the Schrödinger equation to reconstruct the potential term, in the classical regime with $\varepsilon \to 0$, becomes using the initial and final state to reconstruct the potential term in the Liouville equation. This formally bridges an inverse problem in quantum mechanics with an inverse problem in classical mechanics.

5.1 Introduction

The classical limit, or the semi-classical limit of quantum mechanics is the ability of quantum theory to recover, or partially recover classical mechanics when the rescaled

Planck constant ε is considered negligible. More specifically, by setting $\varepsilon \approx 0$ in the Schrödinger equation, one is expected to recover the Newtonian's law of motion (Newton's second law) in the asymptotic limit.

The concept of linking quantum mechanics and classical mechanics was already in formulation in 1920s, and was presented by N. Bohr in his Nobel lecture under the name of "correspondence principle". Since then, there have been abundant studies on deriving and proving the classical limits. While the formal derivation using WKB expansion is relatively easy to show, the discontinuity in the limiting equation (Hamiltonian-Jacobi equation) makes the rigorous mathematics analysis hard to obtain. In [109, 193, 22], the authors, by introducing Wigner measures, flipped the studies to the phase space and expanded out the singularity, upon which, the derivation of classical limit was made rigorous.

We investigate the problem in an inverse setup. Suppose a quantum system is modeled by the Schrödinger equation, and one can measure the initial and final state, can one reconstruct the potential term (the field) in the equation? Moreover, if the quantum system is in the classical regime, with $\varepsilon \approx 0$, can we view this inverse problem as the inverse problem for the Newtonian motion? What is the connection between the inverse Schrödinger and the inverse Newton's law? These questions essentially come down to deriving the classical limit of the inverse problem for the Schrödinger equation.

It is a relatively big topic, and in this chapter in particular, we confine ourselves to the linearized setting. Namely, we assume the potential term is close to a preset background potential, and we are interested only in reconstructing the perturbation term. Under this setting, both the inverse Schrödinger problem and the inverse Newtonian motion problem can be formulated as Fredholm integrals, and it is the representatives (or the kernels) of the integrals that reveal the perturbed potential information. The question of deriving the classical limit, when confined in linearized setting, becomes: are the two representatives asymptotically equivalent when $\varepsilon \to 0$ in some sense?

The problem is of great interest, not only for our mathematical curiosity, but also for its practical use.

Since the fundamental question of bridging quantum mechanics and classical mechanics is mathematically clear, it is very natural to seek for its correspondence in the inverse setting. Indeed, in what sense can one view the inverse Schrödinger problem and the inverse Newtonian motion problem equivalently? Or, is it possible for one problem to be more stable than the other? This type of stability increasing/decreasing problem recently attracts a large amount of attention for various sets of problems [61, 153, 209, 176].

Practically, the Schrödinger equation is not only regarded as the fundamental model for quantum mechanics, but also emerges as the limit of the Helmholtz equation when dynamics in different dimensions is described at separate scales [113], and thus serves as a fundamental model for the wave propagation (for a fixed high frequency) as well. There are abundant applications, in which high-frequency waves are sent to detect the media [21, 26, 29, 39, 200]. Mathematically, this is to seek for reconstructing the speed of sound in the Helmholtz equation, which is to reconstruct the potential term in the Schrödinger equation. Moreover, the inverse Schrödinger problem is also a transformed version of the celebrated Calderón problem, arises from Electrical Impedance Tomography (EIT) [55]. For these reasons, inverse Schrödinger problem has long been regarded as one of the most important inverse problems. Most of the studies, however, set the Planck constant in the Schrödinger equation to be an $O(1)$ value. This is not practical in many applications mentioned above. In the high frequency regime for the Helmholtz equation, or in the classical regime with the rescaled $\varepsilon \to 0$, the stability of the inverse problem may change, and it would be of great practical interests to predict the stability in these regimes, and to quantify the reconstruction error in terms of the rescaled ε. Linking it to the inverse Newtonian motion is a natural strategy.

Despite the great importance of the problem, the theoretical study has been thin, even though it is mentioned a couple of times in the literature [143, 140, 141, 178]. Most of the studies formulate the problem as the (quantum) scattering problem. See also the geometric version for reconstructing the refraction index [175, 174, 123, 173]. The obstacles come from (a) the disparity of the technicalities used in deriving the classical limit,

and in analyzing inverse problems, and (b) the disparity in analyzing the two different inverse problems (inverse Schrödinger and inverse Newtonian motion). In this chapter, we take an initial attempt to bridge the two under the linearized setting, hoping to unveil some connections that could potentially serve as stepping stones for further investigation. We should mention, that when the media encodes randomness, the classical limit of the Schrödinger equation (or similarly the wave equation) is the linear Boltzmann equation (or the radiative transfer equation) that characterizes the dynamics of photons on the mesoscopic level. The associated inverse problem is highly related to imaging, and has been studied in different contexts [24, 27, 25, 58, 126].

This chapter is organized as follows. In Section 5.2, we utilize the linearization approach to set up the frameworks for Schrödinger, Wigner and Liouville inverse problems. The relations between and the three inverse problems are considered in Section 5.3, including the equivalence of the Schrödinger and the Wigner inverse problem, and the convergence from the Wigner to the Liouville inverse problem as $\varepsilon \to 0$. Numerical tests are exploited in Section 5.4 to demonstrate the convergence from the Wigner to the Liouville inverse problem.

5.2 Three inverse problems for the Schrödinger equation in the classical limit

As We have seen from Section 2.3, for the Schrödinger equation in the classical limit, we are now facing three equations: the original Schrödinger equation, the Wigner equation, and the Liouville equation as the classical limit of the Wigner equation. With respect to these three equations, we can formulate three inverse problems, all of which will be derived in this section.

We employ the same setup for the three inverse problems: we assume the equations are Cauchy problems without boundary constraints, and we confine ourselves to the linearized setting. This is to assume the potential term V is close enough to a background V_{b}. The given input is the initial data and one can measure the final state at a given time T. The

to-be-reconstructed parameter is the potential term V (or equivalently $\widetilde{V} = V - V_{\mathrm{b}}$).

We present the three inverse problems in the following three subsection respectively.

5.2.1 A linearized inverse problem for the Schrödinger equation

Recall the Schrödinger equation in $\mathbb{R}^d$ is

$$\mathrm{i}\varepsilon\partial_t\phi^\varepsilon = -\frac{1}{2}\varepsilon^2\Delta_x\phi^\varepsilon + V(x)\phi^\varepsilon\,. \tag{5.1}$$

Let the initial data be $\phi^\varepsilon(0,x) = \phi^\varepsilon_{\mathrm{I}}(x)$, and final data at $t = T$ be $\phi^\varepsilon(T,x) = \phi^\varepsilon_T(x)$. While the forward problem is to compute ϕ^ε_T for every given $\phi^\varepsilon_{\mathrm{I}}$, the inverse problem is to use $(\phi^\varepsilon_{\mathrm{I}}, \phi^\varepsilon_T)$ data pairs to reconstruct V. In other words, denoting

$$\mathcal{M}^\varepsilon_{\mathrm{S}}[V]\,:\,\phi^\varepsilon_{\mathrm{I}} \to \phi^\varepsilon_T\,,$$

the inverse problem is to use the map $\mathcal{M}^\varepsilon_{\mathrm{S}}[V]$ to reconstruct V.

Remark 5.1. *The reconstruction is at most unique up to a gauge transform. Indeed, let $\hat{H}^\varepsilon$ be the Hamiltonian operator* (2.13)*, and define $\hat{H}^\varepsilon_n$ to be a new Hamiltonian operator*

$$\hat{H}^\varepsilon_n = \hat{H}^\varepsilon + \frac{2\pi\varepsilon}{T}n\,,\quad n\in\mathbb{Z}\,.$$

We further define the unitary semi-group generated by $\hat{H}^\varepsilon$ and $\hat{H}^\varepsilon_n$:

$$U^\varepsilon(t) = e^{-\mathrm{i}t\hat{H}^\varepsilon/\varepsilon}\,,\quad U^\varepsilon_n(t) = e^{-\mathrm{i}t\hat{H}^\varepsilon_n/\varepsilon}\,,\quad t>0\,.$$

Clearly the two Hamiltonian operators are different, but $U^\varepsilon(T) = U^\varepsilon_n(T)$, for all $n\in\mathbb{Z}$. This suggests that the initial-to-final map

$$\mathcal{M}^\varepsilon_{\mathrm{S}}[V] = \mathcal{M}^\varepsilon_{\mathrm{S}}\left[V + \frac{2\pi\varepsilon}{T}n\right]\,,$$

and thus the reconstruction cannot be unique.

To derive the linearized version of the inverse problem, we assume there is a known background potential term $V_{\mathrm{b}}(x)$ such that

$$\widetilde{V}(x) = V(x) - V_{\mathrm{b}}(x)$$

is much smaller than $V_b(x)$ in amplitude. We further write the background problem with the same initial condition:

$$\begin{aligned} \mathrm{i}\varepsilon\partial_t \phi_{\mathrm{b}}^{\varepsilon} &= -\frac{1}{2}\varepsilon^2 \Delta_x \phi_{\mathrm{b}}^{\varepsilon} + V_{\mathrm{b}}(x)\phi_{\mathrm{b}}^{\varepsilon}\,, \\ \phi_{\mathrm{b}}^{\varepsilon}(0,x) &= \phi_{\mathrm{I}}^{\varepsilon}(x)\,. \end{aligned} \tag{5.2}$$

For a preset V_{b} and $\phi_{\mathrm{I}}^{\varepsilon}(x)$, one can compute the equation for $\phi_{\mathrm{b}}^{\varepsilon}(T,x) = \phi_{\mathrm{b},T}^{\varepsilon}(x)$.

Let $\widetilde{\phi}^{\varepsilon} = \phi^{\varepsilon} - \phi_b^{\varepsilon}$ be the perturbation of wave ϕ^{ε}, then by subtracting the equation (5.1) from (5.2) and omitting the higher order term $\widetilde{V}\widetilde{\phi}^{\varepsilon}$, we get the equation for the perturbation $\widetilde{\phi}^{\varepsilon}$

$$\begin{aligned} \mathrm{i}\varepsilon\partial_t \widetilde{\phi}^{\varepsilon} &= -\frac{1}{2}\varepsilon^2 \Delta_x \widetilde{\phi}^{\varepsilon} + V_{\mathrm{b}}(x)\widetilde{\phi}^{\varepsilon} + \widetilde{V}(x)\phi_{\mathrm{b}}^{\varepsilon}\,, \\ \widetilde{\phi}^{\varepsilon}(0,x) &= 0\,. \end{aligned} \tag{5.3}$$

Note that $\widetilde{\phi}^{\varepsilon}$ has trivial initial data and implicitly depends on the initial condition $\phi_{\mathrm{I}}^{\varepsilon}$ through the background wave $\phi_{\mathrm{b}}^{\varepsilon}$. Knowing the measured data $\phi_T^{\varepsilon}(x)$, and the computed data $\phi_{\mathrm{b},T}^{\varepsilon}(x)$, we merely take the difference and define

$$\widetilde{\phi}_T^{\varepsilon} = \widetilde{\phi}^{\varepsilon}(T,x) = \phi_T^{\varepsilon}(x) - \phi_{\mathrm{b},T}^{\varepsilon}(x)\,. \tag{5.4}$$

The inverse problem now translates to reconstructing $\widetilde{V}$ using $(\phi_{\mathrm{I}}^{\varepsilon}\,, \widetilde{\phi}_T^{\varepsilon})$ data pairs. To do so, we formulate the adjoint equation ψ^{ε} that solves:

$$\begin{aligned} \mathrm{i}\varepsilon\partial_t \psi^{\varepsilon} &= -\frac{1}{2}\varepsilon^2 \Delta_x \psi^{\varepsilon} + V_{\mathrm{b}}(x)\psi^{\varepsilon}\,, \\ \psi^{\varepsilon}(T,x) &= \psi_T^{\varepsilon}(x)\,, \end{aligned} \tag{5.5}$$

where the data is given at the final time T.

Taking (5.3) $\times\ \overline{\psi^\varepsilon} - \overline{(5.5)} \times \widetilde{\phi}^\varepsilon$, we arrive at

$$\mathrm{i}\varepsilon\partial_t(\widetilde{\phi}^\varepsilon\overline{\psi^\varepsilon}) = -\frac{1}{2}\varepsilon^2(\overline{\psi^\varepsilon}\Delta_x\widetilde{\phi}^\varepsilon - \widetilde{\phi}^\varepsilon\Delta_x\overline{\psi^\varepsilon}) + \widetilde{V}(x)\phi_{\mathrm{b}}^\varepsilon\overline{\psi^\varepsilon}\,.$$

We integrate the equation over $\mathbb{R}^d \times [0,T]$, and then apply the Green's identity and make use of the trivial initial data for $\widetilde{\phi}^\varepsilon$. This finally yields our problem formulation

$$\int_{\mathbb{R}^d} \widetilde{\phi}_T^\varepsilon\overline{\psi_T^\varepsilon}\mathrm{d}x = \frac{1}{\mathrm{i}\varepsilon}\int_{\mathbb{R}^d}\widetilde{V}(x)\int_0^T \phi_{\mathrm{b}}^\varepsilon\overline{\psi^\varepsilon}\mathrm{d}t\mathrm{d}x = \int_{\mathbb{R}^d}\widetilde{V}(x)R_{\mathrm{S}}^\varepsilon[\phi_{\mathrm{I}}^\varepsilon,\psi_T^\varepsilon](x)\mathrm{d}x\,, \tag{5.6}$$

where we call the representative:

$$R_{\mathrm{S}}^\varepsilon[\phi_{\mathrm{I}}^\varepsilon,\psi_T^\varepsilon] = \frac{1}{\mathrm{i}\varepsilon}\int_0^T \phi_{\mathrm{b}}^\varepsilon\overline{\psi^\varepsilon}\mathrm{d}t\,. \tag{5.7}$$

Note that the left hand side of (5.6) is known, with ψ_T^ε given in (5.5) and $\widetilde{\phi}_T^\varepsilon$ calculated from the measured data (5.4). The right hand side formulates a Fredholm integral on the unknown $\widetilde{V}$ and the kernel $R_{\mathrm{S}}^\varepsilon$. Reconstruction of $\widetilde{V}$ amounts to inverting this Fredholm integral using different configurations of $R_{\mathrm{S}}^\varepsilon$, which, in turn, are tuned by $(\phi_{\mathrm{I}}^\varepsilon\,,\psi_T^\varepsilon)$ data pairs.

5.2.2 A linearized inverse problem for the Wigner equation

The counterpart of the Schrödinger equation on the phase space is the Wigner equation. We derive the linearized inverse problem for this equation assuming initial and final states are given. Recall the Wigner equation in $\mathbb{R}^{2d}$:

$$\partial_t f^\varepsilon + k\cdot\nabla_x f^\varepsilon = \mathcal{L}_V^\varepsilon[f^\varepsilon]\,, \tag{5.8}$$

with

$$\mathcal{L}_V^\varepsilon[f^\varepsilon] = \mathrm{i}\int_{\mathbb{R}^{2d}} e^{\mathrm{i}p(x-y)}V(y)\frac{1}{\varepsilon}\left[f^\varepsilon\left(x,k+\frac{1}{2}\varepsilon p\right) - f^\varepsilon\left(x,k-\frac{1}{2}\varepsilon p\right)\right]\mathrm{d}p\mathrm{d}y\,.$$

Let the initial data be $f^\varepsilon(0,x,k) = f^\varepsilon_{\mathrm{I}}(x,k)$, and we call the final time data $f^\varepsilon(T,x,k) = f^\varepsilon_T(x,k)$. The goal is to use initial-final data pairs $(f^\varepsilon_{\mathrm{I}}\,, f^\varepsilon_T)$ to reconstruct the potential term V. This amounts to using the following operator to reconstruct V:

$$\mathcal{M}^\varepsilon_{\mathrm{W}}[V] \,:\, f^\varepsilon_{\mathrm{I}} \to f^\varepsilon_T\,.$$

Remark 5.2. *According to the definition of $\mathcal{L}_V$, it is immediate that $\mathcal{L}^\varepsilon_V = \mathcal{L}^\varepsilon_{V+C}$ where C can be any constant. This makes $\mathcal{M}^\varepsilon_{\mathrm{W}}[V] = \mathcal{M}^\varepsilon_{\mathrm{W}}[V+C]$. Therefore the reconstruction can be at most unique up to an unknown constant.*

To derive the linear inverse problem, we assume that there is a background potential $V_{\mathrm{b}}(x)$ so that $\widetilde{V}(x) = V(x) - V_{\mathrm{b}}(x)$ is much smaller than $V_{\mathrm{b}}(x)$ in amplitude. Call the background problem with the same initial condition:

$$\begin{aligned} \partial_t f^\varepsilon_{\mathrm{b}} + k\cdot\nabla_x f^\varepsilon_{\mathrm{b}} &= \mathcal{L}^\varepsilon_{V_{\mathrm{b}}}[f^\varepsilon_{\mathrm{b}}]\,, \\ f^\varepsilon_{\mathrm{b}}(0,x,k) &= f^\varepsilon_{\mathrm{I}}(x,k)\,, \end{aligned} \tag{5.9}$$

where the operator $\mathcal{L}_{V_{\mathrm{b}}}$ is defined by the background potential. With a preset V_{b} and $f^\varepsilon_{\mathrm{I}}$, $f^\varepsilon_{\mathrm{b},T}(x,k) = f^\varepsilon_{\mathrm{b}}(T,x,k)$ can be pre-computed.

Define $\widetilde{f}^\varepsilon = f^\varepsilon - f^\varepsilon_{\mathrm{b}}$, we subtract (5.9) from (5.8), and drop the higher term $\mathcal{L}^\varepsilon_{\widetilde{V}}[\widetilde{f}^\varepsilon]$ to have the equation for $\widetilde{f}^\varepsilon$:

$$\begin{aligned} \partial_t \widetilde{f}^\varepsilon + k\cdot\nabla_x \widetilde{f}^\varepsilon &= \mathcal{L}^\varepsilon_{V_{\mathrm{b}}}[\widetilde{f}^\varepsilon] + \mathcal{L}^\varepsilon_{\widetilde{V}}[f^\varepsilon_b]\,, \\ \widetilde{f}^\varepsilon(0,x,k) &= 0\,. \end{aligned} \tag{5.10}$$

This equation describes the dynamics of the perturbed data $\widetilde{f}^\varepsilon$. It has trivial initial data, and implicitly depends on $f^\varepsilon_{\mathrm{I}}$ through the $\mathcal{L}^\varepsilon_{\widetilde{V}}[f^\varepsilon_{\mathrm{b}}]$ term. Since $f^\varepsilon_T(x,k)$ is the measured data and $f^\varepsilon_{\mathrm{b},T}(x,k)$ is precomputed, the perturbed equation's final data is also known:

$$\widetilde{f}^\varepsilon_T(x,k) = \widetilde{f}^\varepsilon(T,x,k) = f^\varepsilon(T,x,k) - f^\varepsilon_{\mathrm{b}}(T,x,k) = f^\varepsilon_T(x,k) - f^\varepsilon_{\mathrm{b},T}(x,k)\,.$$

As was done in the case of the Schrödinger equation, we also derive the adjoint equation for g^ε:

$$\begin{aligned} \partial_t g^\varepsilon + k \cdot \nabla_x g^\varepsilon &= \mathcal{L}^\varepsilon_{V_\mathrm{b}}[g^\varepsilon]\,, \\ g^\varepsilon(T,x,k) &= g^\varepsilon_T(x,k)\,, \end{aligned} \tag{5.11}$$

with the data imposed at the final time $t = T$.

Taking (5.10) $\times\, \overline{g^\varepsilon} + \overline{(5.11)} \times \widetilde{f}^\varepsilon$, we arrive at

$$\partial_t(\widetilde{f}^\varepsilon \overline{g^\varepsilon}) + \nabla_x \cdot (k \widetilde{f}^\varepsilon \overline{g^\varepsilon}) = \overline{g^\varepsilon}\mathcal{L}^\varepsilon_{V_\mathrm{b}}[\widetilde{f}^\varepsilon] + \widetilde{f}^\varepsilon \overline{\mathcal{L}^\varepsilon_{V_\mathrm{b}}[g^\varepsilon]} + \overline{g^\varepsilon}\mathcal{L}^\varepsilon_{\widetilde{V}}[f^\varepsilon_\mathrm{b}]\,. \tag{5.12}$$

We integrate the equation over $\mathbb{R}^{2d} \times [0,T]$. Making use of the anti-self-adjointness of $\mathcal{L}^\varepsilon_{V_\mathrm{b}}$, as shown in (2.21), and the trivial initial data of $\widetilde{f}^\varepsilon$, we obtain:

$$\int_{\mathbb{R}^{2d}} \widetilde{f}^\varepsilon_T \overline{g^\varepsilon_T} \mathrm{d}x\mathrm{d}k = \int_{\mathbb{R}^{2d}\times[0,T]} \overline{g^\varepsilon}\mathcal{L}^\varepsilon_{\widetilde{V}}[f^\varepsilon_\mathrm{b}] \mathrm{d}x\mathrm{d}k\mathrm{d}t\,.$$

We note that the integral term on the right hand side of the equation is a linear operator on $\widetilde{V}$. To do so, we expand $\mathcal{L}^\varepsilon_{\widetilde{V}}$, and employ (2.18):

$$\int_{\mathbb{R}^{2d}} \widetilde{f}^\varepsilon_T \overline{g^\varepsilon_T} \mathrm{d}x\mathrm{d}k = \int_{\mathbb{R}^d} \widetilde{V}(x) R^\varepsilon_\mathrm{W}[f^\varepsilon_\mathrm{I}, g^\varepsilon_T](x)\mathrm{d}x\,, \tag{5.13}$$

with the representative

$$R^\varepsilon_\mathrm{W}[f^\varepsilon_\mathrm{I}, g^\varepsilon_T] = \frac{\mathrm{i}}{(2\pi)^d} \int_{\mathbb{R}^{3d}\times[0,T]} e^{\mathrm{i}p(z-x)} \overline{g^\varepsilon}(z,k) D^\varepsilon f^\varepsilon_\mathrm{b}(z,k,p) \mathrm{d}p\mathrm{d}z\mathrm{d}k\mathrm{d}t\,, \tag{5.14}$$

where $D^\varepsilon f^\varepsilon_\mathrm{b}$ is defined in (2.19).

Once again, the left hand side of (5.13) is known, with $\widetilde{f}^\varepsilon_T$ computed and g^ε_T given, and the right hand side of (5.13) is a Fredholm integral on $\widetilde{V}$ with the kernel R^ε_W. The linear inverse problem of the Wigner equation amounts to inverting such an integral. By choosing different configurations of $(f^\varepsilon_\mathrm{I}, g^\varepsilon_T)$, we obtain different profiles of R^ε_W, using which, we try to reconstruct $\widetilde{V}$.

5.2.3 A linearized inverse problem for the Liouville equation

Finally, we derive the inverse problem for the Liouville equation. Recall the Liouville equation in $\mathbb{R}^{2d}$

$$\partial_t f + k \cdot \nabla_x f - \nabla_x V \cdot \nabla_k f = 0 \,. \tag{5.15}$$

Denote the initial data to be $f(0, x, k) = f_{\mathrm{I}}(x, k)$, and we assume one can experimentally measure the final time solution at $t = T$ for $f_T(x, k) = f(T, x, k)$. The goal is to reconstruct V in the Liouville equation using the initial-to-final data pairs $(f_{\mathrm{I}}\,, f_T)$. Namely, to use the following operator to reconstruct V:

$$\mathcal{M}^{\varepsilon}_{\mathrm{L}}[V] \,:\, f_{\mathrm{I}} \to f_T \,.$$

Remark 5.3. *Since the potential term enters the equation* (5.15) *through its gradient* $\nabla_x V$, $\mathcal{M}^{\varepsilon}_{\mathrm{L}}[V] = \mathcal{M}^{\varepsilon}_{\mathrm{L}}[V + C]$ *for any constant* C. *This means the reconstruction of* V *is at most unique up to an arbitrary constant.*

As was done in the previous sections, the problem shall be linearized around a background potential $V_{\mathrm{b}}(x)$. The background equation with the same initial data writes:

$$\begin{aligned} \partial_t f_{\mathrm{b}} + k \cdot \nabla_x f_{\mathrm{b}} - \nabla_x V_{\mathrm{b}} \cdot \nabla_k f_{\mathrm{b}} = 0 \,, \\ f_{\mathrm{b}}(0, x, k) = f_{\mathrm{I}}(x, k) \,. \end{aligned} \tag{5.16}$$

Denoting the perturbation $\widetilde{f} = f - f_{\mathrm{b}}$, we subtract (5.16) from (5.15), and drop the higher order term $\nabla_x \widetilde{V} \cdot \nabla_k \widetilde{f}$ to obtain the equation for the perturbation:

$$\begin{aligned} \partial_t \widetilde{f} + k \cdot \nabla_x \widetilde{f} - \nabla_x V_{\mathrm{b}} \cdot \nabla_k \widetilde{f} = \nabla_x \widetilde{V} \cdot \nabla_k f_{\mathrm{b}} \,, \\ \widetilde{f}(0, x, k) = 0 \,. \end{aligned} \tag{5.17}$$

The equation has trivial initial data, but it implicitly depends on f_{I} through the f_{b} term that enters as the source. Since $f_T(x, k)$ is measured, and $f_{\mathrm{b}}(T, x, k)$ can be precomputed

for any given $V_{\rm b}$, we easily obtain:

$$\widetilde{f}_T = \widetilde{f}(T,x,k) = f(T,x,k) - f_{\rm b}(T,x,k)$$

as a known quantity.

The adjoint equation for g is:

$$\begin{aligned} \partial_t g + k\cdot\nabla_x g - \nabla_x V_{\rm b}\cdot\nabla_k g = 0\,, \\ g(T,x,k) = g_T(x,k)\,. \end{aligned} \tag{5.18}$$

Taking (5.17) $\times\,\overline{g} + \overline{(5.18)}\times\widetilde{f}$, we arrive at

$$\partial_t(\widetilde{f}\overline{g}) + \nabla_x\cdot(k\widetilde{f}\overline{g}) - \nabla_k\cdot(\widetilde{f}\overline{g}\nabla_x V_{\rm b}) = \overline{g}\nabla_x\widetilde{V}\cdot\nabla_k f_{\rm b}\,.$$

We integrate the equation over $\mathbb{R}^{2d}\times[0,T]$, and make use of the trivial initial data for $\widetilde{f}$:

$$\int_{\mathbb{R}^{2d}}\widetilde{f}_T\overline{g}_T{\rm d}x{\rm d}k = \int_{\mathbb{R}^{2d}\times[0,T]}\overline{g}\nabla_x\widetilde{V}\cdot\nabla_k f_b{\rm d}x{\rm d}k{\rm d}t\,.$$

Moving the ∇_x from V to $\overline{g}\nabla_k f_{\rm b}$, this becomes

$$\int_{\mathbb{R}^{2d}}\widetilde{f}_T\overline{g}_T{\rm d}x{\rm d}k = \int_{\mathbb{R}^d}\widetilde{V}(x)R_{\rm L}[f_{\rm I},g_T](x){\rm d}x\,, \tag{5.19}$$

where the Liouville representative is defined as:

$$R_{\rm L}[f_{\rm I},g_T] = -\nabla_x\cdot\int_{\mathbb{R}^d\times[0,T]}\overline{g}\nabla_k f_{\rm b}{\rm d}k{\rm d}t\,. \tag{5.20}$$

Again, the left hand side of (5.19) is known, and the right hand side of (5.19) is a Fredholm integral of $\widetilde{V}$ with the kernel $R_{\rm L}$. The linear inverse problem of the Liouville equation is to invert such an integral.

5.3 Connecting the three inverse problems

The Schrödinger equation, the Wigner equation and the Liouville equation are connected. According to Lemma 2.1 and Theorem 2.2, $f^\varepsilon = W^\varepsilon[\phi^\varepsilon]$ necessarily satisfies the Wigner equation as long as ϕ^ε solves the Schrödinger equation, and when $\varepsilon \to 0$, $f^\varepsilon \to f$ that solves the Liouville equation.

We look for the counterparts of these relations in the inverse setting. This is to investigate the three inverse problems introduced in Section 5.2. More specifically, since the three inverse problems are uniquely represented by the three representatives $R^\varepsilon_{\rm S}$, $R^\varepsilon_{\rm W}$ and $R_{\rm L}$, as defined in (5.7), (5.14) and (5.20) respectively, we essentially need to show the connections between them.

5.3.1 From Schrödinger to Wigner in the inverse setting

This is to study the relation between the linear Schrödinger inverse problem (5.6) and the linear Wigner inverse problem (5.13). For simplicity of notations, we drop the ε superscript throughout the subsection. The theorem below demonstrates that every Wigner representative $R_{\rm W}$ can be written as a linear combination of Schrödinger representatives $R_{\rm S}$. This means the space spanned by all $R_{\rm W}$ is a subspace spanned by all $R_{\rm S}$.

Theorem 5.1. *Let $\phi_{\rm b}(t)$ and $\phi'_{\rm b}(t)$ be the solutions to the background Schrödinger equation* (5.2) *with initial data $\phi_{\rm I}$ and $\phi'_{\rm I}$ respectively. Let $\psi(t)$ and $\psi'(t)$ be the solutions to the adjoint Schödinger equation* (5.5) *with final data ψ_T and ψ'_T respectively. More over, let $f_{\rm I} = W[\phi_{\rm I}, \psi'(0)]$ and $g_T = W[\psi_T, \phi'_{\rm b}(T)]$. Then*

$$(2\pi\varepsilon)^d R_{\rm W}[f_{\rm I}, g_T] = \left\langle \phi'_{\rm I}, \psi'(0) \right\rangle R_{\rm S}[\phi_{\rm I}, \psi_T] - \left\langle \phi_{\rm I}, \psi(0) \right\rangle R_{\rm S}[\phi'_{\rm I}, \psi'_T]\,. \tag{5.21}$$

Proof. According to the definition of Wigner representative (5.14), let $f_{\rm b}$ and g solve the background and the adjoint Wigner equations, $R_{\rm W}[f_{\rm I}, g_T]$ becomes

$$R_{\rm W}[f_{\rm I}, g_T] = \frac{\mathrm{i}}{(2\pi)^d} \int_0^T I(t)\mathrm{d}t \tag{5.22}$$

with

$$I = \int_{\mathbb{R}^{3d}} e^{\mathrm{i}p(z-x)} \overline{g}(z,k) D f_{\mathrm{b}}(z,k,p) \mathrm{d}p \mathrm{d}z \mathrm{d}k \,.$$

According to (2.1), $g(t) = W[\psi(t), \phi_{\mathrm{b}}'(t)]$ and $f_{\mathrm{b}}(t) = W[\phi_{\mathrm{b}}(t), \psi'(t)]$ and thus

$$I = \int_{\mathbb{R}^{3d}} e^{\mathrm{i}p(z-x)} \overline{W[\psi(t), \phi_{\mathrm{b}}'(t)]}(z,k) DW[\phi_{\mathrm{b}}(t), \psi'(t)](z,k,p) \mathrm{d}p \mathrm{d}z \mathrm{d}k \,.$$

Here $DW[\phi_{\mathrm{b}}(t), \psi'(t)]$ is defined as in (2.19). Plugging in the Wigner transform, we get

$$\begin{aligned} I =& \frac{1}{(2\pi)^{2d}\varepsilon} \int_{\mathbb{R}^{5d}} e^{\mathrm{i}k(q-y)} [e^{\mathrm{i}p\left(z+\frac{1}{2}\varepsilon q - x\right)} - e^{\mathrm{i}p\left(z-\frac{1}{2}\varepsilon q - x\right)}] \\ & \overline{\psi}\left(z - \frac{1}{2}\varepsilon y\right) \phi_{\mathrm{b}}'\left(z + \frac{1}{2}\varepsilon y\right) \phi_{\mathrm{b}}\left(z - \frac{1}{2}\varepsilon q\right) \overline{\psi'}\left(z + \frac{1}{2}\varepsilon q\right) \mathrm{d}y \mathrm{d}q \mathrm{d}p \mathrm{d}z \mathrm{d}k \\ =& \frac{1}{(2\pi)^{d}\varepsilon} \int_{\mathbb{R}^{3d}} [e^{\mathrm{i}p\left(z+\frac{1}{2}\varepsilon q - x\right)} - e^{\mathrm{i}p\left(z-\frac{1}{2}\varepsilon q - x\right)}] \\ & \overline{\psi}\left(z - \frac{1}{2}\varepsilon q\right) \phi_{\mathrm{b}}'\left(z + \frac{1}{2}\varepsilon q\right) \phi_{\mathrm{b}}\left(z - \frac{1}{2}\varepsilon q\right) \overline{\psi'}\left(z + \frac{1}{2}\varepsilon q\right) \mathrm{d}q \mathrm{d}p \mathrm{d}z \,, \end{aligned} \tag{5.23}$$

where we used the Fourier inversion formula

$$\frac{1}{(2\pi)^d} \int_{\mathbb{R}^{2d}} e^{\mathrm{i}k(q-y)} h(y) \mathrm{d}y \mathrm{d}k = h(q) \,. \tag{5.24}$$

Let $z' = z + \frac{1}{2}\varepsilon q, z'' = z - \frac{1}{2}\varepsilon q$, we get

$$\begin{aligned} I &= \frac{1}{(2\pi)^d \varepsilon^{d+1}} \int_{\mathbb{R}^{3d}} [e^{\mathrm{i}p(z'-x)} - e^{\mathrm{i}p(z''-x)}] \overline{\psi}(z'') \phi_{\mathrm{b}}'(z') \phi_{\mathrm{b}}(z'') \overline{\psi'}(z') \mathrm{d}z' \mathrm{d}p \mathrm{d}z'' \\ &= \frac{1}{\varepsilon^{d+1}} [\langle \phi_{\mathrm{b}}(t), \psi(t) \rangle_{L^2(\mathbb{R}^d)} \, \phi_{\mathrm{b}}'(x) \overline{\psi'}(x) - \left\langle \phi_{\mathrm{b}}'(t), \psi'(t) \right\rangle_{L^2(\mathbb{R}^d)} \phi_{\mathrm{b}}(x) \overline{\psi}(x)] \end{aligned} \tag{5.25}$$

where we again use the Fourier inversion formula in the second equality.

According to (2.14), $\langle \phi_{\mathrm{b}}(t), \psi(t) \rangle$ and $\langle \phi_{\mathrm{b}}'(t), \psi'(t) \rangle$ are both constants independent of t. Using (5.22) and the definition of the Schrödinger representative (5.7), we integrate the equation over $[0, T]$ to conclude (5.21). □

The unique reconstruction of $\widetilde{V}$ in (5.6) and (5.13) amounts to investigating the dimension of the spaces R_{S} and R_{W} respectively. This theorem suggests that the latter space is

a subspace of the former, meaning the unique reconstruction of (5.13) would indicate the unique reconstruction of (5.6).

5.3.2 From Wigner to Liouville in the inverse setting

According to Theorem 2.2, the Liouville equation is the classical limit of the Wigner equation, meaning that f^ε, which solves the Wigner equation, converges to f, which solves the Liouville equation, when $\varepsilon \to 0$. We expect similar argument holds true in the inverse setting as well. This amounts to study the two representatives $R^\varepsilon_{\mathrm{W}}$ and R_{L}.

Theorem 5.2. *Let $R^\varepsilon_{\mathrm{W}}[f^\varepsilon_{\mathrm{I}}, g^\varepsilon_T]$, and $R_{\mathrm{L}}[f_{\mathrm{I}}, g_T]$ be the representatives defined in* (5.14) *and* (5.20) *respectively, where*

$$f_{\mathrm{I}} = \lim_{\varepsilon\to 0} f^\varepsilon_{\mathrm{I}}\,, \quad g_T = \lim_{\varepsilon\to 0} g^\varepsilon_T\,,$$

then we claim:

$$\lim_{\varepsilon\to 0} R^\varepsilon_{\mathrm{W}}[f^\varepsilon_{\mathrm{I}}, g^\varepsilon_T] = R_{\mathrm{L}}[f_{\mathrm{I}}, g_T]\,. \tag{5.26}$$

Proof. Suppose $f^\varepsilon_{\mathrm{b}}$ solves the background Wigner equation (5.9) with the initial data $f^\varepsilon_{\mathrm{I}}$, and f_{b} solves the background Liouville equation (5.16) with the initial data f_{I}. In the semi-classical regime $\varepsilon \to 0$, by Theorem 2.2, we know that the background wave $f^\varepsilon_{\mathrm{b}}$ converges to f_{b}. Thus, we have, formally,

$$\begin{aligned}\lim_{\varepsilon\to 0} D^\varepsilon f^\varepsilon_{\mathrm{b}}(z,k,p) &= \lim_{\varepsilon\to 0} \frac{1}{\varepsilon}\left[f^\varepsilon_{\mathrm{b}}\left(z, k+\frac{1}{2}\varepsilon p\right) - f^\varepsilon_{\mathrm{b}}\left(z, k-\frac{1}{2}\varepsilon p\right)\right] \\ &\overset{\varepsilon\to 0}{\to} p\cdot\nabla_k f_{\mathrm{b}}(z,k)\,.\end{aligned} \tag{5.27}$$

Similarly, suppose g^ε solves the adjoint Wigner equation (5.11) with the final data g^ε_T, and g solves the adjoint Liouville equation (5.18) with the final data g_T, then the adjoint wave

g^ε converges to g. Combining these:

$$\begin{aligned} R^\varepsilon_{\mathrm{W}}[f^\varepsilon_{\mathrm{I}}, g^\varepsilon_T](x) &= \frac{\mathrm{i}}{(2\pi)^d} \int_{\mathbb{R}^{3d}\times[0,T]} e^{\mathrm{i}p(z-x)} \overline{g^\varepsilon}(z,k) D^\varepsilon f^\varepsilon_{\mathrm{b}}(z,k,p) \mathrm{d}p\mathrm{d}z\mathrm{d}k\mathrm{d}t \\ &\overset{\varepsilon\to 0}{\to} \frac{\mathrm{i}}{(2\pi)^d} \int_{\mathbb{R}^{3d}\times[0,T]} e^{\mathrm{i}p(z-x)} \overline{g}(z,k) p\cdot\nabla_k f_{\mathrm{b}}(z,k) \mathrm{d}p\mathrm{d}z\mathrm{d}k\mathrm{d}t\,. \end{aligned} \tag{5.28}$$

Integrating by parts for the limit and applying the Fourier inversion formula (5.24) lead to the Liouville representative, that is, the right hand side of the last limit becomes:

$$\begin{aligned} &-\frac{1}{(2\pi)^d} \int_{\mathbb{R}^{3d}\times[0,T]} e^{\mathrm{i}p(z-x)} \nabla_z \cdot (\overline{g^\varepsilon}(z,k)\nabla_k f_{\mathrm{b}}(z,k)) \mathrm{d}p\mathrm{d}z\mathrm{d}k\mathrm{d}t \\ =& -\nabla_x \cdot \int_{\mathbb{R}^d\times[0,T]} \overline{g}(x,k)\nabla_k f_{\mathrm{b}}(x,k) \mathrm{d}k\mathrm{d}t = R_{\mathrm{L}}[f_{\mathrm{I}}, g_T]\,, \end{aligned} \tag{5.29}$$

which concludes (5.26). □

The proof above is formal. We assumed enough regularity for the convergence (5.27). We also need the convergence to hold true in the strong sense (in L^2 for example).

This theorem suggests that the inverse problem of the Wigner equation, in the classical regime with $\varepsilon \to 0$, is asymptotically equivalent to the inverse problem of the Liouville equation. This suggests the connection between the Schrödinger equation and the Newton's law of motion in the inverse setting: if the reconstruction of the potential term using the initial-to-final data map is unique (up to a gauge transform) and stable for the Schrödinger equation, it is expected that the same should hold true for the Newton's law of motion.

5.4 Numerical results

As a proof of concept, we provide numerical evidences for the Wigner inverse problem (5.13) in the classical regime.

5.4.1 Numerical setup

In $(1+1)$-dimensional space, the background Wigner equation reads

$$\begin{aligned} &\partial_t f_{\mathrm{b}}^{\varepsilon} + k\partial_x f_{\mathrm{b}}^{\varepsilon} = \frac{\mathrm{i}}{2\pi\varepsilon}\int_{\mathbb{R}^2}\left[V_{\mathrm{b}}\left(x+\frac{\varepsilon y}{2}\right) - V_{\mathrm{b}}\left(x-\frac{\varepsilon y}{2}\right)\right] f_{\mathrm{b}}^{\varepsilon}(x,p)e^{\mathrm{i}y(k-p)}\mathrm{d}y\mathrm{d}p\,, \\ &f_{\mathrm{b}}^{\varepsilon}(0,x,k) = f_{\mathrm{I}}^{\varepsilon}(x,k)\,, \end{aligned} \tag{5.30}$$

and its adjoint equation reads

$$\begin{aligned} &\partial_t g^{\varepsilon} + k\partial_x g^{\varepsilon} = \frac{\mathrm{i}}{2\pi\varepsilon}\int_{\mathbb{R}^2}\left[V_{\mathrm{b}}\left(x+\frac{\varepsilon y}{2}\right) - V_{\mathrm{b}}\left(x-\frac{\varepsilon y}{2}\right)\right] g(x,p)e^{\mathrm{i}y(k-p)}\mathrm{d}y\mathrm{d}p\,, \\ &g^{\varepsilon}(T,x,k) = g_T^{\varepsilon}(x,k)\,. \end{aligned} \tag{5.31}$$

The corresponding Liouville equation for the background equation and the adjoint equation are

$$\begin{aligned} &\partial_t f_{\mathrm{b}} + k\partial_x f_{\mathrm{b}} - \partial_x V_{\mathrm{b}}\partial_k f_{\mathrm{b}} = 0\,, \\ &f_{\mathrm{b}}(0,x,k) = f_{\mathrm{I}}(x,k)\,, \end{aligned} \tag{5.32}$$

and

$$\begin{aligned} &\partial_t g + k\partial_x g - \partial_x V_{\mathrm{b}}\partial_k g = 0\,, \\ &g(T,x,k) = g_T(x,k)\,, \end{aligned} \tag{5.33}$$

respectively. According to the definitions (5.14) and (5.20), the Wigner and Liouville representatives in the inverse problems are

$$\begin{aligned} &R_{\mathrm{W}}^{\varepsilon}[f_{\mathrm{I}}^{\varepsilon}, g_T^{\varepsilon}](x) \\ &= \frac{\mathrm{i}}{2\pi}\int_{\mathbb{R}^3\times[0,T]} e^{\mathrm{i}p(z-x)} g^{\varepsilon}(z,k)\frac{1}{\varepsilon}\left[f_{\mathrm{b}}^{\varepsilon}\left(z,k+\frac{\varepsilon p}{2}\right) - f_{\mathrm{b}}^{\varepsilon}\left(z,k-\frac{\varepsilon p}{2}\right)\right]\mathrm{d}p\mathrm{d}z\mathrm{d}k\mathrm{d}t\,, \end{aligned}$$

and

$$R_{\mathrm{L}}[f_{\mathrm{I}}, g_T](x) = -\partial_x\int_{\mathbb{R}^d\times[0,T]} g(x,k)\partial_k f_{\mathrm{b}}(x,k)\mathrm{d}k\mathrm{d}t\,,$$

where $f_{\mathrm{b}}^{\varepsilon}$, g^{ε}, f_{b} and g satisfy the equations above. We are to demonstrate the relation between the two representatives as $\varepsilon \to 0$.

To set up the experiment, we choose the background potential $V_{\mathrm{b}}(x)$ to have a Gaussian

form

$$V_{\rm b}(x) = A \exp\left(-\frac{(x-a)^2}{w^2}\right), \tag{5.34}$$

and the initial and final time conditions are

$$f_{\rm I}^{\varepsilon}(x,k) = f_{\rm I}(x,k) = B \exp\left(-\frac{(x-b_x)^2}{\sigma_x^2} - \frac{(k-b_k)^2}{\sigma_k^2}\right), \tag{5.35}$$

and

$$g_T^{\varepsilon}(x,k) = g_T(x,k) = C \exp\left(-\frac{(x-c_x)^2}{\delta_x^2} - \frac{(k-c_k)^2}{\delta_k^2}\right). \tag{5.36}$$

To compute the Wigner equation (5.30) and (5.31), we truncate the computational domain to $\Omega = [0, L] \times [K_1, K_2]$ and apply periodic boundary condition on x. The time interval is taken to be $[0, T]$. The transport term is discretized by a fifth-order WENO scheme [137], and the collision term is computed by the trapezoidal approximate [56].

To compute the Liouville equation (5.32) and (5.33), we use the particle method. This is to solve the ODE systems of trajectories. For example, to compute (5.32) for $0 \le t \le T$, we set the trajectory equation

$$\dot{x} = -k\,, \quad \dot{k} = \partial_x V_{\rm b}(x)\,, \quad \text{with} \quad x(0) = y\,,\ k(0) = p\,,$$

and the initial data for the particle (y, p) is determined by $f_{\rm I}$. The final solution is thus $f_{\rm b}(T, y, p) = f_{\rm I}(x(T), k(T))$.

5.4.2 Numerical examples

In the numerical examples, we set the parameters in (5.34) to be

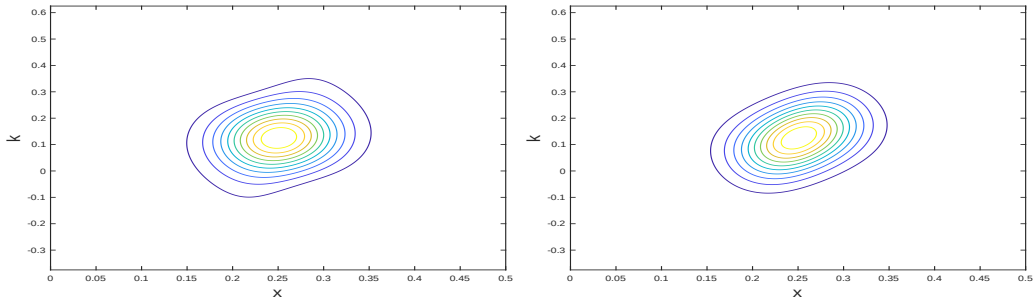
$$A = 1\,, \quad a = 0.25\,, \quad w = 2^{-3}\,,$$

and the parameters defined in (5.35) and (5.36) are

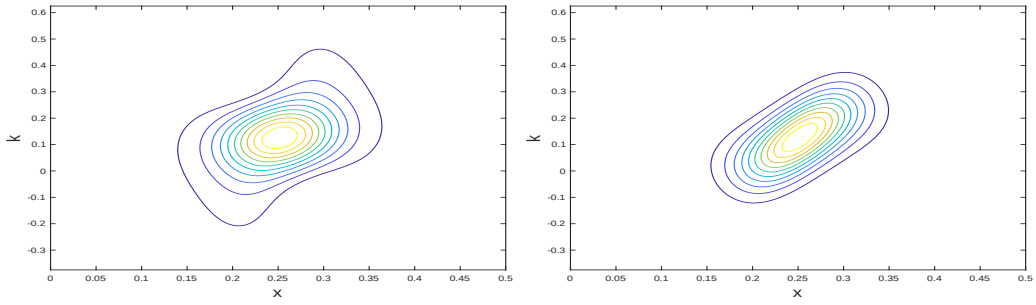
$$B = C = 1\,, \quad \sigma_x = \delta_x = 2^{-4}\,, \quad \sigma_k = \delta_k = 2^{-3}\,, \quad b_k = c_k = 2^{-3}\,.$$

For discretization, we use $\Delta x = 2^{-10}$, $\Delta k = 2^{-10}$ and $\Delta t = 2^{-10}$ in both the Wigner and the Liouville solver. In the Wigner solver, we set $L = 0.5$, $K_1 = -0.375$, $K_2 = 0.625$. The terminal time is set to be $T = 2^{-6}$.

In Figure 5.1 and Figure 5.2, we first plot the level sets of solutions $f_{\mathrm{b}}^{\varepsilon}$ and g^{ε} at different time.



(a) $t = 2^{-7}$



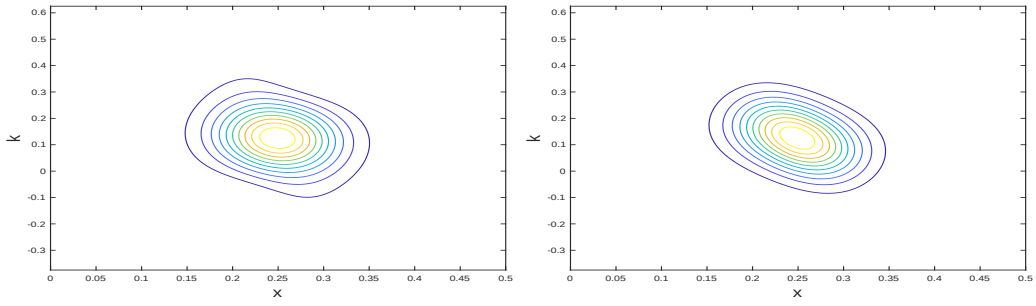
(b) $t = 2^{-6}$

Figure 5.1: The left column shows the contour of $f_{\mathrm{b}}^{\varepsilon}$ for $\varepsilon = \pi^{-1}2^{-4}$ and the right column shows the contour for $\varepsilon = \pi^{-1}2^{-8}$.

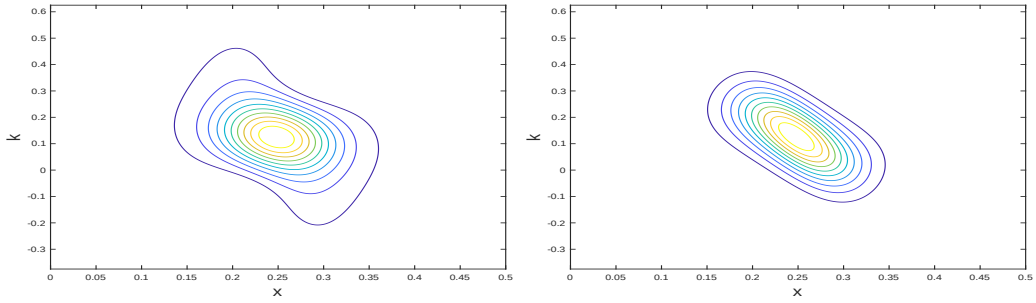
We then compare the two representatives $R_{\mathrm{W}}^{\varepsilon}[f_{\mathrm{I}}^{\varepsilon}, g_T^{\varepsilon}]$ and $R_L[f_{\mathrm{I}}, g_T]$ for two different configurations of (b_x, c_x). As shown in the left column of Figure 5.3, with $\varepsilon \to 0$, the profile of $R_{\mathrm{W}}^{\varepsilon}[f_{\mathrm{I}}^{\varepsilon}, g_T^{\varepsilon}]$ gets closer and closer to that of $R_{\mathrm{L}}[f_{\mathrm{I}}, g_T]$ for both examples. To quantify the convergence, we define

$$\mathrm{Err}_{\mathrm{R}}(\varepsilon) = \frac{\|R^{\varepsilon}[f_{\mathrm{I}}^{\varepsilon}, g_T^{\varepsilon}] - R[f_{\mathrm{I}}, g_T]\|_{L^2(\mathbb{R}^d)}}{\|R[f_{\mathrm{I}}, g_T]\|_{L^2(\mathbb{R}^d)}},$$

and plot the convergence rate with respect to ε, as shown in the right column of Figure 5.3. In both examples, the plots suggest a decay rate of $O(\varepsilon^2)$.

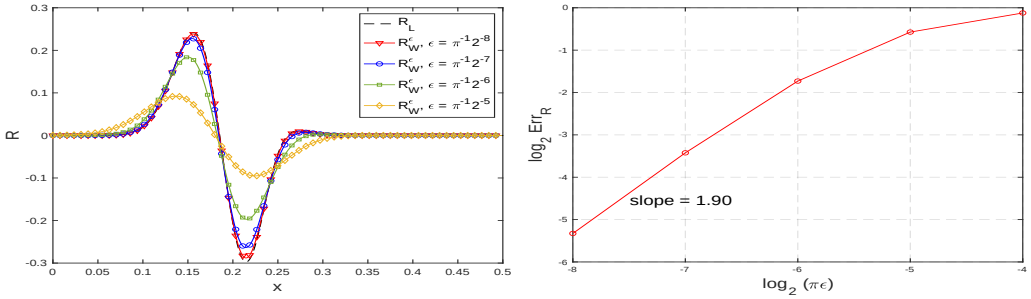


(a) $t = 2^{-7}$

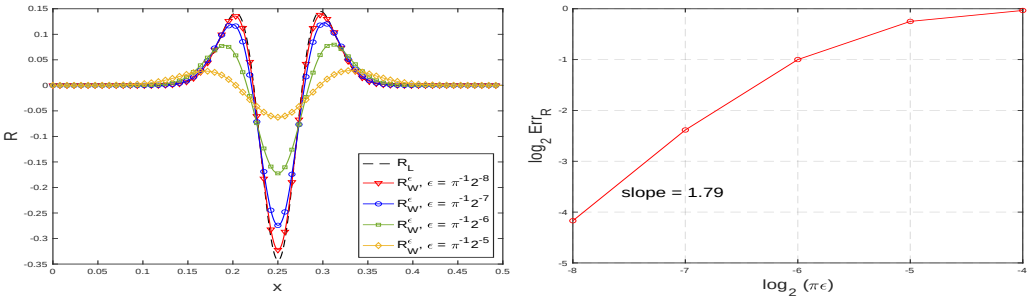


(b) $t = 0$

Figure 5.2: The left column shows the contour of g^{ε}, the solution to the adjoint equation (5.31) that propagates backwards in time, for $\varepsilon = \pi^{-1}2^{-4}$ and the right column shows the contour for $\varepsilon = \pi^{-1}2^{-8}$.



(a) $b_x = c_x = 0.1875$



(b) $b_x = c_x = 0.25$

Figure 5.3: The left column compares the Wigner representative $R^{\varepsilon}_{\mathrm{W}}[f^{\varepsilon}_{\mathrm{bI}}, g^{\varepsilon}_{T}]$ with different values of ε and the limiting Liouville representative $R_{\mathrm{L}}[f_{\mathrm{bI}}, g_T]$. The right column shows $\mathrm{Err}_{\mathrm{R}}(\varepsilon)$ as a function of ε. The decay rate suggests that $\mathrm{Err}_{\mathrm{R}}(\varepsilon)$ is of $O(\varepsilon^2)$.

We finally demonstrate the convergence for a large set of basis functions. To do so, we first take the interval $[x_l, x_r] = [0.1875, 0.3125]$, and denote the discrete points in the interval $x_i = x_l + i\Delta x$, with $i \in \mathbb{N}$ and $x_l \leq x_i \leq x_r$. Considering $\Delta x = 2^{-10}$, we have $N = 129$ configurations of x_i. We then let b_x and c_x, the centers for $f_{\mathrm{I}}^{\varepsilon}$ and g_T^{ε} taking these configurations. The combination provides us a large set of initial/final time data $f_{\mathrm{I},i}^{\varepsilon}$ and $g_{T,j}^{\varepsilon}$. We compute the corresponding solutions, termed f_i^{ε} and g_j^{ε} and formulate a set:

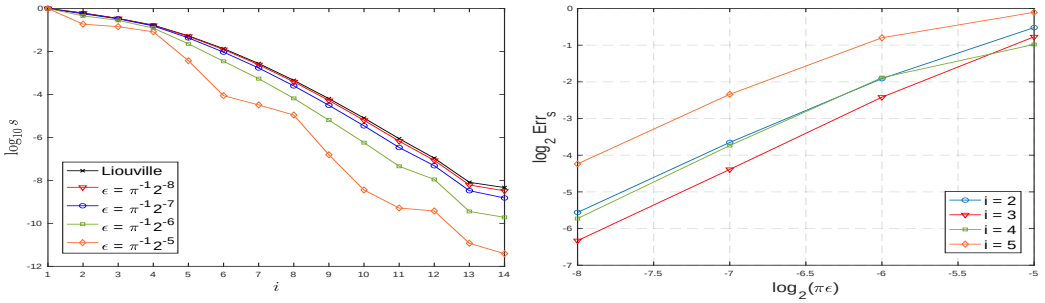
$$\mathcal{R}_{\mathrm{W}}^{\varepsilon} = \{R_{\mathrm{W},ij}^{\varepsilon} = R_{\mathrm{W}}^{\varepsilon}[f_{\mathrm{I},i}^{\varepsilon}, g_{T,j}^{\varepsilon}],\ i,j = 0, \dots, 128\}.$$

The same process is done to obtain $R_{\mathrm{L},ij}$ and the set $\mathcal{R}_{\mathrm{L}}$.

We now compare the set $\mathcal{R}_{\mathrm{W}}^{\varepsilon}$ and $\mathcal{R}_{\mathrm{L}}$. We first compare the singular values of the two sets. In Figure 5.4a, we plot the relative singular value decay of both $\mathcal{R}_{\mathrm{W}}^{\varepsilon}$ at different values of ε, and $\mathcal{R}_{\mathrm{L}}$. As $\varepsilon \to 0$, it is clear the decay profile converges. We also quantify the convergence of relative singular value using the following error term:

$$\mathrm{Err}_{s,i}(\varepsilon) = \frac{|s_i^{\varepsilon} - s_i|}{|s_i|},$$

where s_i^{ε} is the ith relative singular value of $\mathcal{R}_{\mathrm{W}}^{\varepsilon}$, and s_i is the ith relative singular value of $\mathcal{R}_{\mathrm{L}}$. In Figure 5.4b, we plot $\mathrm{Err}_{s_i}(\varepsilon)$ as a function of ε for $i = 2, 3, 4, 5$. It is clear that the relative singular values of $\mathcal{R}_{\mathrm{W}}^{\varepsilon}$ converge to their counterparts in the $\varepsilon \to 0$ classical limit.



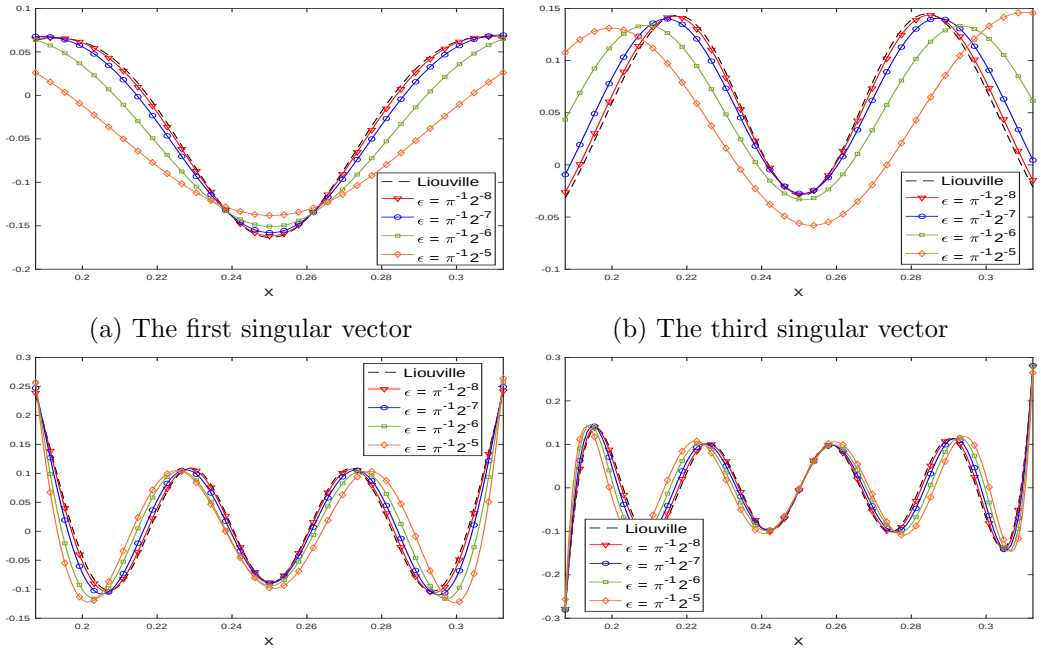
(a) (b)

Figure 5.4: (a) The relative singular values of $\mathcal{R}_{\mathrm{L}}$ and $\mathcal{R}_{\mathrm{W}}^{\varepsilon}$ at different values of ε. (b) $\mathrm{Err}_{s_i}(\varepsilon)$ as a function of ε for the 2nd to the 5th relative singular values.

We then compare the left singular vectors of the basis. In Figure 5.5, we show the first, third, seventh and tenth left singular vectors of $\mathcal{R}^{\varepsilon}_{\mathrm{W}}$. As $\varepsilon \to 0$, the profiles converge to those of $\mathcal{R}_{\mathrm{L}}$. To quantify such convergence, we let Q^{ε}_k and Q_k to denote the column spaces (orthonormalized) spanned by the first k left singular vectors of $\mathcal{R}^{\varepsilon}_{\mathrm{W}}$ and $\mathcal{R}_{\mathrm{L}}$, respectively, and define the angle between the spaces:

$$\mathrm{Err}_{\mathcal{R},k} = \|Q_k - Q^{\varepsilon}_k (Q^{\varepsilon}_k)^{\top} Q_k\|_2 \,.$$

The angle between the two spaces are shown to converge as $\varepsilon \to 0$ for different values of k in Figure 5.6.



(a) The first singular vector

(b) The third singular vector

(c) The seventh singular vector

(d) The tenth singular vector

Figure 5.5: The singular vectors of $\mathcal{R}_{\mathrm{L}}$ and $\mathcal{R}^{\varepsilon}_{\mathrm{W}}$ at different values of ε.

5.5 Conclusion

It is a well-known result that the Schrödinger equation leads to the Newton's second law in the classical limit, when the rescaled Planck constant $\varepsilon \to 0$. We investigate this

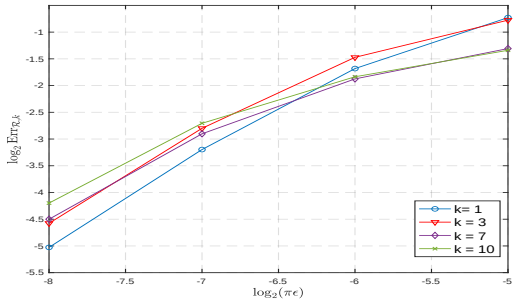


Figure 5.6: $\mathrm{Err}_{\mathcal{R},k}$ as a function of ε for $k = 1, 3, 7, 10$.

limit in the inverse setting. More specifically, we assume the initial and final data is available and we study if the initial-final data pairs can reconstruct the potential term in the Schrödinger equation. The investigation is done in the linearized setting, assuming the potential is close enough to a background, and this boils the problem down to the study of the representative of the associated Fredholm integral derived from the inverse problem.

We employ the Wigner transform tool. In particular, we translate the information of the Schrödinger equation to that of the Wigner equation, and pass its limit to obtain the Liouville equation which presents particle trajectories following the classical mechanics. We are able to show that the representative derived under the Wigner framework indeed converges to the representative derived under the Liouville framework when $\varepsilon \to 0$, and thus we link the inverse Schrödinger problem with the inverse Newton's law of motion.

Chapter 6

High-frequency limit of the Helmholtz inverse scattering problem

In this chapter, we investigate the asymptotic relation between the inverse problems relying on the Helmholtz equation and the radiative transfer equation as physical models, in the high-frequency limit. In particular, we evaluate the asymptotic convergence of a generalized version of the inverse scattering problem based on the Helmholtz equation, to the inverse scattering problem of the Liouville equation (a simplified version of RTE). The two inverse problems are connected through the Wigner transform that translates the wave-type description on the physical space to the kinetic-type description on the phase space, and the Husimi transform that models data localized both in location and direction. The finding suggests that impinging tightly concentrated monochromatic beams can indeed provide stable reconstruction of the medium, asymptotically in the high-frequency regime. This fact stands in contrast with the unstable reconstruction for the classical inverse scattering problem when the probing signals are plane-waves.

6.1 Introduction

The wave-particle duality of light has been one of the greatest enigmas in the natural sciences, dating back to Euclid's treatise in light, *Catoptrics* (280 B.C.) and spanning more than two millennia. In a nutshell, light can be either described as an electromagnetic (EM) wave governed by the Maxwell's equations, or as a stream of particles, called photons, governed by the radiative transport equation (RTE).

Although the advent of quantum mechanics at the onset of the last century partially solved the riddle, due to computational considerations, light continues to be modeled either as a particle or as a wave depending on the target application. Among those applications, inverse problems are perhaps the ones that have gained the most attention in the last decades, which in return have fueled many breakthroughs in telecommunications [211, 212], radar [74], biomedical imaging [195, 28] and, more recently, in chip manufacturing [145]. In this context, inverse problems can be roughly described as reconstructing unknown parameters within a domain of interest by data comprised of observations on its boundary.

Unfortunately, the properties of the inverse problems are highly dependent on the specific modeling of the underlying physical phenomena, even though, in principle, they share the same microscopic description. In particular, the stability of the inverse problem, i.e., how sensitive is the reconstruction of the unknown parameter to perturbations in the data, is surprisingly disparate [176, 66], thus creating an important gap between the wave and particle descriptions, which we seek to bridge in this chapter. We point out that understanding this gap is not only of theoretical importance, it would also play an important role in designing new reconstruction algorithms with improved stability applicable to a broader set of wave-based inverse problems, which are ubiquitous in science [202, 189, 179] and engineering [184, 14, 82].

For simplicity, we consider a time-harmonic wave-like description governed by the Helmholtz equation, which can be derived from the time-harmonic Maxwell's equations after some simplifications. Alternatively, the Helmholtz equation can also be obtained by computing the Fourier transform of the constant-density acoustic wave equation at

frequency k, and is given by[1]

$$\left(\Delta + k^2 n\right) u(x) = 0\,, \tag{6.1}$$

where u is the wave field, and $n(x)$ is the refractive index of the medium. We point out that even if this is a simplified model, it retains the core difficulty of more complex physics.

We also consider a particle-like description governed by the Liouville equation, which is a simplified RTE, given by:

$$v \cdot \nabla_x f - \nabla_x n \cdot \nabla_v f = 0\,, \tag{6.2}$$

where $f(x, v)$ is the distribution of photon particles, and n is still the refractive index. The Liouville equation describes the trajectories of photons via its characteristics: $\dot{x} = v$ and $\dot{v} = -\nabla_x n$. For simplicity we neglect the photon interactions which are usually encoded by the collision operator.

Following the wave and photon descriptions, we define the forward problem as calculating either the wave-field, or the photon distribution from the refractive index by solving either the Helmholtz or the Liouville equations. The wave-particle duality, when translated to mathematical language, corresponds to the fact that the solutions obtained by the Helmholtz and Liouville equations are asymptotically close when $k \to \infty$, see [23].

For the sake of conciseness, we consider a simplified inverse problem consisting of reconstructing an unknown environment within a domain of interest by probing it with tightly concentrated monochromatic beams originated from the boundary of the domain, in which the response of the unknown medium to the impinging beam is measured at its boundary. This measurement is performed by a measurement operator that is model-specific and it will play an important role in what follows. For simplicity, we consider the full aperture regime, i.e., we can probe the medium from any direction, and we sample its impulse response in all possible directions. When the beam is modeled as a wave, i.e., using the Helmholtz equation as a forward model, this process can be considered as

[1]The domain of definition, source, and boundary conditions will be specified in Section 6.2.

a *generalized* version of the inverse scattering wave problem (which we, for the sake of clarity, just refer to as the *generalized Helmholtz scattering problem*. When the beam is modeled as a flux of photons, i.e., using the Liouville equation as a forward model, this process is often referred to as the *optical tomography problem*, but we will refer to as the *Liouville scattering problem* in this chapter.

Although the two different formulations seek to solve the same underlying physical problem, our understanding of the two inverse problems seems to suggest different stability properties. The traditional inverse scattering problem, using either near-field or far-field data is ill-posed: small perturbations in the measurements usually lead to large deviations in the reconstructions [83, 120]. Thus, sophisticated algorithms [160, 51, 50, 185, 72, 48, 31] have been designed to artificially stabilize the process by appropriately restricting the class of possible unknown environments, usually in the form of band-limited environments. Conversely, the inverse Liouville equation is well-conditioned: a small perturbation is reflected by a small error in the reconstruction [178].

Thus the observation that the stability for both problems is different seems to be at odds with the fact that the Liouville equation and the Helmholtz equation are asymptotically close in the high-frequency regime. Fortunately, as what we will see, this somewhat contradictory property stems from the inability of *traditional* formulations of the inverse problems to agree in the high-frequency limit. When the measurement operators are accordingly adjusted, we show that the new formulations, which we call the *generalized* inverse scattering, are equivalent in the limit as $k \to \infty$, producing a stable inverse problem. The convergence from the Helmholtz equation to the Liouville equation is conducted through the Wigner transform [109, 193, 23], and the convergence of the measuring operators is achieved through the Husimi transform [42]. Both convergences are obtained asymptotically in the $k \to \infty$ limit. This convergence allows us to conclude the following:

The inverse Liouville scattering problem is asymptotically equivalent to the generalized inverse Helmholtz scattering problem in the high-frequency regime.

The current chapter is dedicated to formulating the statement above in a mathe-

matically precise manner, while providing extensive numerical evidence supporting the statement.

On the mathematical level, the current chapter carries the following important features:

- The result connects the two seemingly distinct inverse problems, and suggests that in the high-frequency regime, probing an unknown object with a single frequency is already enough for its reconstruction, with properly prepared data in the generalized inverse scattering setting. This partially answers the stability question regarding the inverse scattering.

- The result can be viewed as the counterpart of the asymptotic multiscale study conducted in the forward setting. In particular, semi-classical limit is a theory that connects quantum mechanical and the classical mechanical description: the proposed formulation for the inverse scattering problem can be regarded as taking the (semi-)classical limit in the inverse setting, and thus the work carries conceptual merits. This is in line with [176, 66]. See also [153] for a different setting.

These mathematical understandings also naturally bring numerical and practical benefits. The new inverse wave scattering formulation coupled with PDE-constrained optimization seems to be empirically less prone to cycle-skipping, i.e., convergence to spurious local minima [205], than its standard counterparts [206, 48], thus potentially opening the way to more robust algorithmic pipelines for inverse problems.

We point out that even though this current study is motivated by the wave-particle duality of light, the current results are also applicable to other oscillatory phenomena, see [66] for a discussion on inverse Schrödinger problem in the classical limit.

Organization

In Section 6.2, we briefly review the Helmholtz equation and present the corresponding inverse problem that fits the particular experimental setup that allows passing the system to the $k \to \infty$ limit. In Section 6.3, we discuss the limiting Liouville equation and the

inverse Liouville scattering problem, by conducting the Wigner and Husimi transforms. The connections between the two inverse problems will thus be immediate. Finally, we present our numerical evidences that justify the convergence in Section 6.4 and we showcase the stability of the inverse problem in Section 6.5.

6.2 Experimental setup and inverse problem formulation

Suppose we use tightly concentrated monochromatic beams, or laser beams, to probe the medium. Each beam impinges in the area of interest, thus producing a scattered field which is then measured by directional receivers[2] placed on a manifold around the domain of interest (See Figure 6.1). The data, which is used to reconstruct the optical properties of the medium, is the intensity captured by each receiver for each incoming beam. Thus, the data is indexed by the position and direction of the impinging beam, and the location and direction of the receivers.

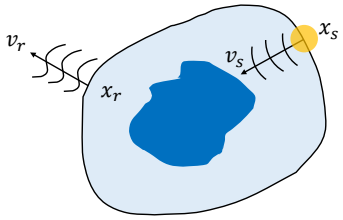


Figure 6.1: Illustration of the setup. Here x_{s} and v_{s} denote the location and direction of the incident beam, respectively, while x_{r} and v_{r} denote the location and direction of the receiver, respectively.

In this section, we set up the experiment and provide the mathematical formulation, using both the wave and the particle forms for the forward model. This prepares us to link the two problems in Section 6.3.

6.2.1 Helmholtz equation and inverse wave scattering problem

The Helmholtz equation is a model equation for time-harmonic wave propagation. After some approximations, both the constant-density acoustic wave equation and the Maxwell

[2]Experimentally, this is often achieved by placing a collimator before the receiver, and changing the orientation of the collimator.

equations for the EM waves, can be recast, through the Fourier transform in t, into the Helmholtz equation. It writes as:

$$\Delta u^k + k^2 n(x) u^k = S^k(x)\,. \tag{6.3}$$

In the equation, u^k is the wave-field, with the superscript, $k > 0$ represents the wave number (that carries the frequency information, and thus in the chapter we use the two words interchangeably). $n(x)$ is a complex-valued refractive index having non-negative imaginary part, $\mathrm{Im}(n(x)) \geq 0$, reflecting the heterogeneity of the medium. We assume $n(x)$ is the constant one in all $\mathbb{R}^d$ except in a convex bounded open set $\Omega \subset \mathbb{R}^d$, meaning $\mathrm{supp}(n-1) \subset \Omega$. In order to streamline the notation, we let $\Omega = B_1$, the ball with radius 1 centered around the origin. The right-hand side $S^k(x)$ is the source term, which is wave-number dependent.

The classical setup for the scattering problem is to probe the medium with an incident wave-field $u^{i,k}$ that triggers the response from the medium. Noting that the total field, which satisfies (6.3), is the sum of the incident and the scattered wave-fields, we can write:

$$u^k = u^{i,k} + u^{s,k}\,,$$

and derive the equation for the scattered wave-field $u^{s,k}$. Suppose the incident wave is designed so that it absorbs all the external source information:

$$\Delta u^{\mathrm{i},k} + k^2 u^{\mathrm{i},k} = S^k(x)\,, \tag{6.4}$$

then by simply subtracting it from (6.3), we have the equation for $u^{s,k}$:

$$\begin{aligned} \Delta u^{\mathrm{s},k} + k^2 n(x) u^{\mathrm{s},k} &= k^2 (1 - n(x)) u^{\mathrm{i},k} \quad x \in \mathbb{R}^d\,, \\ \frac{\partial u^{\mathrm{s},k}}{\partial r} - \mathrm{i}k u^{\mathrm{s},k} &= \mathcal{O}(r^{-(d+1)/2}) \text{ as } r = |x| \to \infty\,. \end{aligned} \tag{6.5}$$

In this equation, we can view the incident wave $u^{\mathrm{i},k}$ impinging in the perturbation $n - 1$

as the source term for $u^{\mathrm{s},k}$. Clearly, this source term $k^2(1-n(x))u^{\mathrm{i},k}$ is zero outside B_1, the support of $n-1$. The Sommerfeld radiation condition is imposed at infinity to ensure the uniqueness for $u^{\mathrm{s},k}$.

When $d=3$, a typical approach is to set $S(x)=\delta_y$, a point Dirac delta, then the solution $u^{\mathrm{i},k}$ to (6.4) becomes the fundamental solution to the homogeneous Helmholtz equation in $\mathbb{R}^3$

$$\Phi(x;y) = -\frac{1}{4\pi}\frac{\exp(ik|x-y|)}{|x-y|}, \quad x,y\in\mathbb{R}^3, x\neq y\,,$$

for any given y. We can clearly observe that the function is radially symmetric centered in y thus it is often termed a spherical wave. If $|y|\gg|x|$, i.e., y is far away from the origin, we have the far-field regime, in which the fundamental solution is approximately a plain wave: $\Phi(x;y)\approx -\frac{e^{ik|y|}}{4\pi|y|}\exp(-ik\hat{y}\cdot x)$ with $\hat{y}=\frac{y}{|y|}$ being the unit vector.

In this case, however, instead of using the Dirac delta, we handcraft a specially designed source term, which will be crucial for the re-scaling proposed in this chapter. In particular, we choose $S^k_{\mathrm{H}}(x)$ to be the following:

$$S^k_{\mathrm{H}}(x;x_s,v_s) = -k^{\frac{3+d}{2}}S_{v_s}(k(x-x_s)) \quad x\in\mathbb{R}^d\,, \tag{6.6}$$

where the subscript H stands for Helmholtz, and

$$S_{v_s}(x) = C(\sigma,d)\exp\left(-\sigma^2\frac{|x|^2}{2}+\mathrm{i}v_s\cdot x\right)\,. \tag{6.7}$$

Here $C(\sigma,d)$ is the normalization constant $C(\sigma,d)=\sqrt{2}\left(\frac{\sigma}{\sqrt{\pi}}\right)^{\frac{d+1}{2}}$.

Physically this source term can be understood as the source generating a tight beam being shone onto the medium from the location x_s in the direction of v_s. The profile of this tight beam, or "laser beam", is a Gaussian centered around the light-up location x_s and the width of the Gaussian is characterized by $(k\sigma)^{-1}$. With σ fixed, as $k\to\infty$, the beam is more and more concentrated.

Following the explanation above we incorporate the source term in (6.6) into (6.3)-

(6.4), to probe the medium from the positions, x_s, in the direction of v_s, that are physically pertinent. In particular, we let $(x_s, v_s) \in \Gamma_-$ where

$$\Gamma_\pm = \{(x, v) \in \partial B_1 \times \mathbb{S}^{d-1} : \pm v \cdot \nu(x) > 0\}.$$

In which, $\nu(x)$ denotes the outer-normal direction at $x \in \partial B_1$. This means the laser beams shine from the boundary of B_1 in the direction v pointing inward the interior of the domain.

From (6.6) we can observe that as $k \to \infty$, the laser beam becomes increasingly concentrated. In particular, in the $k \to \infty$ limit, the incident wave $u^{\mathrm{i},k}$ becomes a ray, propagating through a straight line[3].

As usual in inverse problems (in particular, in non-intrusive experimental setups), we take measurements of u^k near the boundary ∂B_1. To take such measurement we design a family of test functions of the form:

$$\phi_v^k(x) = k^{d/4} \chi\left(\sqrt{k}x\right) e^{-\mathrm{i}kv\cdot x}, \tag{6.8}$$

where $\chi : \mathbb{R}^d \to \mathbb{R}$ is a smooth radially symmetric function that vanishes as $|x| \to \infty$.

We define the measurement of u^k as its Husimi transform

$$H^k u^k(x, v) = \left(\frac{k}{2\pi}\right)^d \left|u^k * \phi_v^k\right|^2 \qquad \text{for } (x, v) \in \Gamma_+. \tag{6.9}$$

The measurement then consists of the intensity of the field that convolves with the test function. This measurement is conducted only on the boundary, and only in the directions pointing outside the domain.

This measurement has a clear physical interpretation: it measures the intensity of the wave-field at location x propagating in direction v, using χ as the impulse response of the receiver, or test function.

[3]The incoming ray propagates in a straight line due to the assumption that the background is constant. Otherwise, the ray would bend if a smooth non-constant background is considered.

One typical choice for the family of test functions is to set χ as a Gaussian (normalized in L^2 norm)

$$\chi(x) = \left(\frac{1}{\pi}\right)^{d/4} \exp\left(-\frac{|x|^2}{2}\right) . \tag{6.10}$$

It is straightforward to see that as $k \to \infty$, the test function ϕ_v^k concentrates around zero due to the $\sqrt{k}$ scaling. As such, the measurement $u^k * \phi_v^k$ at a location x_s only takes value of u^k in a very small neighborhood around x_s.

Remark 6.1. *We note that the choice of χ in* (6.10) *is not essential. We use this specific form to make the calculation explicit, as it will be shown in 6.1. Other forms of χ would also work well as long as the corresponding $G^k = W^k[\phi_0^k]$ converges to a Dirac delta when $k \to \infty$, as it will be explained in 6.5.*

Forward Map: now we have all the elements to define the forward map. For any $(x_s, v_s) \in \Gamma_-$, we shine laser beam into B_1 according to the format in (6.6), then the solution to the Helmholtz equation (6.3), u^k is tested by $\phi_v^k(x)$ and evaluated on Γ_+:

$$\Lambda_n^k : \quad S_{\mathrm{H}}^k(x; x_s, v_s) \to H^k u^k(x_r, v_r)|_{\Gamma_+} . \tag{6.11}$$

As a consequence, the dataset generated by this forward map is the collection of:

$$\mathcal{D}^k[n] = \left\{ \left(S_{\mathrm{H}}^k(x; x_s, v_s), \Lambda_n^k[S_{\mathrm{H}}^k](x_r, v_r) \right) : (x_s, v_s) \in \Gamma_-, (x_r, v_r) \in \Gamma_+ \right\} . \tag{6.12}$$

We now formulate the generalized inverse scattering problem as

$$\textit{to reconstruct } n \textit{ using the information in } \mathcal{D}^k[n]. \tag{6.13}$$

Traditional inverse scattering problem

Given that we use a non-standard formulation of the inverse scattering problem, we will stress a couple of similarities and differences between the generalized and classical inverse scattering problems.

In particular, the form of the forward map introduced in our setting differs from the classical one, where the incident wave is typically a plane wave, meaning $u^{i,k}(x; v_s) = \exp(ikv_s \cdot x)$, where $v_s \in \mathcal{S}^{d-1}$, see [147].

So the forward map is given by the far field map, $\widetilde{\Lambda}_n^k$:

$$\widetilde{\Lambda}_n^k : u^{i,k}(x; v_s) \to u^{\infty,k}(\hat{x}; v_s)\,,$$

where $u^{\infty,k} : \mathcal{S}^{d-1} \to \mathbb{C}$ is defined as

$$u^{\infty,k}(\hat{x}; v_s) = \lim_{r\to\infty} r u^{s,k}(r\hat{x}; v_s) \exp(-ikr)|_{\hat{x}\in\mathcal{S}^{k-1}}, \quad \forall \hat{x} \in \mathcal{S}^{d-1}\,,$$

with $u^{s,k}$ being the solution of (6.5), where we leverage that $u^{i,k}(x; v_s)$ satisfies (6.4) with $S = 0$. Therefore, in this setting, the data set induced by the forward map is defined as:

$$\widetilde{\mathcal{D}}^k[n] = \left\{ \left(u^{i,k}(x; v_s), \widetilde{\Lambda}_n^k \left[u^{i,k} \right] (\hat{x}) \right) : v_s \in \mathcal{S}^{d-1}, \hat{x} \in \mathcal{S}^{d-1} \right\}.$$

The well-posedness and stability of the inverse scattering problem in this context has been studied in [120, Theorem 1.2].

The differences from the classical inverse scattering formulation is twofold: i) we use a richer set of probing functions, instead of using incident waves that are directionally localized (as plane waves) or whose sources are localized (as Green's functions), we use tight beams that combine these two properties, and ii) instead of measuring the scattered wave-field on a manifold around the domain of interest, we multiply it with a set of directional filters localized on the same manifold, and we compute its intensity. We should emphasize that this difference is significant. Take the plane-wave as the probing wave, as an example, it is only the direction of the incoming wave that can be tuned, and this composes 2 dimensions of degrees of freedom in 3D with $v_s \in \mathcal{S}^{d-1}$. The way our source term is designed automatically carries 4 dimensions of degrees of freedom with $(x_s, v_s) \in \Gamma_-$. Similarly, the way data gets taken also expands the degrees of freedom the measuring operator can access. It is a widely accepted fact that more data leads to more

stable reconstruction. This will be indeed demonstrated in the later sections.

Remark 6.2. *We note that even though the conventional inverse scattering problem has been shown to be ill-conditioned, a couple of strategies have been introduced in the literature to stabilize the problem. The most prominent strategy is to add the phase information (microlocally) [34, 198, 77]. At the first look, the Husimi data* (6.9) *also extracts the phase information, by integrating the scattered wave with an oscillatory test function* (6.8) *that is localized in position and direction. In very simple cases, we can even show that the two sets of information are equivalent. For example, suppose the wave field is of the simple form of* $u^k(x) = A(x)e^{\mathrm{i}kp\cdot v}$ *with* $p \in \mathbb{S}^{d-1}$ *and* $A(x) \geq 0$*, for all* $x \in \mathbb{R}^d$*. Then in the limit* $k \to \infty$*, we can fully recover* $u^k(x)$*, both the amplitude and the phase, on the boundary* ∂B_1 *using the Husimi data* (6.9)

$$\lim_{k\to\infty} H^k u^k(x,v) = |A(x)|^2 \delta(v-p)\,, \quad \forall (x,v) \in \partial B_1 \times \mathbb{S}^{d-1}\,.$$

However, in general cases, we are not aware of results that translate Husimi data to the phase data. Indeed, according to [116, 117, 7], this might be a very complicated phase retrieval problem that is beyond the scope of the dissertation.

Remark 6.3. *Another strategy to stabilize the inverse scattering problem is to transform the Helmholtz equation back to the time-domain, and solve the inverse acoustic wave problem, with either full or partial data available for all time* $T \geq 0$*. In various settings [30, 135, 199, 217], it is proved that the time-domain data is sufficient to reconstruct the medium. The wave equation and Helmholtz equation are Fourier transform of each other in time. Roughly speaking, the temporal data collected on the boundary translates to the boundary information for all frequency* k*. As such, the temporal data has wide-band information instead of being monochromatic, and thus is expected to be more stable. In our setting, though we require* $k \gg 1$*, we still use monochromatic information, and thus the data does not directly translate.*

We should note, however, that though the time-domain data is expected to be more

informative in theory, in practice, however, especially within the optimization-based reconstruction algorithm framework, the typical ℓ^2 misfit loss function results in an extremely non-linear problem that often leads to cycle-skipping, and convergence to spurious, non-physical, local minima. The numerical problem is usually attenuated by using the time/frequency duality and localizing the frequency content of the data, which is then processed in a hierarchical fashion [72, 185]. These are beyond the focus of the dissertation.

6.2.2 High-frequency limit and inverse Liouville scattering problem

The Liouville equation is a well-studied classical model for describing particle propagation. Any system with a large number of identical particles can be described by the Liouville equation, or its variants, which is often written as:

$$v \cdot \nabla_x f + \frac{1}{2} \nabla_x n \cdot \nabla_v f = S_{\mathrm{L}}(x, v)\,, \tag{6.14}$$

where $f(x, v)$ characterizes the number of particles on the phase space (x, v). Following the characteristics, we see that the particles follow Newton's second law:

$$\dot{x} = v\,, \quad \dot{v} = \frac{1}{2} \nabla_x n\,.$$

As usual in classical mechanics, we can define the Hamiltonian for each particle to be

$$H(x(t), v(t)) = \frac{1}{2} |v(t)|^2 - \frac{1}{2} n(x(t))\,,$$

which is preserved along the characteristics of the particles, i.e., $\frac{\mathrm{d}H}{\mathrm{d}t} = 0$.

We use (6.14) to describe photon propagation, and use the same setup as that in Section 6.2.1. The source term $S_{\mathrm{L}}(x, v)$ on the right-hand side of (6.14) describes how laser beams are shone into the medium, and takes the form of:

$$S_{\mathrm{L}}(x, v; x_s, v_s) = \phi(x - x_s)\psi(v - v_s)\,, \quad \text{with} \quad (x_s, v_s) \in \Gamma_-\,, \tag{6.15}$$

where both $\phi : \mathbb{R}^d \to \mathbb{R}$ and $\psi : \mathbb{R}^d \to \mathbb{R}$ are radially symmetric smooth functions that concentrate at the origin. By setting $(x_s, v_s) \in \Gamma_-$, we have the laser beam shining from the boundary ∂B_1 inward to the domain. The concentration of the beam is determined by ϕ and ψ in physical- and velocity-space respectively.

Similar to the previous section, we take the measurements of the light intensity at the boundary pointing outside of the domain. To do so, we set the test function $\zeta(x, v)$ and the measurements would be its convolution with the solution to (6.14):

$$Lf(x, v) = f * \zeta(x, v)\,. \tag{6.16}$$

The physical setup is clear. Imaging ζ a blob centers around $(x, v) = (0, 0)$, then $Lf(x_r, v_r)$ essentially represents a measuring equipment that takes in light intensity concentrated around (x_r, v_r) with the concentration determined by the size of the blob. The specific format of ζ will be specified in Section 6.3.

Forward Map: we define the forward map in a similar fashion as in Section 6.2.1. For any $(x_s, v_s) \in \Gamma_-$, we solve (6.14) with S_{L} defined in (6.15), and test the solution on $\zeta(x, v)$ evaluated on Γ_+:

$$\Lambda_n : \quad S_{\mathrm{L}}(x, v; x_s, v_s) \to Lf(x_r, v_r)|_{\Gamma_+}\,.$$

As a consequence, the dataset generated by this forward map is the collection of:

$$\mathcal{D}[n] = \{(S_{\mathrm{L}}(x, v; x_s, v_s), \Lambda_n[S_{\mathrm{L}}](x_r, v_r)) : (x_s, v_s) \in \Gamma_-, (x_r, v_r) \in \Gamma_+\}\,. \tag{6.17}$$

While the forward problem is to compute and construct this $\mathcal{D}[n]$ for any given n, the inverse problem amounts to inferring n using the information in $\mathcal{D}[n]$.

6.3 Relation between the two problems in the high-frequency regime

In this section we discuss the connection between the forward maps for the wave- and particle-like descriptions introduced in the section above. We start introducing the Wigner transform, and we use it to present the equivalence of the two descriptions for the forward maps in the high-frequency regime. Then we introduce the Husimi transform to take the limit of the measuring operator, and this is used to show the equivalence of the two inverse problems. Finally, we briefly introduce the stability of the inverse Liouville problem.

6.3.1 High-frequency limit of the forward problem

We first present their connection in the forward setting. We discuss the derivation of the Liouville equation as the limiting equation for the Helmholtz. This process is typically called taking the "classical"-limit, to reflect the passage from quantum mechanics to classical mechanics by linking the Schrödinger equation to the Liouville equation in the small $\hbar$ regime.

Among the multiple techniques to derive the classical limit we utilize the Wigner transform [109, 193, 23, 67]. Compared to other techniques, such as WKB expansion [100] and Gaussian beam expansion [201, 169, 187] , Wigner transform presents the equation on the phase space, and avoids the emerging singularities during the evolution. Let u_1^k and u_2^k be two functions, then the corresponding Wigner transform is defined as

$$W^k[u_1^k, u_2^k](x,v) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} e^{\mathrm{i}v\cdot y} u_1^k \left(x - \frac{y}{2k}\right) \overline{u_2^k} \left(x + \frac{y}{2k}\right) \mathrm{d}y \,. \tag{6.18}$$

Here $\overline{u_2^k}$ is the complex conjugate of u_2^k. We furthermore abbreviate $W^k[u_1^k, u_2^k]$ to be $W^k[u^k]$.

The Wigner transform $W^k[u^k]$ is defined on the phase space, is always real-valued, and the moments in v of $W^k[u^k]$ carry interesting physical meanings. In particular, the first

moment recovers the energy density $\mathcal{E}^k$:

$$\mathcal{E}^k(x) = \int_{\mathbb{R}^d} W^k[u^k](x,v)\mathrm{d}v = \left|u^k(x)\right|^2 , \tag{6.19}$$

and its second moment expresses the energy flux $\mathcal{F}^k$:

$$\mathcal{F}^k(x) = \int_{\mathbb{R}^d} vW^k[u^k](x,v)\mathrm{d}v = \frac{1}{k}\mathrm{Im}\left(\overline{u^k(x)}\nabla_x u^k(x)\right) . \tag{6.20}$$

Most importantly, if u^k solves the Helmholtz equation (6.3), one can show that $W^k[u^k]$ solves an equation in the form of the radiative transfer equation, and in the $k \to \infty$ limit, this degenerates to the Liouville equation (6.14). In what follows we seek to make this statement more precise by defining the functional space and an appropriate metric.

Let $\lambda > 0$, we define X_λ a space that contains all scalar real valued functions defined on the phase-space $\mathbb{R}^3 \times \mathbb{R}^3$:

$$X_\lambda = \left\{ \phi(x,y) \;\middle|\; \int_{\mathbb{R}^3} \sup_{x\in\mathbb{R}^3} (1+|x|+|\xi|)^{1+\lambda} |\hat{\phi}(x,\xi)| \mathrm{d}\xi < \infty \right\} , \tag{6.21}$$

with associated norm given by

$$\|\phi\|_{X_\lambda} = \int_{\mathbb{R}^3} \sup_{x\in\mathbb{R}^3} (1+|x|+|\xi|)^{1+\lambda} |\hat{\phi}(x,\xi)| \mathrm{d}\xi \,,$$

where $\hat{\phi}(x,\xi) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d} \phi(x,y) e^{-i\xi\cdot y}\mathrm{d}y$ is the Fourier transform in velocity-space. Now we cite a result from [42, Theorem 3.11, 3.12].

Theorem 6.1. *Let $n(x)$ be a $C^2(\mathbb{R}^d;\mathbb{R}_+)$ function that satisfies certain conditions (see Remark 6.4). Let u^k be the solution to* (6.3) *with radiation conditions, where the source term S^k_{H} is defined in* (6.6)*. Then the Wigner transform of u^k, denoted by $f^k(x,v) = W^k[u^k](x,v)$ solves*

$$v\cdot\nabla_x f^k + \frac{1}{2}\mathcal{L}^k_n[f^k] = -\frac{1}{k}\mathrm{Im}\left(W^k[u^k,S^k]\right) , \quad (x,v)\in\mathbb{R}^{2d} , \tag{6.22}$$

with the operator $\mathcal{L}_n^k$ *defined as*

$$\mathcal{L}_n^k[f^k] := \frac{\mathrm{i}}{(2\pi)^d} \int_{\mathbb{R}^{2d}} \delta^k[n](x,y) f^k(x,p) e^{\mathrm{i}y(v-p)} \,\mathrm{d}y \,\mathrm{d}p\,. \tag{6.23}$$

Here $\delta^k[n](x,y) = k\left[n\left(x+\frac{y}{2k}\right) - n\left(x-\frac{y}{2k}\right)\right]$. *Furthermore, when* $k\to\infty$, f^k *converges in weak-*$\star$ *sense to* $f(x,v)$ *in* $(X_\lambda)^\star$, *the solution to the Liouville equation* (6.14) *with the radiation condition* $\lim_{|x|\to\infty} f(x,v) = 0$ *for all* $x\cdot v < 0$, *and the source* $S_{\mathrm{L}}(x,v)$ *is:*

$$S_{\mathrm{L}}(x,v) = (2\pi)^d \frac{\pi}{2} \delta(x-x_s) |\hat{S}_{v_s}(v)|^2 \delta\left(|v|^2 = n(x_s)\right)\,. \tag{6.24}$$

Here $\hat{S}_{v_s}$ *denotes the Fourier transform, and the delta function* $\delta\left(|v|^2 = n(x_s)\right) \in \mathcal{D}'(\mathbb{R}^d)$ *means*

$$\langle \delta\left(|v|^2 = n(x_s)\right), g\rangle = \int_{|v|^2 = n(x_s)} g(v) dS_v, \quad \forall g \in \mathcal{S}(\mathbb{R}^d)\,.$$

Suppose S_v takes the form of (6.6), we can explicitly calculate its Fourier transform:

$$|\hat{S}_{v_s}(v)|^2 = C(\sigma,d)^2 \frac{1}{(2\pi)^d \sigma^{2d}} e^{-\frac{|v-v_s|^2}{\sigma^2}}\,.$$

Remark 6.4. *The formal derivation of the limit is shown in Appendix B. To prove it rigorously, we refer to [42, Theorem 3.11, 3.12] and [57]. The conditions for a rigorous proof are rather complicated to obtain. However, we mention that if* n *is radially symmetric, i.e.,* $n(x) = n(|x|)$, *the statement of the theorem holds true rigorously.*

Theorem 6.1 suggests that the wave model and the particle model are asymptotically equivalent in the high-frequency regime. According to (6.24), the source term concentrates at (x_s, v_s), the source location and the source velocity, when $k\to\infty$. The concentration on x is already achieved by taking to limit as $k\to\infty$, but the concentration profile in v still needs to be tuned by σ. Smaller σ results in a more concentrated source in this

limiting regime. Let $\sigma \to 0$, we have the source term S_{L} turning into:

$$\begin{aligned}(2\pi)^d \frac{\pi}{2}\delta(x-x_s)|\hat{S}_{v_s}(v)|^2\delta(|v|=1) &= \delta(x-x_s)\left(\frac{1}{\sigma\sqrt{\pi}}\right)^{d-1} e^{-\frac{|v-v_s|^2}{\sigma^2}}\delta(|v|=1) \\ &\to \delta(x-x_s)\delta(v-v_s)\,,\end{aligned} \tag{6.25}$$

where we used $n(x_s)=1$, given that x_s is out of the domain interest B_1.

In this specific limit, we have the explicit solution to the Liouville equation (6.14):

$$f(x,v) = \delta_{(x(t;(x_s,v_s)),v(t;(x_s,v_s)))}\,, \quad k\to\infty\,, \tag{6.26}$$

where $(x(t;(x_s,v_s)),v(t;(x_s,v_s)))$ are the location and velocity of a particle at time t that starts off at (x_s,v_s), meaning $(x(0;(x_s,v_s)),v(0;(x_s,v_s))) = (x_s,v_s)$ and

$$\begin{cases} \dfrac{\mathrm{d}x(t;(x_s,v_s))}{\mathrm{d}t} = v(t;(x_s,v_s))\,, \\ \dfrac{\mathrm{d}v(t;(x_s,v_s))}{\mathrm{d}t} = \dfrac{1}{2}\nabla_x n(x(t;(x_s,v_s)))\,. \end{cases} \tag{6.27}$$

The formulation in (6.26) means in this limit, with $k\to\infty$ and $\sigma \ll 1$, the wave becomes a curved ray that follows the trajectory of the particle that is governed by Newton's laws. As a consequence, recall the definition of energy and energy flux in (6.19)-(6.20):

$$\lim_{\sigma\to 0}\lim_{k\to\infty}\mathcal{E}^k(x) = \mathbf{1}_{t>0}\delta_{x(t;(x_s,v_s))}, \quad \lim_{\sigma\to 0}\lim_{k\to\infty}\mathcal{F}^k(x) = \mathbf{1}_{t>0}\delta_{x(t;(x_s,v_s))}v(t;(x_s,v_s))\,,$$

suggesting that $\mathcal{E}^k$ and $\mathcal{F}^k$ respectively show approximately the location and velocity of the trajectory.

6.3.2 High-frequency limit of the inverse problem

In the prequel we linked the two forward problems. We now proceed to connect the two inverse problems, by evaluating the convergence of the measurements. To do so, we first introduce Lemma 6.1 from [190, Section 2.5] that connects the Husimi and Wigner transforms.

Lemma 6.1. *Assume $u \in L^2(\mathbb{R}^d;\mathbb{R})$, and let $H^k u$ be the Husimi transform defined in (6.9) with ϕ_v^k being the test function (defined in (6.8)). Denote $f^k = W^k[u]$, and $G^k = W^k[\phi_0^k]$, the Wigner transform of u^k and ϕ_0^k respectively. Here $\phi_0^k = \phi_{v=0}^k$. Then*

$$H^k u(x,v) = f^k * G^k(x,v)\,, \quad \forall (x,v) \in \mathbb{R}^{2d}\,. \tag{6.28}$$

Proof. This theorem is a directly result of the Moyal identity

$$(W^k[h_1], W^k[h_2])_{L^2(\mathbb{R}^{2d})} = \left(\frac{k}{2\pi}\right)^d |(h_1,h_2)_{L^2(\mathbb{R}^d)}|^2\,, \quad \forall h_1, h_2 \in L^2(\mathbb{R}^d;\mathbb{R})\,, \tag{6.29}$$

and the fact that

$$W^k[\phi_v^k(x-\cdot)](y,p) = W^k[\phi_0^k](x-y, v-p)\,. \tag{6.30}$$

Using (6.9), we have

$$\begin{aligned}
H^k u(x,v) &= \left(\frac{k}{2\pi}\right)^d \left| u * \phi_v^k \right|^2 \\
&= \left(\frac{k}{2\pi}\right)^d \left| \Big(u(\cdot), \phi_v^k(x-\cdot)\Big)_{L^2(\mathbb{R}^d)} \right|^2 \\
&= \Big(W^k[u], W^k[\phi_v^k(x-\cdot)]\Big)_{L^2(\mathbb{R}^{2d})} \\
&= \Big(W^k[u], W^k[\phi_0^k](x-\cdot, v-\cdot)\Big)_{L^2(\mathbb{R}^{2d})} \\
&= f^k * G^k\,,
\end{aligned}$$

where we use (6.29) in the third equality, (6.30) in the fourth equality, and the definitions of f^k and G^k in the last equality. □

This lemma connects the measurement of u^k with the measurement on the phase space. Testing u^k using the test function ϕ_0^k is translated to testing f^k using the test function G^k. This allows us to pass to the limit on the phase space. Combining with Theorem 6.1, we have the following proposition:

Proposition 6.1. *Let the assumption in Theorem 6.1 hold true. Denote $f^k = W^k[u^k]$,*

with u^k solving the Helmholtz equation (6.3) *with the source term S_{H} defined in* (6.6)*, and denote f the solution to the Liouville equation* (6.14) *with source term S_{L} defined in* (6.24)*. If χ takes the form of* (6.10)*, so that G^k takes the form of:*

$$G^k(x,v) = \left(\frac{k}{\pi}\right)^d \exp\left(-k\left(|x|^2 + |v|^2\right)\right), \tag{6.31}$$

as $k \to \infty$, we have:

$$f^k * G^k(x,v) \to f(x,v)$$

weak-$\star$ in $(X_\lambda)^\star$.

Proof. Given the form of G^k in (6.31), for any $\phi \in X_\lambda$, as $k \to \infty$:

$$G^k * \phi(x,v) \longrightarrow \phi(x,v) \quad \text{in} \quad X_\lambda\,.$$

Thus,

$$\begin{aligned}\lim_{k\to\infty} \int_{\mathbb{R}^3\times\mathbb{R}^3} \left(f^k * G^k(x,v)\right)\phi(x,v)\mathrm{d}x\,\mathrm{d}v &= \lim_{k\to\infty} \int_{\mathbb{R}^3\times\mathbb{R}^3} f^k(x,v)\left(G^k * \phi(x,v)\right)\mathrm{d}x\,\mathrm{d}v \\ &= \lim_{k\to\infty} \int_{\mathbb{R}^3\times\mathbb{R}^3} f^k(x,v)\phi(x,v)\mathrm{d}x\,\mathrm{d}v \\ &= \int_{\mathbb{R}^3\times\mathbb{R}^3} f(x,v)\phi(x,v)\mathrm{d}x\,\mathrm{d}v\,,\end{aligned}$$

where we use $\|f^k\|_{(X_\lambda)^*}$ being bounded in the second equality, and $f^k \to f$ in the weak-$\star$ sense in the last equality. □

Remark 6.5. *We note that the statement of the proposition indeed uses the explicit form of χ as defined in* (6.10)*, but the use only lies in the fact that $G^k * \phi(x,v) \longrightarrow \phi(x,v)$ in the high frequency limit. Other forms of χ works equally well as long as this G^k serves as a delta measure when $k \to \infty$.*

Theorem 6.2. *Let the assumptions in Theorem 6.1 and Lemma 6.1 hold true, then:*

$$\lim_{k\to\infty} H^k u^k(x,v) = \lim_{k\to\infty} f^k * G^k(x,v) \xrightarrow[weak-\star]{} f(x,v),$$

in $(X_\lambda)^*$. *Furthermore, if* $H^k u^k$ *and* f *are continuous, then each element in* $\mathcal{D}^k[n]$ *has a limit in* $\mathcal{D}[n]$. *More specifically:*

$$\left(S_{\mathrm{H}}^k(x; x_s, v_s)\,, \Lambda_n^k[S_{\mathrm{H}}^k](x_r, v_r)\right) \to \left(S_{\mathrm{L}}(x, v; x_s, v_s)\,, \Lambda_n[S_{\mathrm{L}}](x_r, v_r)\right) \tag{6.32}$$

where S_{L} *takes the form of* (6.24), *and* $\Lambda_n[S_{\mathrm{L}}](x_r, v_r) = f(x_r, v_r)$. *In particular, if* $\sigma \to 0$,

$$\Lambda_n[S_{\mathrm{L}}](x_r, v_r) = f * \delta_{(\vec{0},\vec{0})}|_{\Gamma_+} = f(x_r, v_r)|_{\Gamma_+} = \delta(x - x_{r_s})\delta(v - v_{r_s})\,, \tag{6.33}$$

with (x_{r_s}, v_{r_s}) *being the outgoing location and velocity when the photon particle leaves the domain, namely:*

$$x_{r_s} = x(T; (x_s, v_s)), \quad v_{r_s} = v(T; (x_s, v_s))\,, \tag{6.34}$$

where $T = \sup_{t\geq 0}\{t | x(t; (x_s, v_s)) \in B_1\}$ *and* $\{x(t; (x_s, v_s), v(t; (x_s, v_s))\}$ *solves* (6.27).

This theorem naturally links the two inverse problems. In the $k \to \infty$ limit, the two datasets (6.12),(6.17) are asymptotically close with $\zeta = \delta_{(\vec{0},\vec{0})}(x, v)$ in (6.16). In the limit of $k \to \infty$ and $\sigma \to 0$, the dataset (6.12) is asymptotically approximately equivalent to

$$\mathcal{D}^\infty[n] = \{((x_s, v_s), (x_r, v_r)) : (x_s, v_s) \in \Gamma_-,\ (x_r, v_r) \text{ from (6.34)}\}\,. \tag{6.35}$$

6.3.3 Stability of Liouville inverse problem

In this section, we consider the stability of Liouville inverse problem. In particular, we focus on the stability of (6.35). We will show that when n is close enough to 1, D_n^∞ almost contains the information of the X-ray transforms of $n(x)$ and $\nabla_x n(x)$, while the inverse of X-ray transform is a well-posed inverse problem.

We first introduce the X-ray transform. Define

$$TS^{d-1} = \left\{(x, v) \,\middle|\, x \in \mathbb{R}^d,\, v \in \mathcal{S}^{d-1},\, \langle v, x\rangle = 0\right\}\,.$$

Assuming that $n(x)$ is continuous, we introduce the X-ray transform P, which maps

$n(x), \nabla_x n(x)$ into functions $Pn \in C(TS^{d-1}, \mathbb{R})$ and $P(\nabla_x n) \in C(TS^{d-1}, \mathbb{R}^d)$, such that

$$Pn(v,x) = \int_{-\infty}^{\infty} n(tv+x)\mathrm{d}t, \quad P(\nabla_x n)(v,x) = \int_{-\infty}^{\infty} \nabla_x n(tv+x)\mathrm{d}t.$$

To connect D_n^∞ with X-ray transform, we define a projection map $\mathcal{P} : \partial B_1 \times \mathcal{S}^{d-1} \to \mathbb{R}^d \times \mathcal{S}^{d-1}$

$$\mathcal{P}((x,v)) = (x - \langle x, v\rangle v, v)$$

that projects x to the plane with normal vector v. We also define in-out map $\mathcal{L} : \Gamma_- \to \Gamma_+$ corresponding to (6.34):

$$\mathcal{L}((x_s, v_s)) = (x_r, v_r)\,.$$

Remark 6.6. *We remark that the in-out map may not be well-defined for arbitrarily given n. Suppose $n(x) \geq c_0$ for all $x \in \mathbb{R}^d$ and some $c_0 > 0$, then according to the conservation of Hamiltonian*

$$H(x,v) = \frac{1}{2}|v|^2 - \frac{1}{2}n(x) = \frac{1}{2} - \frac{1}{2} = 0\,, \tag{6.36}$$

the velocity of the particle satisfies

$$|v(t)| = \sqrt{n(x(t))} \geq \sqrt{c_0} > 0\,,$$

for all time $t \geq 0$. This by no means suggests the non-trapping property, but it at least ensures that the potential is not a sink. In the general case, we do assume that n is non-trapping, so that any incoming particle can eventually be expelled out of the domain again, making the map $\mathcal{L}$ well-defined. Such non-trapping condition is closely related to geodesic X-ray transforms, and we list references [198, 77, 174] for interested readers. In our numerical examples, we choose the media to be locally repulsive in the sense that

$$n(x) + x \cdot \nabla n(x) \geq c_1 > 0\,, \quad \forall x \in \mathbb{R}^d\,. \tag{6.37}$$

Let $(x(t), v(t))$ be any particle trajectory that solves (6.27)*. Given* (6.37)*, we obtain the*

inequality

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\frac{1}{2}|x(t)|^2\right) = |v(t)|^2 + x(t)\cdot\frac{\mathrm{d}v}{\mathrm{d}t} = n(x(t)) + x(t)\cdot\nabla n(x(t)) \geq c_1 > 0\,. \tag{6.38}$$

In the last equality, we have used (6.36)*. By making use of* (6.38)*, the particle is non-trapped since* $|x(t)| \geq t\sqrt{\frac{1}{2}c_0}$ *for sufficiently large* $s > 0$*.*

We note that $\mathcal{P}((x,v)) \in TS^{d-1}$ for any $(x,v) \in \partial B_1 \times \mathcal{S}^{d-1}$, and $\mathcal{P}|_{\Gamma_-} : \Gamma_- \to \mathbb{R}^d \times \mathcal{S}^{d-1}, \mathcal{P}|_{\Gamma_+} : \Gamma_+ \to \mathbb{R}^d \times \mathcal{S}^{d-1}$ are invertible. Now, we are ready to introduce the following approximation theorem [178, Theorem 4.1]:

Theorem 6.3. *Assume*

$$\|\nabla n(x)\|_{L^\infty} \leq \Delta, \quad \|\|Hn(x)\|_F\|_{L^\infty} \leq \Delta$$

for some $\Delta > 0$*, then for any* $(v,x) \in TS^{d-1}$*, we have*

$$\left|(Pn(v,x), P(\nabla_x n)(v,x)) - \mathcal{P}|_{\Gamma_+} \circ \mathcal{L} \circ \left(\mathcal{P}|_{\Gamma_-}\right)^{-1}(v,x)\right| \leq C\Delta^2\,,$$

where $C > 0$ *is a constant only depends on* d*.*

According to Theorem 6.3, if n is almost a constant (close enough to 1), then we can use the data set to recover X-ray transform of $n, \nabla n(x)$. Thus, we can separate (6.35) into two inverse problems

$$\mathcal{D}^\infty[n] \Longrightarrow (Pn(v,x), P(\nabla_x n)(v,x)) \Longrightarrow n(x)\,,$$

where the first one can be approximately calculated if n is almost constant 1 and the second one is the inverse of X-ray transform that is well-posed according to [177, Theorem 5.1].

6.4 Numerical experiments

In this section we provide numerical evidence showcasing the theory developed above. In particular, we would like to demonstrate that as k increases, the measurement taken on the solution to the Helmholtz equation through the Husimi transform converges to the pointwise evaluation of the solution to the Liouville equation, and that the data becomes more and more sensitive to the perturbation in media, making the inverse problem more and more stable.

We first summarize the numerical setup and unify the notations, and then present a class of numerical results.

6.4.1 Numerical setup

We set up our experiment in a two-dimensional domain that takes the form of:

$$\Delta u^k + k^2 n(x) u^k = -k^{\frac{5}{2}} S_{v_{\mathrm{s}}}(k(x - x_{\mathrm{s}})), \quad x \in \mathbb{R}^2 . \tag{6.39}$$

The Sommerfeld radiation condition is imposed at infinity as well. The source term is given by

$$S_{v_{\mathrm{s}}}(x) = \sqrt{2} \left(\frac{\sigma}{\sqrt{\pi}} \right)^{\frac{3}{2}} \exp\left(-\sigma^2 \frac{|x|^2}{2} + \mathrm{i} v_{\mathrm{s}} \cdot x \right) , \tag{6.40}$$

for $(x_{\mathrm{s}}, v_{\mathrm{s}}) \in \Gamma_-$. We denote the solution to (6.39) by $u^k_{x_{\mathrm{s}}, v_{\mathrm{s}}}$ whenever the source center and the incident direction are relevant for the discussion. The Husimi transform defined in (6.9) takes the form

$$H^k u^k(x_{\mathrm{r}}, v_{\mathrm{r}}) = \left(\frac{k}{2\pi} \right)^d \left| u^k * \phi^k_{v_{\mathrm{r}}}(x_{\mathrm{r}}) \right|^2 , \tag{6.41}$$

with $(x_{\mathrm{r}}, v_{\mathrm{r}}) \in \Gamma_+$. We let the refractive index $n(x)$ set to be $n(x) = 1 + q(x)$ with the support of the heterogeneity $q(x) \subset B(r)$. The measurement is taken on $\partial B(R)$ with $R > r$. See Figure 6.2 for an illustration of the configuration.

Computationally we set the domain $D = [-L/2, L/2]^2$, with L significantly bigger

than R, and choose the spatial mesh size $h = 1/N$ with N being an even integer. For simplicity of representation, we use the angles $\theta_{\rm s}$ and $\theta_{\rm r}$ to denote the center of the sources and the center of the receivers, respectively, and the angles $\theta_{\rm i}$ and $\theta_{\rm o}$ are used to denote the incident and outgoing direction of the sources and receivers, respectively, so that:

$$\begin{aligned} x_{\rm s} &= (R\cos\theta_{\rm s}, R\sin\theta_{\rm s})\,, \\ v_{\rm s} &= (-\cos(\theta_{\rm s}+\theta_{\rm i}), -\sin(\theta_{\rm s}+\theta_{\rm i}))\,, \end{aligned} \tag{6.42}$$

and

$$\begin{aligned} x_{\rm r} &= (R\cos(\theta_{\rm s}+\theta_{\rm r}), R\sin(\theta_{\rm s}+\theta_{\rm r})) \\ v_{\rm r} &= (\cos(\theta_{\rm s}+\theta_{\rm r}+\theta_{\rm o}), \sin(\theta_{\rm s}+\theta_{\rm r}+\theta_{\rm o}))\,. \end{aligned} \tag{6.43}$$

The angles $\theta_{\rm i}$ and $\theta_{\rm o}$ take values in $[0, 2\pi)$, whereas the angles $\theta_{\rm i}$ and $\theta_{\rm o}$ take values in $(-\frac{\pi}{2}, \frac{\pi}{2})$. An illustration of the angles can be found in Figure 6.2. Since the mapping between $(\theta_{\rm s}, \theta_{\rm i}, \theta_{\rm s}, \theta_{\rm o})$ and the corresponding $(x_{\rm s}, v_{\rm s}, x_{\rm r}, v_{\rm r})$ is one-to-one, we present the quantities u^k and $H^k u^k$ on the θ coordinate system whenever there is no confusion.

The angles are discretized with step size $\Delta\theta$ and the angular grids are denoted by $\theta_{\rm s}^j\,, \theta_{\rm r}^j = j\Delta\theta$ for all $j = 0, \ldots, 2\pi/\Delta\theta - 1$, and $\theta_{\rm i}^j\,, \theta_{\rm o}^j = -\frac{\pi}{2} + j\Delta\theta$ for all $j = 1, \ldots, \pi/\Delta\theta - 1$.

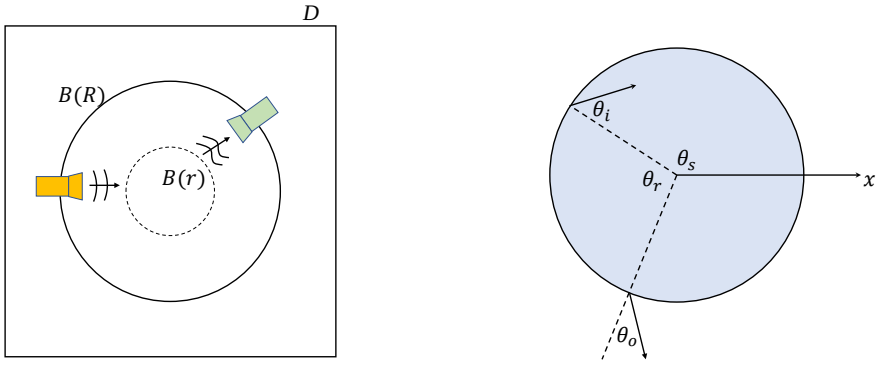


Figure 6.2: (left) illustration of the setup for numerical experiments, (right) sketch of the definition of the angles on the circle $\partial B(R)$ used to parameterize the data.

To compare the Husimi transform of the solutions, we further define two quantities.

The first quantity is the Husimi transform integrated in the outgoing direction

$$M_{\mathrm{o}}^{k}(x_{\mathrm{s}}, v_{\mathrm{s}}, x_{\mathrm{r}}) := \int_{\mathbb{S}^{+}_{x_{\mathrm{r}}}} H^{k} u^{k}_{x_{\mathrm{s}}, v_{\mathrm{s}}}(x_{\mathrm{r}}, v_{\mathrm{r}}) \mathrm{d}v_{\mathrm{r}} = \int_{-\pi/2}^{\pi/2} H^{k} u^{k}_{\theta_{\mathrm{s}}, \theta_{\mathrm{i}}}(\theta_{\mathrm{r}}, \theta_{\mathrm{o}}) \mathrm{d}\theta_{\mathrm{o}}\,, \tag{6.44}$$

where $\mathbb{S}^{\pm}_{x_{\mathrm{r}}} = \{v \in \mathbb{S}^{1} : \pm\nu(x_{\mathrm{r}}) \cdot v > 0\}$ and $\nu(x)$ is the unit outer normal vector at $x \in \partial\Omega$. Similarly, we also define the Husimi transform integrated along the outgoing boundary

$$M_{\mathrm{r}}^{k}(x_{\mathrm{s}}, v_{\mathrm{i}}, v_{\mathrm{r}}) := \int_{\partial\Omega^{+}_{v_{\mathrm{r}}}} H^{k} u^{k}_{x_{\mathrm{s}}, v_{\mathrm{i}}}(x_{\mathrm{r}}, v_{\mathrm{r}}) \mathrm{d}x_{\mathrm{r}} = \int_{(-\pi/2+\theta_{\mathrm{or}}, \pi/2+\theta_{\mathrm{or}})} H^{k} u^{k}_{\theta_{\mathrm{s}}, \theta_{\mathrm{i}}}(\theta_{\mathrm{r}}, \theta_{\mathrm{or}} - \theta_{\mathrm{r}}) \mathrm{d}\theta_{\mathrm{r}}\,, \tag{6.45}$$

where we denote $\theta_{\mathrm{or}} = \theta_{\mathrm{o}} + \theta_{\mathrm{r}} \in [0, 2\pi)$, and define $\partial\Omega^{\pm}_{v_{\mathrm{r}}} = \{x \in \partial\Omega : \pm\nu(x) \cdot v_{\mathrm{r}} > 0\}$.

To solve the Helmholtz equation (6.39), we use the truncated kernel method [204], and solve for the Lippmann-Schwinger equation to obtain the scattered field $u^{\mathrm{s},k}$. This allows us to push for high-frequency without suffering from the numerical pollution that finite difference or finite element methods often have. The scattered field is then combined with the incident field $u^{\mathrm{i},k}$ to yield u^{k}.

6.4.2 Numerical examples

In the first example, we set $L = 1$, $R = 0.3$ and $r = 0.25$. For the medium, we set the heterogeneity to be the radially symmetric smooth function

$$q(x) = \begin{cases} A \exp\left(-\frac{1}{1-|x|^2/r^2}\right)\,, & |x| < r\,, \\ 0\,, \quad \text{otherwise}\,. \end{cases} \tag{6.46}$$

Clearly, the support of $q(x)$ is contained in $B(r)$; see Figure 6.3. We note that with $-1 < A \leq 0$, the media is locally repulsive, and the incident wave is guaranteed to be expelled out of the domain. For the source term, we fix $\sigma = 2^{-5}$ in the following experiments. Noting that the medium $n(x)$ is radially symmetric, one can study the scattered data for a fixed source location. We choose $\theta_{\mathrm{s}} = \pi/4$; see Figure 6.3. For discretization, we choose spatial step size $h = 1/(2k)$ in the truncated kernel solver, and

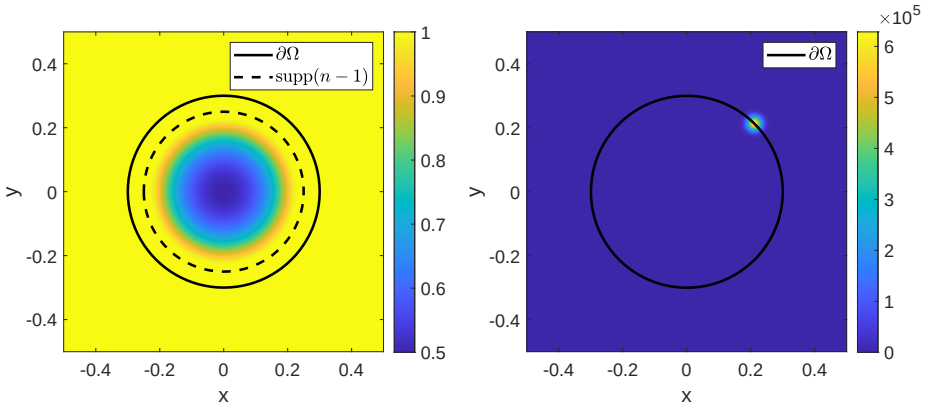


Figure 6.3: The left plot illustrates the medium $n(x) = 1 + q(x)$ in (6.46) with $A = -0.5$. The right plot shows the amplitude of source $|S_{v_s}(k(x - x_s))|$ with $k = 2^{11}$, $\sigma = 2^{-5}$ and $\theta_{\mathrm{s}} = \pi/4$.

$\Delta\theta = \pi/30$ for the angular grids.

We first show the solution's behavior as k increases in Figure 6.4. As k increases, the solution converges to a narrow beam that follows the characteristic equation (6.27).

We compute the Husimi transform $H^k u^k$ for different k and we compare them with the trajectories of the Liouville equation. The results are shown in Figure 6.5, where we can observe that for a fixed θ_{i}, $H^k u^k$ converges to a delta function on the θ_{r}-θ_{o} plane, as k increases. This agrees with the statement in Theorem 6.1, especially equation (6.33).

We then compare the integrated Husimi transform defined in (6.44) and (6.45). In Figure 6.6 and Figure 6.7, we demonstrate the convergence of M_{o}^k and M_{r}^k as k increases.

As k increases, the outgoing data becomes more and more sparse, and fewer and fewer detectors can receive outgoing light, leading to the sparser matrix presentation of Λ_n^k (see definition in (6.11). This is shown in Figure 6.8 for different k.

Finally we compare the change of Λ_n^k as n differs, for different k. Let $n_0(x) = 1$ as the background media whose corresponding map is denoted Λ_0^k, and by adjusting A we design a sequence of $n(x)$. We measure how the Frobenius norm $\|\Lambda_n^k - \Lambda_0^k\|_{\mathrm{F}}$ changes with respect to $\|n - n_0\|_{L^\infty}$ for different k. As can be seen in Figure 6.9, as k increases, the slope of $\|\Lambda^k - \Lambda_0^k\|_{\mathrm{F}}$ as $\|n - n_0\|_{L^\infty} \to 0$ increases. This confirms that bigger k sees more sensitivity of the data when n changes, hence the reconstruction is expected to be better

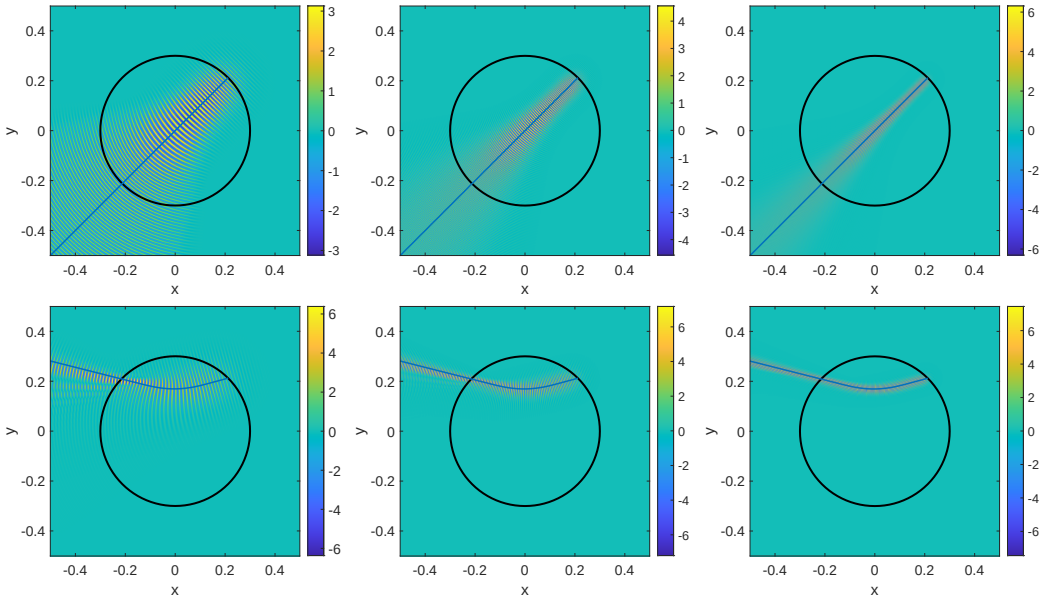


Figure 6.4: The real part of u^k for $k = 2^9$ (left), $k = 2^{10}$ (middle) and $k = 2^{11}$ (right). The blue lines show the Liouville trajectory that solves (6.27). The medium (6.46) has amplitude $A = -0.5$. The incident direction $\theta_{\mathrm{i}} = 0$ (upper) and $\theta_{\mathrm{i}} = -\pi/6$ (lower).

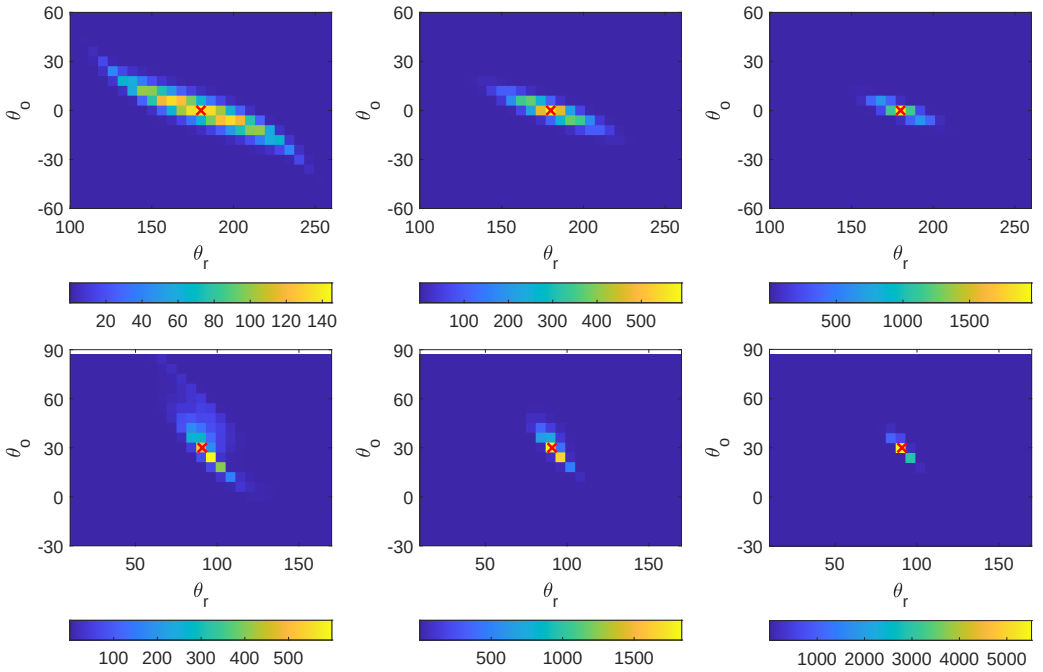


Figure 6.5: The Husimi transform $H^k u^k$ for $k = 2^9$ (left), $k = 2^{10}$ (middle) and $k = 2^{11}$ (right). The upper row shows the results with $\theta_{\mathrm{i}} = 0$, and the lower row shows the results with $\theta_{\mathrm{i}} = -\pi/6$. The red crosses show the outgoing position and direction (6.34) of the Liouville trajectory. The medium (6.46) has amplitude $A = -0.5$.

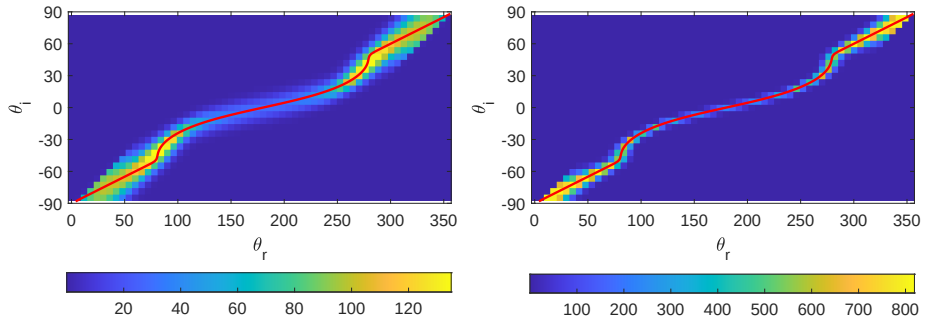


Figure 6.6: The averaged Husimi transform M_o^k for $k = 2^9$ (left) and $k = 2^{11}$ (right). The red lines show the outgoing position (6.34) of the Liouville trajectory. The medium (6.46) has amplitude $A = -0.5$.

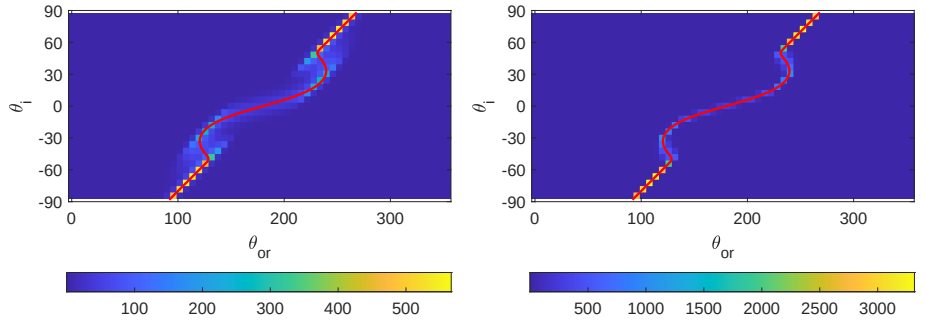


Figure 6.7: The averaged Husimi transform M_r^k for $k = 2^9$ (left) and $k = 2^{11}$ (right). The red lines show the outgoing direction (6.34) of the Liouville trajectory. The medium (6.46) has amplitude $A = -0.5$.

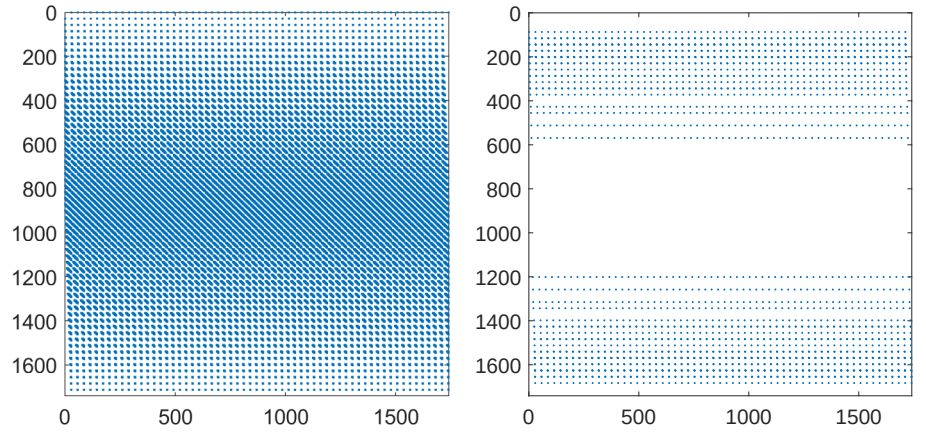


Figure 6.8: Sparsity of the matrix Λ_n^k for $k = 2^4$ (left) and $k = 2^{11}$ (right). Rows represent different (θ_r, θ_o), and columns represent different (θ_s, θ_i). Elements that are larger than half of the maximal element in Λ_n^k are shown. For $k = 2^4$, we use larger computational domain $[-8, 8]^2$, and the step size is $h = 2^{-8}$.

for higher k.

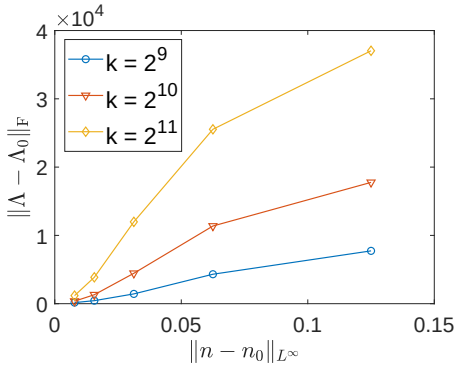


Figure 6.9: The dependence of $\|\Lambda^k - \Lambda_0^k\|_{\mathrm{F}}$ on the medium perturbation $\|n - n_0\|_{L^\infty}$. Different $\|n - n_0\|_{L^\infty}$ is obtained by tuning the amplitude A in the medium (6.46).

6.5 Inversion Algorithm

The inverse problem that we study in this chapter has a different setup from the conventional one. While the conventional setup has either the concentration in the incoming direction, or in the incoming source location, our experimental setup requires concentration in both direction and source location. Naturally we expect a better stability in the reconstruction process, compared to the traditional formulation. In this section we showcase such stability.

Numerically the reconstruction process is formulated as a PDE-constrained minimization problem, where we seek to minimize the misfit between the data and the forward model:

$$\min_n \left\| D - \mathcal{D}^k[n] \right\|^2_{L^2(\Gamma_- \times \Gamma_+)} , \tag{6.47}$$

or equivalently, in the discretized form:

$$\min_n \mathcal{J}[n], \qquad \text{where} \qquad \mathcal{J}[n] := \frac{1}{2n_{\mathtt{rcv}} n_{\mathtt{src}}} \sum_{i=1}^{n_{\mathtt{rcv}}} \sum_{j=1}^{n_{\mathtt{src}}} \left| D^{i,j} - \left(\mathcal{D}^k[n] \right)^{i,j} \right|^2 . \tag{6.48}$$

In particular, $n_{\mathtt{rcv}}$ and $n_{\mathtt{src}}$ stand for the number of receivers and sources, and each point

$(\mathcal{D}^k[n])^{i,j}$ is the intensity squared of the impulse response generated by illuminating the medium n with a tight beam given by (6.6) originated at x_{s}^i with direction v_{s}^i, which is then filtered using (6.8) centered at x_{r}^j with direction v_{r}^j. See definition in (6.41), with $(x_{\mathrm{r}}, v_{\mathrm{r}})$ replaced by $(x_{\mathrm{r}}^j, v_{\mathrm{r}}^j)$, and u^k solving (6.39) with $(x_{\mathrm{s}}, v_{\mathrm{s}})$ replaced by $(x_{\mathrm{s}}^i, v_{\mathrm{s}}^i)$.

We employ quasi-newton methods for finding a local minimum[4], thus we need to efficiently compute the gradient of the misfit function. In order to provide a fully self-contained exposition we briefly summarize below how to compute the gradient for only one data point using the adjoint-state methods. From there the computation for the full gradient can be easily deduced.

We can readily compute the application of the gradient to a perturbation δn by using the chain rule, which results in

$$\nabla \mathcal{J}[n]\delta n = \left(\frac{k}{2\pi}\right)^d \Big(D - H^k u^k(x_{\mathrm{r}}, v_{\mathrm{r}})\Big) \operatorname{Real}\Big(2\overline{(u^k * \phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}}))}(\phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}}) * F[n]\delta n)\Big)\,, \tag{6.49}$$

where $F[n]$ is linearized forward wave-propagation operator, given by the Born approximation of the scattered wave-field [49]. Thus the gradient can be easily computed by applying the adjoint of the Born approximation to the residual times the filter function, i.e.,

$$\nabla \mathcal{J}[n] = 2\left(\frac{k}{2\pi}\right)^d \operatorname{Real}\Big(F[n]^*\Big(\big(D - H^k u^k(x_{\mathrm{r}}, v_{\mathrm{r}})\big)(u^k * \phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}}))\overline{(\phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}} - x))}\Big)\Big)\,.$$

Fortunately, the application of the adjoint of the Born approximation operator is well studied: it can be performed by solving the adjoint equation followed by a multiplication by the solution of the forward wave problem[5]. In this case the adjoint equation is the same Helmholtz equation, but with adjoint Sommerfeld radiation conditions, i.e., we solve

$$\begin{aligned} \Delta v + k^2 n(x) v &= \Big(D - H^k u^k(x_{\mathrm{r}}, v_{\mathrm{r}})\Big)\,(u^k * \phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}}))\overline{(\phi_{v_{\mathrm{r}}}^k(x_{\mathrm{r}} - x))} \quad x \in \mathbb{R}^d\,, \\ \frac{\partial v}{\partial r} + \mathrm{i}kv &= \mathcal{O}(r^{-(d+1)/2}) \text{ as } r = |x| \to \infty\,. \end{aligned} \tag{6.50}$$

[4]Given that the problem is very non-linear, there is no guarantee that we can find the global minimum.
[5]We redirect the interested readers to [48] for a modern self-contained presentation.

Thus, using (6.50), we can easily compute the application of the adjoint of the Born approximation

$$F[n]^* \left(\big(D - H^k u^k(x_{\mathrm{r}}, v_{\mathrm{r}})\big)(u^k * \phi^k_{v_{\mathrm{r}}}(x_{\mathrm{r}}))\overline{(\phi^k_{v_{\mathrm{r}}}(x_{\mathrm{r}} - x))} \right) = \overline{u^k} v. \tag{6.51}$$

where v solves (6.50).

We point out that in (6.51), the source for the adjoint is conjugated, thus following (6.8), we can see that it means that the $\overline{(\phi^k_{v_{\mathrm{r}}}(x - x_{\mathrm{r}}))}$ is pointing towards the interior of the domain in direction $-v_{\mathrm{r}}$.

We solve (6.48) using L-BFGS [53, 220], a quasi-Newton method in Matlab. We consider the initial perturbation equal to zero. We set a first order optimality tolerance of 10^{-5} and let the algorithm run for a maximum of 300 iterations or until the tolerance is achieved.

To avoid the inverse crime [83], the data is generated by solving the Lippmann-Schwinger equation discretized by the truncated kernel method [204] as in Section 6.4, and the inversion is performed with an 4th-order finite difference scheme for both (6.39) and (6.50). To generate the data, we set the computational domain to be $K = [-1, 1]^2$ with $N_{\mathrm{LS}} = 256^2 = 65536$ grid points so that there are at least 12 points per wavelength for the largest $k = 2^6$. In the inversion, we discretize the same domain K with $N_{\mathrm{FD}} = 163^2 = 26569$ grid points so that there are at least 8 points per wavelength for $k = 2^6$. We enclose the domain K with perfect matching layer (PML) to avoid reflection. We choose the thickness of PML to be 2.5 times wavelength.

The measurement is taken on $\partial B(R)$ with $R = 0.4$ in all the examples. To generate the probing ray, we set $\sigma = 2^{-2}$ in (6.40). We compute the data with the source position and incident direction

$$\begin{aligned} x_{\mathrm{s}}^{i_1} &= (R\cos\theta_{\mathrm{s}}^{i_1}, R\sin\theta_{\mathrm{s}}^{i_1}) \\ v_{\mathrm{s}}^{i_1,i_2} &= (-\cos(\theta_{\mathrm{s}}^{i_1} + \theta_{\mathrm{i}}^{i_2}), -\sin(\theta_{\mathrm{s}}^{i_1} + \theta_{\mathrm{i}}^{i_2})) \end{aligned}$$

where $\theta_{\mathrm{s}}^{i_1} = \pi + i_1 \frac{\pi}{48}$ for all $i_1 = 0, \ldots, 95$ and $\theta_{\mathrm{i}}^{i_2} = -\frac{\pi}{2} + i_2 \frac{\pi}{49}$ for all $i_2 = 1, \ldots, 48$, and the receiver position

$$x_{\mathrm{r}}^{j_1} = (R\cos\theta_{\mathrm{r}}^{j_1}, R\sin\theta_{\mathrm{r}}^{j_1})$$
$$v_{\mathrm{r}}^{j_1,j_2} = (\cos(\theta_{\mathrm{r}}^{j_1} + \theta_{\mathrm{o}}^{j_2}), \sin(\theta_{\mathrm{r}}^{j_1} + \theta_{\mathrm{o}}^{j_2}))$$

where $\theta_{\mathrm{r}}^{j_1} = j_1 \frac{\pi}{48}$ for all $j_1 = 0, \ldots, 95$ and $\theta_{\mathrm{o}}^{j_2} = -\frac{\pi}{2} + j_2 \frac{\pi}{49}$ for all $j_2 = 1, \ldots, 48$.

In all the examples, the scattered data is perturbed with the noise in the form

$$\widetilde{D}^{i,j} = D^{i,j} + 0.05\varepsilon \frac{D^{i,j}}{|D^{i,j}|} \tag{6.52}$$

where ε is symmetric Bernoulli random variable that takes the values ± 1.

All the experiments are reported on a server with 64-core Intel Xeon CPU and 256 Gigabytes RAM. The code accompanying this chapter are publicly available [70].

In order to illustrate the reconstruction using Husimi data, we choose three examples of increasing complexity. The exact contrast function $q(x)$'s are shown in Figure 6.10.

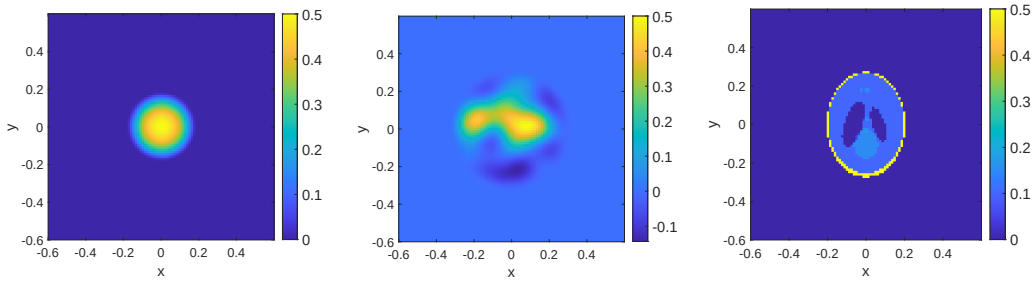


Figure 6.10: The contrast function $q(x)$ for our three examples: a bump function (left), a delocalized function (middle) and the Shepp-Logan phantom (right).

In the first example, we consider a single bump in the form (6.46) with $A = 0.5$ and $r = 0.2$, which is shown in Figure 6.10 (left). We run the minimization loop as described above using $k = 2^4$ and $k = 2^6$, and the resulting reconstruction are shown in Figure 6.11. From Figure 6.11 we can clearly see that as k becomes larger, the reconstruction becomes closer to the true medium. The solution time for $k = 2^6$ is 15787.1 seconds.

In the second example, we consider a delocalized medium. The delocalized contrast

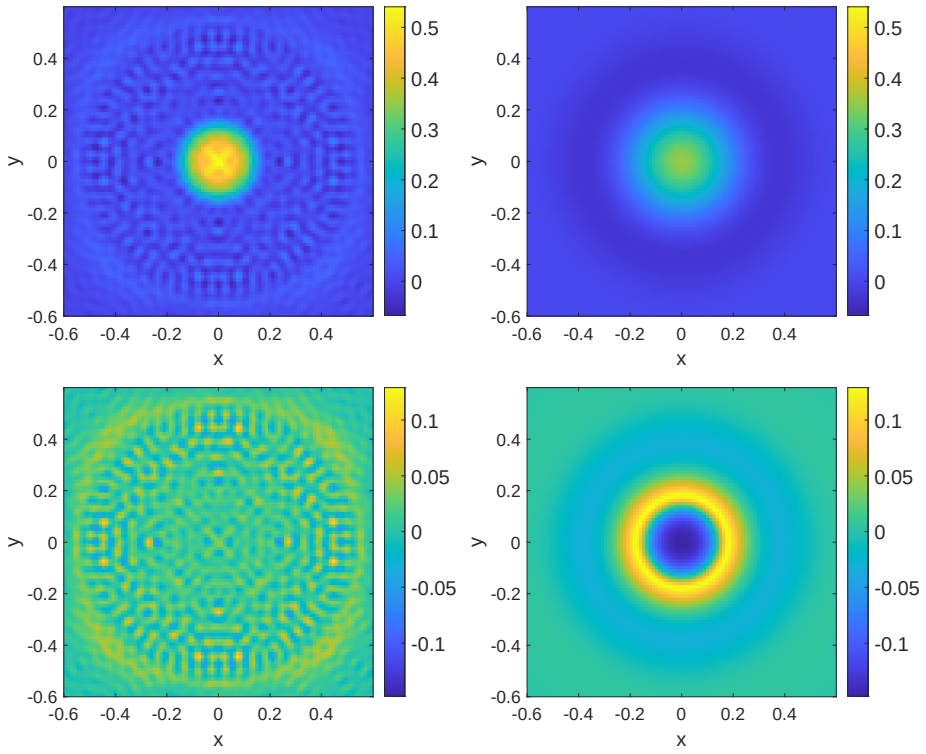


Figure 6.11: Recovering a single bump contrast function. The upper row shows the estimated contrast function and the lower row shows the reconstruction error at $k = 2^6$ (left) and $k = 2^4$ (right).

function $q(x)$ is obtained by convolving a pointwise independent Gaussian random field with a Gaussian mollifier. The main difference with the single bump example is that the refractive index, can be smaller than the background one, thus allowing for more complex ray paths as shown in Figure 6.10 (center). We repeat the same experiments, whose results are shown in Figure 6.12. The solution time required for $k = 2^6$ is 13185.3 seconds.

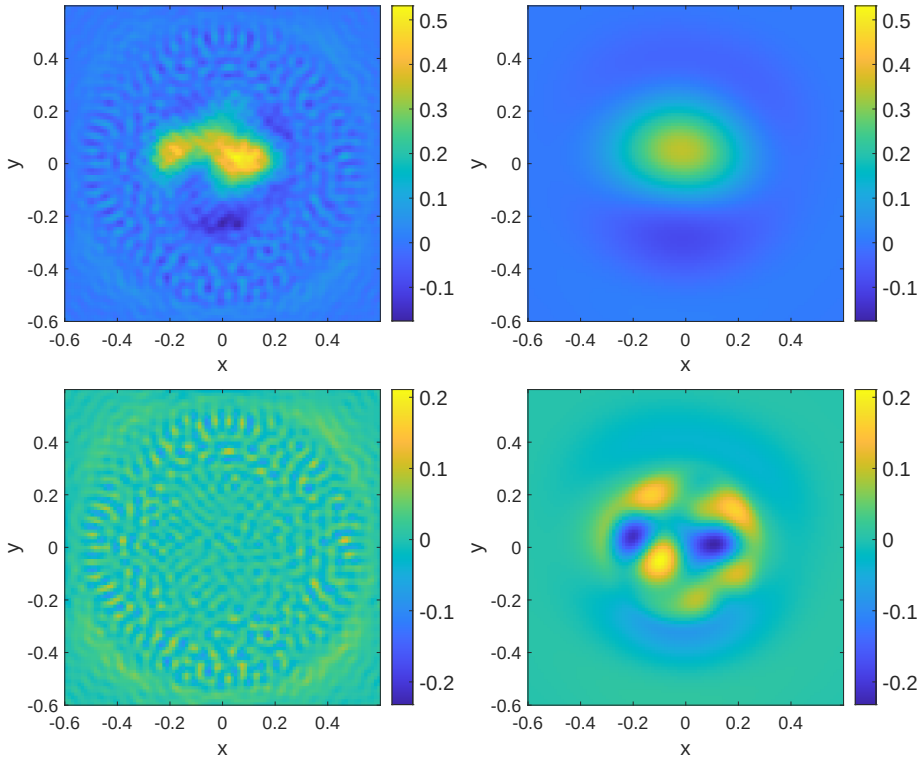


Figure 6.12: Recovering a delocalized contrast function. The upper row shows the estimated contrast function and the lower row shows the reconstruction error at $k = 2^6$ (left) and $k = 2^4$ (right).

Finally, for the third example, we consider the more challenging, and more practical, problem of recovering the Shepp-Logan phantom, depicted in Figure 6.10 (right). In this case we have very sharp transitions of the refractive index, which will generate a strong reflection, compared to the refraction-dominated media considered before. In addition, the interior of the still acts as a resonant cavity, thus creating a large amount of interior reflections, which are exacerbated as the frequency increases. We perform the same experiments as above, whose results are depicted in in Figure 6.13. The solution time required

for $k = 2^6$ is 14640.4 seconds. In this case, the reconstruction is qualitatively worse than before. We can still see the shape of the phantom, but with a large amount of artifacts. These artifacts are common to the three examples, but are somewhat more notorious for the Shepp-Logan phantom. Indeed, these artifacts can be in part explained by the large difference in the dispersion relation between the forward and backwards discretization. The Lippmann-Schwinger discretization used for the forward problem is known to be highly accurate if the media is smooth. In the cases before, the data generated by the Lippmann-Schwinger solver is close to the analytical solution, and the artifacts seems to come mostly for the phase errors in the finite-difference discretization. However, in this case the phantom is discontinuous thus creating large phase errors in the solution of the equation, and therefore the forward map, which in return produce more notorious artifacts.

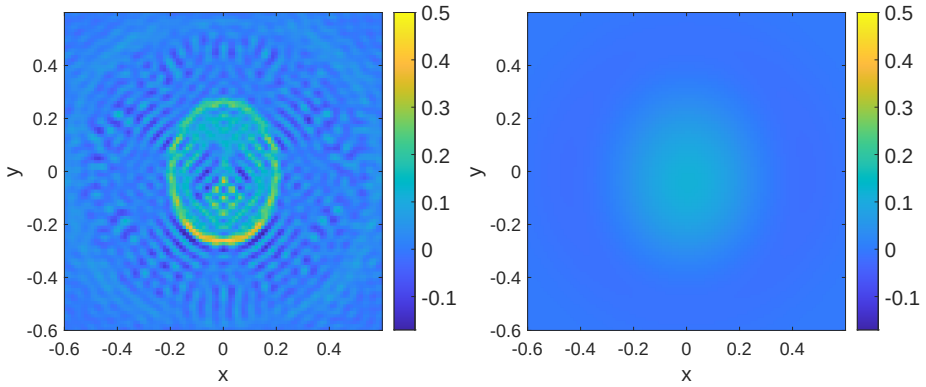


Figure 6.13: Recovering the Shepp-Logan phantom. The estimated contrast functions are shown for $k = 2^6$ (left) and $k = 2^4$ (right).

To avoid inverse crime, we have used two different solvers for computing the equation. The two solvers produce relatively large phase errors that propagate in the reconstruction. The reconstruction can be significantly improved if we use the same PDE solvers in generating data and reconstructing the media. In Figure 6.14, we show the reconstructions of the same single bump medium as in Figure 6.11 but with the 4th-order finite difference for both data generation and inversion. It can be seen that the artifacts in the estimated medium are much smaller for larger k and the reconstructed medium achieves a relative

L^2 error of 0.0389 for $k = 2^6$. Better reconstruction can also be seen in Figure 6.15 for the reconstructed delocalized medium, whose relative L^2 error is 0.0341 for $k = 2^6$. In Figure 6.16, we show the reconstruction for the Shepp-Logan phantom. We can observe that as the frequency increased the reconstruction becomes better, though due to computational limitations induced by the current implementation, we were unable to test with a higher frequency. However, we would expect to obtain even a better reconstruction.

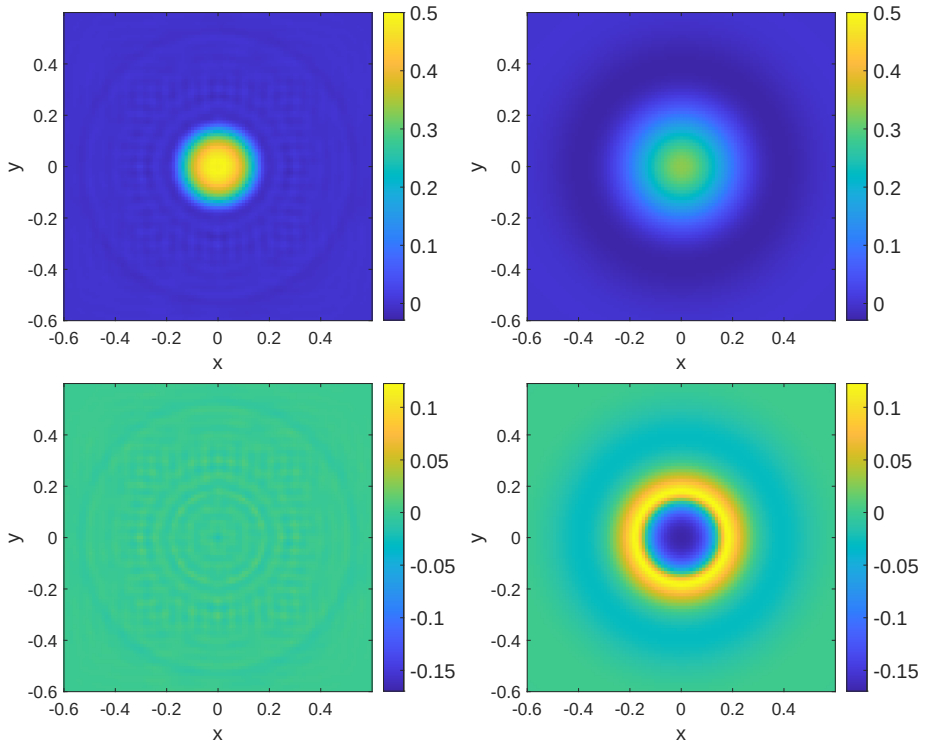


Figure 6.14: Recovering a single bump contrast function with 4th-order finite difference solver for both data and inversion. The upper row shows the estimated contrast function and the lower row shows the reconstruction error at $k = 2^6$ (left) and $k = 2^4$ (right).

Lastly, we compare the conventional inverse scattering problem and our new inverse problem using the Husimi data. We choose the incident wave $u^{\mathrm{i},k} = e^{\mathrm{i}\omega\hat{\theta}\cdot x}$ with $\hat{\theta} \in \mathbb{S}^1$ in (6.4), and measure the scattered far field data $u^{\mathrm{s},k}$. Again we cast the problem as a nonlinear least square problem, and solve it using L-BFGS. We consider the initial perturbation equal to zero, and set a first order optimality tolerance of 10^{-5}.

For simplicity, we use 4th-order finite difference for both data generation and inversion. The setup of the computational domain and the discretization are the same as in the

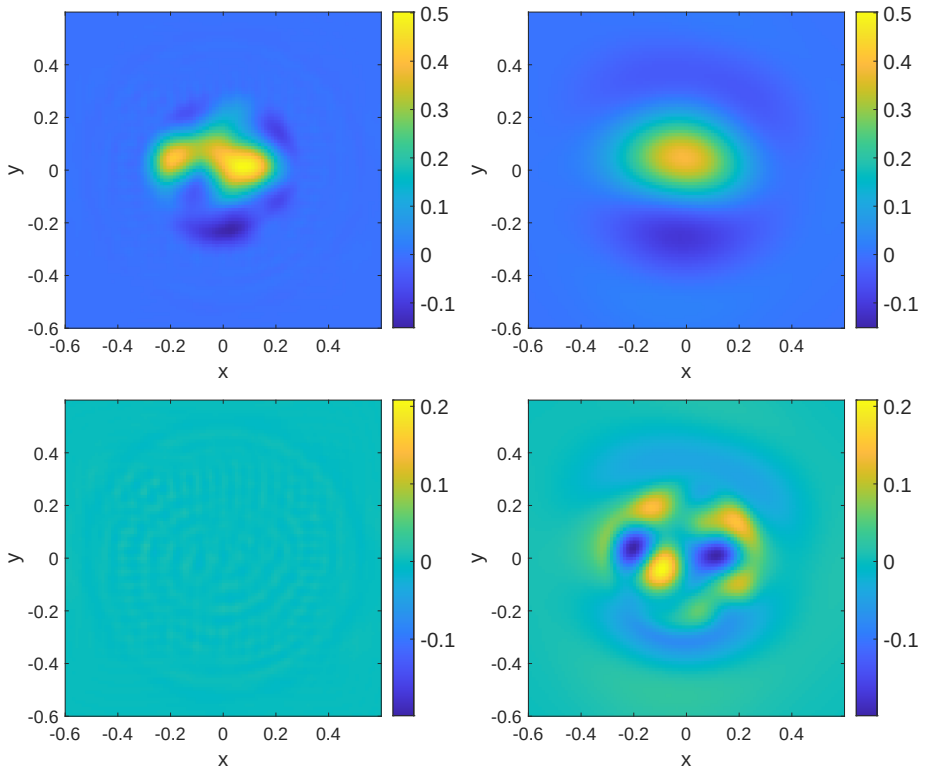


Figure 6.15: Recovering a delocalized contrast function with 4th-order finite difference solver for both data and inversion. The upper row shows the estimated contrast function and the lower row shows the reconstruction error at $k = 2^6$ (left) and $k = 2^4$ (right).

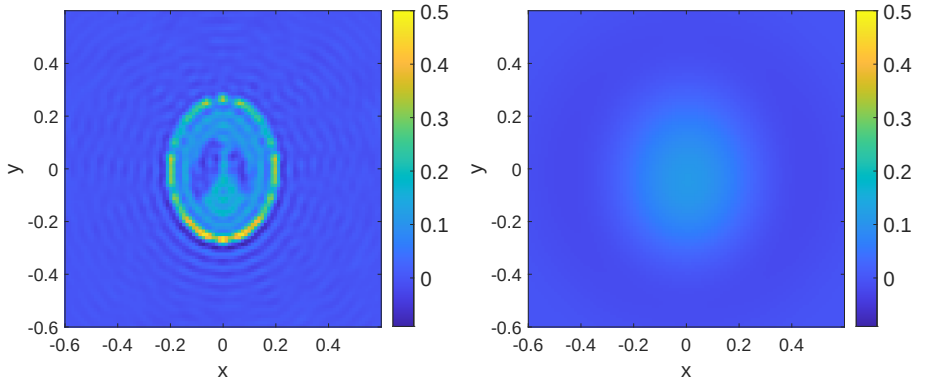


Figure 6.16: Recovering the Shepp-Logan phantom with 4th-order finite difference solver for both data and inversion. The estimated contrast functions are shown for $k = 2^6$ (left) and $k = 2^4$ (right).

previous examples.

The far field measurement is taken on the boundary $\partial B(\widetilde{R})$ with $\widetilde{R} = 1$. We compute the data with 180 incident directions $\hat{\theta}$ that are equally distributed on $\mathbb{S}^1$ and 180 receivers that are equally distributed on $\partial B(\widetilde{R})$. We add 5% noise to the scattered data in the form of (6.52).

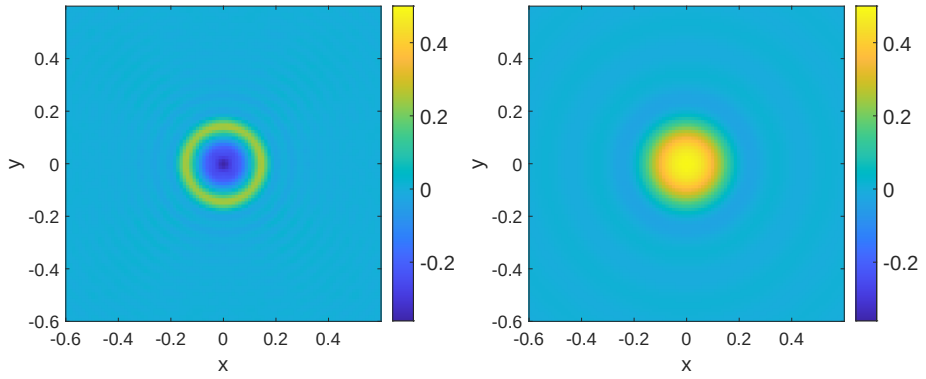


Figure 6.17: Recovering a single bump contrast function by plane waves. The estimated contrast function at $k = 2^6$ (left) and $k = 2^4$ (right) are shown. 4th-order finite difference solver is used for both data and inversion.

Finally, we test the robustness of the new formulation with respect to the non-convexity of the loss function. The ill-posedness of the inverse scattering problem is often manifested as a very non-convex loss function with a myriad of local minima. As a consequence, any PDE constrained optimization-based reconstruction has a higher chance of converging to a non-physical minimum, a process that is often called cycle-skipping [206]. For comparing the new formulation and the traditional one we also run the classical full-wave form inversion in frequency domain, using data at a single frequency, using the delocalized media in Figure 6.10. As discussed in Section 6.2.1, in the classical formulation one probes the medium with plane waves, and the measurement operator samples the wavefield directly on the boundary of the domain of interest. Numerically, we minimize the ℓ^2 misfit of the wavefield at the boundary, using the same L-BFGS solver as before. Initial guess is zero. We repeat the experiments for two different wave numbers that are used in the new formulation as well. The results are shown in Figures 6.17, 6.18, and 6.19, respectively. In the plots we can observe that at low-frequencies we recover a smoothed version of the

medium, but as the frequency increases we encounter cycle-skipping, i.e., the algorithm converges to a spurious medium. This is an stark contrast with the inversion results of the new formulation shown in Figures 6.14, 6.15, and 6.16, where at low-frequency the reconstruction does not perform as well, but it is more stable at high-frequencies, providing an accurate reconstruction.

In summary the numerical experiments seem to indicate that the new inverse formulation is far more robust to cycle skipping than its traditional counterpart.

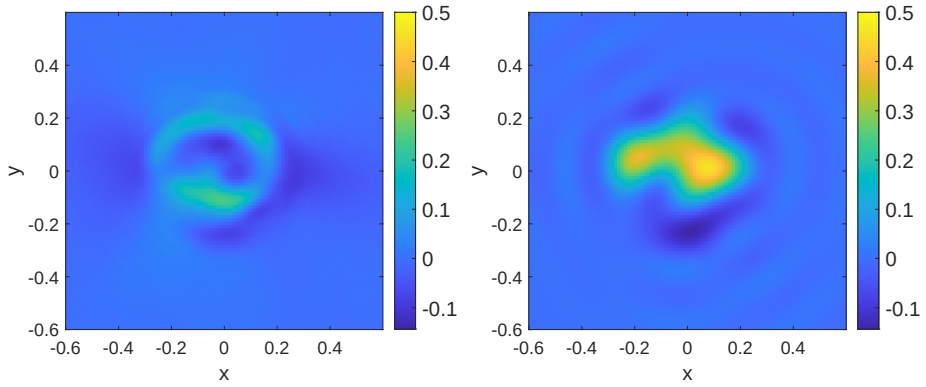


Figure 6.18: Recovering a delocalized contrast function by plane wave. The estimated contrast function at $k = 2^6$ (left) and $k = 2^4$ (right) are shown. 4th-order finite difference solver is used for both data and inversion.

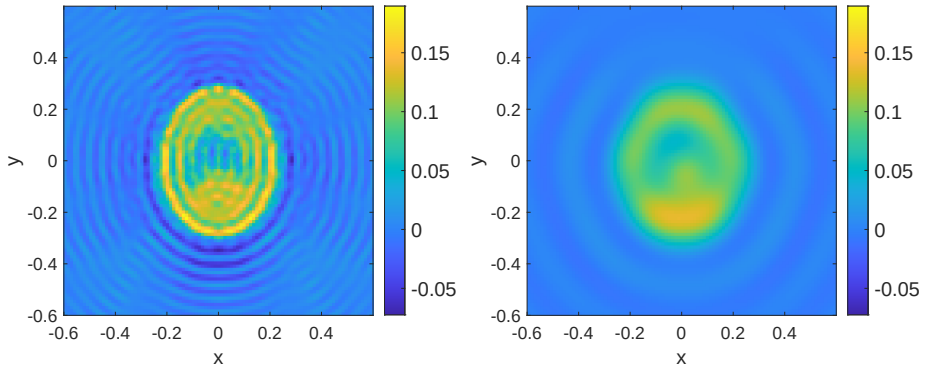


Figure 6.19: Recovering the Shepp-Logan phantom by plane wave. The estimated contrast function at $k = 2^6$ (left) and $k = 2^4$ (right) are shown. 4th-order finite difference solver is used for both data and inversion.

6.6 Conclusions

To reconstruct an unknown medium, the generalized Helmholtz inverse scattering problem uses data pairs consisting of the impinging and scattered wave fields, while Liouville inverse scattering problems uses data pairs consisting of incoming and outgoing wave location and direction. The former is regarded ill-posed in the high-frequency regime, while the latter is well-posed. This is intuitively contradicting to the fact that Liouville equation is the asymptotic limit of the Helmholtz equation.

We investigate this issue in this chapter. In particular, we develop a new formulation for studying the Helmholtz inverse scattering problem with a new data collection process, and we show that this new formulation, in the high-frequency limit, becomes the Liouville inverse scattering problem, and thus inherits the well-posedness nature. This discovery bares the conceptual merit of providing the mathematical description of the wave-particle duality for light propagation in the inverse setting. In addition, this discovery also suggests a more stable numerical reconstruction process for studying the Helmholtz inverse scattering problem, which we showcase using several numerical experiments.

6.7 Conclusions

To reconstruct an unknown medium, the generalized Helmholtz inverse scattering problem uses data pairs consisting of the impinging and scattered wave fields, while Liouville inverse scattering problems uses data pairs consisting of incoming and outgoing wave location and direction. The former is regarded ill-posed in the high-frequency regime, while the latter is well-posed. This is intuitively contradicting to the fact that Liouville equation is the asymptotic limit of the Helmholtz equation.

We investigate this issue in this chapter. In particular, we develop a new formulation for studying the Helmholtz inverse scattering problem with a new data collection process, and we show that this new formulation, in the high-frequency limit, becomes the Liouville inverse scattering problem, and thus inherits the well-posedness nature. This discovery

bares the conceptual merit of providing the mathematical description of the wave-particle duality for light propagation in the inverse setting. In addition, this discovery also suggests a more stable numerical reconstruction process for studying the Helmholtz inverse scattering problem, which we showcase using several numerical experiments.

Chapter 7

Conclusion

This dissertation investigates data-driven numerical methods for multiscale elliptic and wave-type partial differential equations and their inverse problems. Our primary motivation is to integrate analytical insights about multiscale PDEs with data and numerical methodologies.

In the first part of the dissertation, we focus on developing efficient numerical solvers for elliptic multiscale PDEs. Although the discrete solution manifold of these PDEs is high-dimensional, it is essentially compressible if a homogenization limit exists. To compress the solution manifold, we employ two distinct strategies from data science, both within the domain decomposition framework and use Schwarz iteration. Our first strategy uses neural networks as surrogate models for the boundary-to-boundary map in each Schwarz iteration. In the second strategy, inspired by manifold learning techniques, the reduced solution manifolds are used as surrogates for local solvers.

The second part of the dissertation focuses on the inverse problems associated with wave-type PDEs in the high-frequency or classical limit. Observing the disparity in the stability between traditional inverse scattering problems and their asymptotic limits, we introduce new formulations for both a time-dependent and a time-independent inverse scattering problem. These formulations involve new data collection processes that align them with the Liouville inverse scattering problem, thus inheriting its well-posedness. This

discovery suggests that properly integrating data into the multiscale inverse problems is crucial for developing numerical reconstruction processes that are stable and reliable in practice.

We comment on some problems and directions that deserve further study. First, in Chapter 4, we propose using local bases over the solution manifold to construct numerical solutions for nonlinear PDEs, supported by numerical examples demonstrating the efficacy of this approach. In practice, the performance of this manifold-learning-based method crucially depends on the properties of the solution manifold, such as curvature and intrinsic dimension. A detailed error analysis of the method, along with a quantification of the sample complexity required to ensure accuracy and stability, would be interesting. Although the current approach is inspired by the Local Linear Embedding method, the perspective of the solution manifold could potentially be extended beyond this setting. Therefore, it would be interesting to explore adapting other geometric machine learning methods to solving PDEs, such as diffusion maps [81] and functional maps [180].

Second, in Chapter 6, we demonstrate that the new Helmholtz inverse scattering problem framework, when formulated as a PDE-constrained optimization problem, is asymptotically stable in the high-frequency limit. However, this setting has not yet been fully understood theoretically. Two main challenges remain. On the PDE level, the well-posedness of the inverse problem, particularly the stability of the new inverse scattering problem, needs clarification—specifically, how perturbations in the measured data affect the reconstructed medium. On the optimization level, it would be beneficial to study the landscape of the optimization problem, including the local and global minima of the objective functional. Addressing these challenges is crucial for understanding the increasing stability observed in numerical experiments and would require further exploration of related problems such as phase retrieval [116]. Additionally, while the current framework is asymptotically stable, the dimension of the measured data is doubled, leading to quadratic growth in computational complexity. It is uncertain whether the current algorithm will yield a better method, taking this into account. Therefore, developing efficient algorithms

based on the current framework would be an interesting direction for future research.

Appendix A

Sampling method for solution manifold

A.1 Sampling method for the semilinear elliptic equation

We explain here the sampling method for the semilinear elliptic equation in Section 4.3.4. To enforce the boundary condition on the physical boundary, patches that intersect this boundary should be treated differently from patches inside the domain. (We call the patch $\widetilde{\Omega}_m$ an "interior patch" if it satisfies $\partial\widetilde{\Omega}_m \cap \partial\Omega = \varnothing$, and a "boundary patch" otherwise.)

A.1.1 Sampling for interior patches

For the interior patch $\partial\widetilde{\Omega}_m$, each sample in $B(R_m; \widetilde{\mathcal{X}}_m)$ is decomposed into radial and angular parts $\widetilde{\phi} = rX$, with the two parts r and X sampled independently. The radial part r is generated so that $(\frac{r}{R_m})^D$ is uniformly distributed in the unit interval $[0, 1]$, where D is a preset integer. (We choose $D = 5$ and $R_m = R = 20$ in our tests.) The angular part X is a N_m-dimensional vector uniformly distributed in the set $\{X \in \mathbb{R}^{N_m} : \|X\|_{1/2} = 1\}$,

where N_m is the number of grid points on $\partial\widetilde{\Omega}_m$, and the norm $\|\cdot\|_{1/2}$ is defined by

$$\|\widetilde{\phi}\|_{1/2} = \sqrt{h\sum_{i=1}^{N_m}|\widetilde{\phi}_i|^2 + h^2\sum_{\substack{i,j=1\\ i\neq j}}^{N_m}\frac{|\widetilde{\phi}_i-\widetilde{\phi}_j|^2}{|z_i-z_j|^2}}\,.$$

Here $\widetilde{\phi} = (\widetilde{\phi}_i)_{i=1}^{N_m}$ is any discrete boundary condition, and z_i denotes the grid point on $\partial\widetilde{\Omega}_m$.

In order to generate X, let $Y_1,\ldots,Y_{N_m}\sim\mathcal{N}(0,1)$ be i.i.d. standard Gaussian random variables. Define the weight matrix $W=(W_{ij})_{N_m\times N_m}$ by

$$W_{ii} = h + \sum_{\substack{j=1\\ j\neq i}}^{N_m}\frac{2h^2}{|z_i-z_j|^2}\,,\quad W_{ij} = -\frac{2h^2}{|z_i-z_j|^2}\,,$$

and suppose that its Cholesky decomposition is $W = C^\top C$. Then the vector $Z = C^{-1}(Y_1,\ldots,Y_{N_m})^\top$ has uniform angular distribution with respect to the norm $\|\cdot\|_{1/2}$, so its normalization $X=\frac{Z}{\|Z\|_{1/2}}$ is uniformly distributed on the unit sphere $\{X\in\mathbb{R}^{N_m}: \|X\|_{1/2}=1\}$.

A.1.2 Sampling for boundary patches

Let

$$\widetilde{\phi}_m = \begin{bmatrix}\widetilde{\phi}_{m,d}\\ \widetilde{\phi}_{m,r}\end{bmatrix}\in\mathbb{R}^{N_m}$$

be a random sample, with $\widetilde{\phi}_{m,d}$ representing the physical boundary part and $\widetilde{\phi}_{m,r}$ representing the random part. When we rearrange the weight matrix W as

$$W = \begin{bmatrix}W_{dd} & W_{dr}\\ W_{rd} & W_{rr}\end{bmatrix}\,,$$

so that $\|\widetilde{\phi}_m\|_{1/2}^2 = \widetilde{\phi}_m^\top W \widetilde{\phi}_m$, then it yields

$$\widetilde{\phi}_{m,r}^\top W_{rr} \widetilde{\phi}_{m,r} = R_m^2 - \widetilde{\phi}_{m,d}^\top (W_{dd} - W_{dr} W_{rr}^{-1} W_{rd}) \widetilde{\phi}_{m,d}\,,$$

indicating that the random part lies in an ellipsoid.

Hence, the random part $\widetilde{\phi}_{m,r}$ can be sampled as follows. We decompose it into independently sampled radial and angular part $\widetilde{\phi}_r = r_m X_m$, so that r_m^D is uniformly distributed in the interval $\left[0, (R_m^2 - \widetilde{\phi}_{m,d}^\top (W_{dd} - W_{dr} W_{rr}^{-1} W_{rd}) \widetilde{\phi}_{m,d})^{D/2}\right]$, and X_m is uniformly distributed on the set $\{X_m : X_m^\top W_{rr} X_m = 1\}$.

A.2 Sampling method for the nonlinear radiative transfer equations

Here we describe the sampling method for the nonlinear radiative transfer equations discussed in Section 4.4.4.

To generate samples for the interior patches $\widetilde{\mathcal{K}}_m$, $m = 2, \ldots, M-1$, each sample is decomposed into radial and angular parts $\widetilde{\phi} = rX$, which are sampled independently. We take $(\frac{r}{R_m})^2$ to be uniformly distributed in $[0,1]$, while X is a (N_v+2)-dimensional vector uniformly distributed in the set $\{X \in \mathbb{R}^{N_v+2} : \|X\| = 1, X \geq 0\}$, where the norm $\|\cdot\|$ is defined by

$$\|\widetilde{\phi}\|^2 = \sum_{j=1}^{\frac{N_v}{2}} w_j |\widetilde{g}^{(2)}(s, v_j)|^2 + \sum_{j=\frac{N_v}{2}+1}^{N_v} w_j |\widetilde{g}^{(1)}(t, v_j)|^2 + |\widetilde{\theta}^{(1)}|^2 + |\widetilde{\theta}^{(2)}|^2\,,$$

given any discrete boundary condition

$$\widetilde{\phi} = \left(\{\widetilde{g}^{(2)}(s, v_j)\}_{j=1}^{\frac{N_v}{2}}, \{\widetilde{g}^{(1)}(t, v_j)\}_{j=\frac{N_v}{2}+1}^{N_v}, \widetilde{\theta}^{(1)}, \widetilde{\theta}^{(2)} \right).$$

Here N_v is the number of grid points in the velocity direction and the w_j are the Gaussian-

Legendre weights. (We choose $R_m = R = 25$ in our tests.)

To generate X, let $Y_1, \ldots, Y_{N_v+2} \sim \mathcal{N}(0,1)$ be i.i.d. standard Gaussian random variables. Denote the vector

$$Z = \left(\frac{Y_1}{\sqrt{w_1}}, \ldots, \frac{Y_{N_v}}{\sqrt{w_{N_v}}}, Y_{N_v+1}, Y_{N_v+2} \right) .$$

Then the normalized vector $X = \frac{Z}{\|Z\|}$ is uniformly distributed on the unit sphere $\{X \in \mathbb{R}^{N_v+2} : \|X\| = 1\}$. Note that (4.46) is invariant under x-translation, so we need only learn one interior dictionary on one interior patch, then re-use in for the other interior patches.

Sampling the boundary conditions on the boundary patches can be done in the same way. However, we do adjust the radius r. In particular, $(\frac{r}{R_{1/M}})^2$ is chosen uniformly in $[0,1]$, where $R_{1/M}$ has the fixed boundary condition deducted from R.

Appendix B

Formal derivation of Theorem 6.1

We start from the equation

$$\mathrm{i}k\alpha^k u^k + \Delta u^k + k^2 n(x) u^k = -S^k(x) = -k^{\frac{d+3}{2}} S(k(x - x_s)) \,, \quad x \in \mathbb{R}^d \,, \tag{B.1}$$

and assume that $\alpha^k \to \alpha \geq 0$ in the limit $k \to \infty$. We denote the density matrix of u^k satisfying (B.1) by

$$g^k(x, y) = u^k \left(x - \frac{y}{2k} \right) \overline{u^k} \left(x + \frac{y}{2k} \right) \,, \tag{B.2}$$

and the Fourier transform of a generic u by

$$\widehat{u}(v) = \mathcal{F}_{y \to v} u(y) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} e^{-\mathrm{i}yv} u(y) \mathrm{d}y \,. \tag{B.3}$$

The inverse Fourier transform is then

$$\mathcal{F}^{-1}_{v \to x} u(v) = \int_{\mathbb{R}^d} e^{\mathrm{i}xv} u(v) \mathrm{d}v \,. \tag{B.4}$$

Now we compute the equation satisfied by the Wigner transform. The first step is to compute the derivatives of g^k

$$\nabla_y \cdot \nabla_x g^k(x,y) = -\frac{1}{2k}\left[\Delta u^k\left(x-\frac{y}{2k}\right)\overline{u^k}\left(x+\frac{y}{2k}\right) - u^k\left(x-\frac{y}{2k}\right)\Delta\overline{u^k}\left(x+\frac{y}{2k}\right)\right], \tag{B.5}$$

and thus we have

$$\begin{aligned}\alpha^k g^k + \mathrm{i}\nabla_y \cdot \nabla_x g^k(x,y) + \frac{\mathrm{i}k}{2}\left[n\left(x+\frac{y}{2k}\right) - n\left(x-\frac{y}{2k}\right)\right] g^k(x,y) = \\ = \sigma^k(x,y) \\ := \frac{\mathrm{i}}{2k}\left[S^k\left(x-\frac{y}{2k}\right)\overline{u^k}\left(x+\frac{y}{2k}\right) - \overline{S^k}\left(x+\frac{y}{2k}\right)u^k\left(x-\frac{y}{2k}\right)\right].\end{aligned} \tag{B.6}$$

Therefore, after a Fourier transform, we obtain the following transport equation on the Wigner transform f^k

$$\alpha^k f^k(x,v) + v\cdot\nabla_x f^k(x,v) + Z^k(x,v) *_v f^k(x,v) = Q^k(x,v)\,, \tag{B.7}$$

where the last term denotes the convolution in v

$$Z^k(x,v) *_v f^k(x,v) = \int_{\mathbb{R}^d} Z^k(x,v-p) f^k(x,p)\mathrm{d}p$$

and the quantities Z^k, Q^k arising in this equation are given by

$$\begin{aligned} Z^k(x,v) &= \frac{1}{(2\pi)^d}\frac{\mathrm{i}k}{2}\mathcal{F}^{-1}_{y\to v}\left[n\left(x+\frac{y}{2k}\right) - n\left(x-\frac{y}{2k}\right)\right], \\ Q^k(x,v) &= \frac{1}{(2\pi)^d}\mathcal{F}^{-1}_{y\to v}\sigma^k(x,y)\,. \end{aligned} \tag{B.8}$$

From this equation we can compute the formally compute the limits. For Z^k we have that

$$Z^k(x,v) \xrightarrow{k\to\infty} \frac{1}{(2\pi)^d}\frac{\mathrm{i}}{2}(\mathcal{F}^{-1}_{y\to v}y)\cdot\nabla_x n(x) = -\frac{1}{2}\nabla_x n(x)\cdot\nabla_v\delta(v)\,. \tag{B.9}$$

The limit of the source term Q^k is slightly more involved. First, we define the complex valued function

$$w^k(y) = \frac{1}{k^{\frac{d-1}{2}}} u^k\left(x_s + \frac{y}{k}\right), \tag{B.10}$$

which after a change of variable can be rewritten as

$$u^k(x) = k^{\frac{d-1}{2}} w^k(k(x - x_s)), \tag{B.11}$$

where function w^k satisfies the rescaled Helmholtz equation

$$\mathrm{i}\frac{\alpha^k}{k} w^k + \Delta w^k + n\left(x_s + \frac{y}{k}\right) w^k = -S(y). \tag{B.12}$$

In the high-frequency limit, w^k converges towards a solution w of

$$\Delta w + n(x_s) w = -S(y). \tag{B.13}$$

The second step is to compute the Fourier transform of w. To do so, we add an absorption term to the equation above, resulting in

$$\mathrm{i}\beta w + \Delta w + n(x_s) w = -S(y). \tag{B.14}$$

where $\beta > 0$. This new term is used as a broadening factor, which helps to smooth the Fourier transform. We perform a Fourier transform on both sides, which leads to

$$\widehat{w}(v) = \frac{-\hat{S}(v)}{n(x_s) - |v|^2 + \mathrm{i}\beta} = \hat{S}(v)\hat{G}(v;\beta). \tag{B.15}$$

where $\hat{G}(v;\beta)$ denotes the Fourier transform of the outgoing Green's function that vanishes at infinity

$$\hat{G}(v;\beta) \equiv -\frac{1}{n(x_s) - |v|^2 + \mathrm{i}\beta} = -\frac{n(x_s) - |v|^2}{(n(x_s) - |v|^2)^2 + \beta^2} + \frac{\mathrm{i}\beta}{(n(x_s) - |v|^2)^2 + \beta^2}, \quad \beta > 0. \tag{B.16}$$

As usual, we take the limit $\beta \to 0+$. The first term converges weakly to the principal value

$$-\frac{n(x_s)-|v|^2}{(n(x_s)-|v|^2)^2+\beta^2} \xrightarrow{\beta\to 0+} -\text{P.V.}\left(\frac{1}{n(x_s)-|v|^2}\right). \tag{B.17}$$

The second term converges to a delta function on the sphere $\{|v|^2 = n(x_s)\}$ as $\beta \to 0+$

$$\frac{\mathrm{i}\beta}{(n(x_s)-|v|^2)^2+\beta^2} \xrightarrow{\beta\to 0+} \frac{\mathrm{i}\pi}{2}\delta(|v|^2 = n(x_s)). \tag{B.18}$$

In summary, we obtain the Fourier transform of the outgoing solution to (B.13)

$$\widehat{w}(v) = \lim_{\beta\to 0+} \hat{S}(v)\hat{G}(v;\beta) = \hat{S}(v)\left[\frac{\mathrm{i}\pi}{2}\delta(|v|^2 = n(x_s)) - \text{P.V.}\left(\frac{1}{n(x_s)-|v|^2}\right)\right]. \tag{B.19}$$

Now we are ready to compute Q^k. We take two test functions $\phi(x)$ and $\psi(y)$

$$\begin{aligned}
&\int_{\mathbb{R}^{2d}} \sigma^k(x,y)\phi(x)\psi(y)\,\mathrm{d}x\,\mathrm{d}y \\
&= \frac{\mathrm{i}}{2k}\int_{\mathbb{R}^{2d}} \left[S^k\left(x-\frac{y}{2k}\right)\overline{u^k}\left(x+\frac{y}{2k}\right) - \overline{S^k}\left(x+\frac{y}{2k}\right)u^k\left(x-\frac{y}{2k}\right)\right]\phi(x)\psi(y)\mathrm{d}x\mathrm{d}y \\
&= \frac{\mathrm{i}k^d}{2}\int_{\mathbb{R}^d}\Bigg[S\left(k\left(x-\frac{y}{2k}-x_s\right)\right)\overline{w^k}\left(k\left(x+\frac{y}{2k}-x_s\right)\right) \\
&\qquad\qquad - \overline{S}\left(k\left(x+\frac{y}{2k}-x_s\right)\right)w^k\left(k\left(x-\frac{y}{2k}-x_s\right)\right)\Bigg]\phi(x)\psi(y)\mathrm{d}x\mathrm{d}y \\
&= \frac{\mathrm{i}}{2}\int_{\mathbb{R}^{2d}}\left[S(z)\overline{w^k}(z+y)\phi\left(\frac{z}{k}+\frac{y}{2k}+x_s\right) - \overline{S}(z)w^k(z-y)\phi\left(\frac{z}{k}-\frac{y}{2k}+x_s\right)\right]\psi(y)\mathrm{d}z\mathrm{d}y \\
&\xrightarrow{k\to\infty} \frac{\mathrm{i}}{2}\phi(x_s)\int_{\mathbb{R}^{2d}}\left[S(z)\overline{w}(z+y) - \overline{S}(z)w(z-y)\right]\psi(y)\mathrm{d}z\mathrm{d}y\,.
\end{aligned} \tag{B.20}$$

In other words, we have formally obtained that

$$\sigma^k(x,y) \xrightarrow{k\to\infty} \frac{\mathrm{i}}{2}\delta(x-x_s)\int_{\mathbb{R}^d}\left[S(z)\overline{w}(z+y) - \overline{S}(z)w(z-y)\right]\mathrm{d}z\,, \tag{B.21}$$

which after a Fourier transform gives

$$\begin{aligned} Q^k(x,v) &= \frac{1}{(2\pi)^d}\mathcal{F}^{-1}_{y\to v}\sigma^k(x,y) \\ &\xrightarrow{k\to\infty} \frac{1}{(2\pi)^d}\frac{\mathrm{i}}{2}\delta(x-x_s)\mathcal{F}^{-1}_{y\to v}\left\{\int_{\mathbb{R}^d}\left[S(z)\overline{w}(z+y)-\overline{S}(z)w(z-y)\right]\mathrm{d}z\right\} \\ &= \frac{\mathrm{i}}{2}\delta(x-x_s)(2\pi)^d\left[\hat{S}(v)\overline{\hat{w}(v)}-\overline{\hat{S}(v)}\hat{w}(v)\right] \\ &= (2\pi)^d\delta(x-x_s)\mathrm{Im}\left[\overline{\hat{S}(v)}\hat{w}(v)\right]\,. \end{aligned} \tag{B.22}$$

We finally obtain

$$Q^k(x,v) \xrightarrow{k\to\infty} (2\pi)^d\frac{\pi}{2}\delta(x-x_s)|\hat{S}(v)|^2\delta(|v|^2=n(x_s))\,. \tag{B.23}$$

by substituting (B.19) in (B.22).

Bibliography

[1] N.B. Abdallah, M. Puel, and M.S. Vogelius. "Diffusion and homogenization limits with separate scales". In: *Multiscale Model. Simul.* 10.4 (2012), pp. 1148–1179.

[2] A. Abdulle and Y. Bai. "Reduced basis finite element heterogeneous multiscale method for high-order discretizations of elliptic homogenization problems". In: *J. Comput. Phys.* 231.21 (2012), pp. 7014–7036.

[3] A. Abdulle, Y. Bai, and G. Vilmart. "Reduced basis finite element heterogeneous multiscale method for quasilinear elliptic homogenization problems". In: *Discrete Contin. Dyn. Syst. Ser. S* 8.1 (2015), pp. 91–118.

[4] A. Abdulle and C. Schwab. "Heterogeneous multiscale FEM for diffusion problems on rough surfaces". In: *Multiscale Model. Simul.* 3.1 (2005), pp. 195–220.

[5] A. Abdulle and G. Vilmart. "Analysis of the finite element heterogeneous multiscale method for quasilinear elliptic homogenization problems". In: *Math. Comp.* 83.286 (2014), pp. 513–536.

[6] V. Agoshkov. *Boundary Value Problems for Transport Equations*. Springer Science & Business Media, 2012.

[7] R. Alaifari et al. "Stable phase retrieval in infinite dimensions". In: *Found. Comput. Math.* 19.4 (2019), pp. 869–900.

[8] G. Allaire. "Homogenization and Two-Scale Convergence". In: *SIAM J. Math. Anal.* 23.6 (1992), pp. 1482–1518.

[9] W. K. Allard, G. Chen, and M. Maggioni. "Multi-scale geometric methods for data sets II: Geometric multi-resolution analysis". In: *Appl. Comput. Harmon. Anal.* 32.3 (2012), pp. 435–462.

[10] H. Amann and J. Moser. "On the existence of positive solutions of nonlinear elliptic boundary value problems". In: *Indiana Univ. Math. J.* 21.2 (1971), pp. 125–146.

[11] D. G. Anderson. "Iterative procedures for nonlinear integral equations". In: *J.ACM* 12.4 (1965), pp. 547–560.

[12] A. Andoni, P. Indyk, and I. Razenshteyn. "Approximate nearest neighbor search in high dimensions". In: *Proceedings of the International Congress of Mathematicians—Rio de Janeiro 2018. Vol. IV. Invited lectures*. World Sci. Publ., Hackensack, NJ, 2018, 3287–3318, arXiv preprint arXiv:1806.09823.

[13] G. Astarita and G. Marrucci. *Principles of non-Newtonian fluid mechanics*. McGraw-Hill Companies, 1974.

[14] D. Atkinson and N. D. Aparicio. "An inverse problem method for crack detection in viscoelastic materials under anti-plane strain". In: *Int. J. Eng. Sci.* 35.9 (1997), pp. 841–849. ISSN: 0020-7225.

[15] I. Babuška. "Homogenization and its application. Mathematical and computational problems". In: *Numerical solution of partial differential equations–III*. Elsevier, 1976, pp. 89–116.

[16] I. Babuška and R. Lipton. "Optimal local approximation spaces for generalized finite element methods with application to multiscale problems". In: *Multiscale Model. Simul.* 9.1 (2011), pp. 373–406.

[17] I. Babuška and J. M. Melenk. "The partition of unity method". In: *Internat. J. Numer. Methods Engrg.* 40.4 (1997), pp. 727–758.

[18] I. Babuška and J. Osborn. "Can a finite element method perform arbitrarily badly?" In: *Mathematics of computation* 69.230 (2000), pp. 443–462.

[19] I. Babuška and S. Sauter. "Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?" In: *SIAM J. Numer. Anal.* 34.6 (1997), pp. 2392–2423.

[20] I. Babuška, R. Tempone, and G. E. Zouraris. "Galerkin finite element approximations of stochastic elliptic partial differential equations". In: *SIAM J. Numer. Anal.* 42.2 (2004), pp. 800–825.

[21] G. Bal. "Hybrid inverse problems and internal functionals". In: *Inverse problems and applications: inside out. II* 60 (2013), pp. 325–368.

[22] G. Bal, T. Komorowski, and L. Ryzhik. "Kinetic limits for waves in a random medium". In: *Kinet. Relat. Models* 3.4 (2010), pp. 529–644.

[23] G. Bal, G. Papanicolaou, and L. Ryzhik. "Radiative transport limit for the random Schrödinger equation". In: *Nonlinearity* 15.2 (2002), p. 513.

[24] G. Bal and O. Pinaud. "Kinetic Models for Imaging in Random Media". In: *Multiscale Model. Simul.* 6.3 (2007), pp. 792–819.

[25] G. Bal, O. Pinaud, and L. Ryzhik. "Random Media in Inverse Problems, Theoretical Aspects". In: *Encyclopedia of Applied and Computational Mathematics*. Ed. by B. Engquist. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 1219–1222.

[26] G. Bal and K. Ren. "Multi-source quantitative photoacoustic tomography in a diffusive regime". In: *Inverse Problems* 27.7 (2011), p. 075003.

[27] G. Bal and K. Ren. "Transport-Based Imaging in Random Media". In: *SIAM J. Appl. Math.* 68.6 (2008), pp. 1738–1762.

[28] G. Bal and G. Uhlmann. "Inverse diffusion theory of photoacoustics". In: *Inverse Problems* 26.8 (June 2010), p. 085010.

[29] G. Bal et al. "Quantitative thermo-acoustics and related problems". In: *Inverse Problems* 27.5 (2011), p. 055007.

[30] G. Bao and H. Zhang. "Sensitivity analysis of an inverse problem for the wave equation with caustics". In: *J. Amer. Math. Soc* 27.4 (2014), pp. 953–981.

[31] G. Bao et al. “Inverse scattering problems with multi-frequencies”. In: *Inverse Problems* 31.9 (2015), p. 093001.

[32] C. Bardos, F. Golse, and D. Levermore. “Fluid dynamic limits of kinetic equations. I. Formal derivations”. In: *J. Stat. Phys.* 63.1-2 (1991), pp. 323–344.

[33] C. Bardos, F. Golse, and B. Perthame. “The Rosseland approximation for the radiative transfer equations”. In: *Comm. Pure Appl. Math.* 40.6 (1987), pp. 691–721.

[34] C. Bardos, G. Lebeau, and J. Rauch. “Sharp sufficient conditions for the observation, control, and stabilization of waves from the boundary”. In: *SIAM J. Control Optim.* 30.5 (1992), pp. 1024–1065.

[35] C. Bardos, R. Santos, and R. Sentis. “Diffusion approximation and computation of the critical size”. In: *Trans. Amer. Math. Soc.* 284.2 (1984), pp. 617–649.

[36] C. Bardos et al. “The nonaccretive radiative transfer equations: existence of solutions and Rosseland approximation”. In: *J. Funct. Anal.* 77.2 (1988), pp. 434–460.

[37] A. R. Barron. “Universal approximation bounds for superpositions of a sigmoidal function”. In: *IEEE Trans. Inform. Theory* 39.3 (1993), pp. 930–945.

[38] A. Barth, C. Schwab, and N. Zollinger. “Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients”. In: *Numer. Math.* 119.1 (2011), pp. 123–161.

[39] P. Beard. “Biomedical photoacoustic imaging”. In: *Interface focus* 1.4 (2011), pp. 602–631.

[40] M. Bebendorf. “Why finite element discretizations can be factored by triangular hierarchical matrices”. In: *SIAM J. Numer. Anal.* 45.4 (2007), pp. 1472–1494.

[41] M. Belkin and P. Niyogi. “Laplacian eigenmaps and spectral techniques for embedding and clustering”. In: *Adv. in Neural Inform. Process. Systems*. 2002, pp. 585–591.

[42] J.-D. Benamou et al. “High frequency limit of the Helmholtz equations”. In: *Rev. Mat. Iberoam.* 18.1 (2002), pp. 187–209.

[43] A. Bensoussan, L. Boccardo, and F. Murat. “H convergence for quasi-linear elliptic equations with quadratic growth”. In: *Appl. Math. Optim.* 26.3 (1992), pp. 253–272.

[44] A. Bensoussan, J.-L. Lions, and G.C. Papanicolaou. *Asymptotic Analysis for Periodic Structures*. Vol. 374. American Mathematical Soc., 2011.

[45] A. Bensoussan, J.-L. Lions, and G.C. Papanicolaou. “Boundary layers and homogenization of transport processes”. In: *Publ. Res. Inst. Math. Sci.* 15.1 (1979), pp. 53–157.

[46] J. Berner, M. Dablander, and P. Grohs. “Numerically solving parametric families of high-dimensional Kolmogorov partial differential equations via deep learning”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16615–16627.

[47] M. Bieri and C. Schwab. "Sparse high order FEM for elliptic SPDEs". In: *Comput. Methods Appl. Mech. Engrg.* 198.13-14 (2009), pp. 1149–1170.

[48] C. Borges, A. Gillman, and L. Greengard. "High Resolution Inverse Scattering in Two Dimensions Using Recursive Linearization". In: *SIAM J. Imaging Sci.* 10.2 (2017), pp. 641–664.

[49] M. Born. "Quantenmechanik der Stoßvorgänge". In: *Zeitschrift für Physik* 38.11 (1926), pp. 803–827.

[50] M. de Buhan and M. Darbas. "Numerical resolution of an electromagnetic inverse medium problem at fixed frequency". In: *Comput. Math. Appl.* 74.12 (2017), pp. 3111–3128. ISSN: 0898-1221.

[51] M. de Buhan and M. Kray. "A new approach to solve the inverse scattering problem for waves: combining the TRAC and the adaptive inversion methods". In: *Inverse Problems* 29.8 (July 2013), p. 085009.

[52] A. Buhr and K. Smetana. "Randomized local model order reduction". In: *SIAM J. Sci. Comput.* 40.4 (2018), A2120–A2151.

[53] R. H. Byrd et al. "A Limited Memory Algorithm for Bound Constrained Optimization". In: *SIAM J. Sci. Comput.* 16.5 (1995), pp. 1190–1208.

[54] L. A. Caffarelli and X. Cabré. *Fully nonlinear elliptic equations*. Vol. 43. American Mathematical Soc., 1995.

[55] A. P. Calderón. "On an inverse boundary value problem". In: *Comput. Appl. Math.* 25.2-3 (2006), pp. 133–138.

[56] J.A. Carrillo et al. "A WENO-solver for the transients of Boltzmann–Poisson system for semiconductor devices: performance and comparisons with Monte Carlo methods". In: *J. Comput. Phys.* 184.2 (2003), pp. 498–525.

[57] F. Castella, B. Perthame, and O. Runborg. "High frequency limit of the Helmholtz equation. II. Source on a general smooth manifold". In: *Comm. Partial Differential Equations* (2002).

[58] A. Cazé and J.C. Schotland. "Diagrammatic and asymptotic approaches to the origins of radiative transport theory: tutorial". In: *JOSA A* 32.8 (2015), pp. 1475–1484.

[59] S. Chandrasekhar. *An introduction to the study of stellar structure*. Vol. 2. Courier Corporation, 1957.

[60] K. Chen, Q. Li, and J. -G. Liu. "Online learning in optical tomography: a stochastic approach". In: *Inverse Problems.* 34.7 (2018), p. 075010.

[61] K. Chen, Q. Li, and L. Wang. "Stability of stationary inverse transport equation in diffusion scaling". In: *Inverse Problems* 34.2 (2018), p. 025004.

[62] K. Chen, Q. Li, and S. J. Wright. "Schwarz iteration method for elliptic equation with rough media based on random sampling". In: *Proceedings of the International Consortium of Chinese Mathematicians 2018*. Int. Press, Boston, MA, 2020, pp. 163–178.

[63] K. Chen et al. “A Low-Rank Schwarz Method for Radiative Transfer Equation With Heterogeneous Scattering Coefficient”. In: *Multiscale Model. Simul.* 19.2 (2021), pp. 775–801.

[64] K. Chen et al. “Random sampling and efficient algorithms for multiscale PDEs”. In: *SIAM J. Sci. Comput* 42.5 (2020), A2974–A3005.

[65] K. Chen et al. “Randomized Sampling for Basis Function Construction in Generalized Finite Element Methods”. In: *Multiscale Model. Simul.* 18.2 (2020), pp. 1153–1177.

[66] S. Chen and Q. Li. “Semiclassical limit of an inverse problem for the Schrödinger equation”. In: *Res. Math. Sci.* 8.3 (2021), pp. 1–18.

[67] S. Chen, Q. Li, and X. Yang. “Classical limit for the varying-mass Schrödinger equation with random inhomogeneities”. In: *J. Comput. Phys.* 438 (2021), p. 110365.

[68] S. Chen et al. *A two-layer neural network-based reduced order Schwarz method for fully nonlinear multiscale elliptic PDEs*. 2021. URL: `https://github.com/simonchenthu/nn_reduced_Schwarz`.

[69] S. Chen et al. “High-frequency limit of the inverse scattering problem: Asymptotic convergence from inverse Helmholtz to inverse Liouville”. In: *SIAM J. Imaging Sci.* 16.1 (2023), pp. 111–143.

[70] S. Chen et al. *Inverse scattering with Husimi data*. Version 1.0.0. Dec. 2021. URL: `https://github.com/simonchenthu/inverse_scattering/tree/husimi`.

[71] S. Chen et al. “Manifold learning and nonlinear homogenization”. In: *Multiscale Model. Simul.* 20.3 (2022), pp. 1093–1126.

[72] Y. Chen. “Inverse scattering via Heisenberg’s uncertainty principle”. In: *Inverse Problems* 13.2 (1997), p. 253.

[73] Z. Chen and T. Y. Savchuk. “Analysis of the multiscale finite element method for nonlinear and random homogenization problems”. In: *SIAM J. Numer. Anal.* 46.1 (2008), pp. 260–279.

[74] M. Cheney. “A Mathematical Tutorial on Synthetic Aperture Radar”. In: *SIAM Rev.* 43.2 (2001), pp. 301–312.

[75] M. Chipot. *Elliptic Qquations: An Introductory Course*. Springer, 2009.

[76] E. Chung et al. “A multiscale model reduction method for nonlinear monotone elliptic equations in heterogeneous media”. In: *Netw. Heterog. Media* 12.4 (2017), p. 619.

[77] E. Chung et al. “A new phase space method for recovering index of refraction from travel times”. In: *Inverse Problems* 23.1 (2007), p. 309.

[78] E. Chung et al. “Cluster-based generalized multiscale finite element method for elliptic PDEs with random coefficients”. In: *J. Comput. Phys.* 371 (2018), pp. 606–617.

[79] E. Chung et al. “Nonlinear nonlocal multicontinua upscaling framework and its applications”. In: *International Journal for Multiscale Computational Engineering* 16.5 (2018).

[80] A. Cohen, R. Devore, and C. Schwab. "Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's". In: *Anal. Appl.* 9.01 (2011), pp. 11–47.

[81] R. R. Coifman et al. "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps". In: *Proc. Natl. Acad. Sci. USA* 102.21 (2005), pp. 7426–7431.

[82] A. Colli et al. "The Use of a Pocket-Sized Ultrasound Device Improves Physical Examination: Results of an In-and Outpatient Cohort Study". In: *PLOS ONE* 10.3 (Mar. 2015), pp. 1–10.

[83] D. L. Colton and R. Kress. *Inverse Acoustic and Electromagnetic Scattering Theory.* Vol. 93. Springer, 2019.

[84] F. Coron and B. Perthame. "Numerical passage from kinetic to fluid equations". In: *SIAM J. Numer. Anal.* 28.1 (1991), pp. 26–42.

[85] S. Cuomo et al. "Scientific machine learning through physics–informed neural networks: Where we are and what's next". In: *J. Sci. Comput.* 92.3 (2022), p. 88.

[86] P. Degond. "Asymptotic-preserving schemes for fluid models of plasmas". In: *CEMRACS 2010: Numerical methods for fusion, arXiv preprint arXiv:1104.1869* (2011).

[87] E. Di Nezza, G. Palatucci, and E. Valdinoci. "Hitchhiker's guide to the fractional Sobolev spaces". In: *Bull. Sci. Math.* 136.5 (2012), pp. 521–573.

[88] G. Dimarco and L. Pareschi. "High order asymptotic-preserving schemes for the Boltzmann equation". In: *C. R. Math. Acad. Sci. Paris* 350.9-10 (2012), pp. 481–486.

[89] L. Dumas and F. Golse. "Homogenization of transport equations". In: *SIAM J. Appl. Math.* 60.4 (2000), pp. 1447–1470.

[90] W. E and B. Engquist. "The heterogeneous multiscale methods". In: *Commun. Math. Sci.* 1.1 (2003), pp. 87–132.

[91] W. E, J. Han, and A. Jentzen. "Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations". In: *Commun. Math. Stat.* 5.4 (2017), pp. 349–380.

[92] W. E, C. Ma, and L. Wu. "The Barron space and the flow-induced function spaces for neural network models". In: *Constr. Approx.* 55.1 (2022), pp. 369–406.

[93] W. E, P. Ming, and P. Zhang. "Analysis of the heterogeneous multiscale method for elliptic homogenization problems". In: *J. Amer. Math. Soc.* 18.1 (2005), pp. 121–156.

[94] W. E and B. Yu. "The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems". In: *Commun. Math. Stat.* 6.1 (2018), pp. 1–12.

[95] W. E et al. "Towards a mathematical understanding of neural network-based machine learning: What we know and what we don't". In: *CSIAM Trans. Appl. Math.* 1.4 (2020), pp. 561–615.

[96] Y. Efendiev and T. Y. Hou. *Multiscale finite element methods: theory and applications.* Vol. 4. Springer Science & Business Media, 2009.

[97] Y. Efendiev, T. Y. Hou, and V. Ginting. "Multiscale finite element methods for nonlinear problems and their applications". In: *Commun. Math. Sci.* 2.4 (2004), pp. 553–589.

[98] Y. Efendiev, T. Y. Hou, and X.-H. Wu. "Convergence of a nonconforming multiscale finite element method". In: *SIAM J. Numer. Anal.* 37.3 (2000), pp. 888–910.

[99] Y. Efendiev et al. "Generalized multiscale finite element methods. Nonlinear elliptic equations". In: *Commun. Comput. Phys.* 15.3 (2014), pp. 733–755.

[100] B. Engquist and O. Runborg. "Computational high frequency wave propagation". In: *Acta Numer.* 12 (2003), pp. 181–266.

[101] L. C. Evans. "Periodic homogenisation of certain fully nonlinear partial differential equations". In: *Proc. Roy. Soc. Edinburgh Sect. A* 120.3-4 (1992), pp. 245–265.

[102] Y. Fan, C. O. Bohorquez, and L. Ying. "BCR-Net: A neural network based on the nonstandard wavelet form". In: *J. Comput. Phys.* 384 (2019), pp. 1–15.

[103] Y. Fan et al. "A multiscale neural network based on hierarchical matrices". In: *Multiscale Model. Simul.* 17.4 (2019), pp. 1189–1213.

[104] H-R Fang and Y. Saad. "Two classes of multisecant methods for nonlinear acceleration". In: *Numer. Linear Algebra Appl.* 16.3 (2009), pp. 197–221.

[105] P. Frauenfelder, C. Schwab, and R. A. Todor. "Finite elements for elliptic problems with stochastic coefficients". In: *Comput. Methods Appl. Mech. Engrg.* 194.2-5 (2005), pp. 205–228.

[106] E. Gabetta, L. Pareschi, and G. Toscani. "Relaxation schemes for nonlinear kinetic equations". In: *SIAM J. Numer. Anal.* 34.6 (1997), pp. 2168–2194.

[107] M. J. Gander. "Schwarz methods over the course of time". In: *Electron. Trans. Numer. Anal.* 31 (2008), pp. 228–255.

[108] M. G. D. Geers et al. "Homogenization methods and multiscale modeling: nonlinear problems". In: *Encyclopedia of Computational Mechanics Second Edition* (2017), pp. 1–34.

[109] P. Gérard et al. "Homogenization limits and Wigner transforms". In: *Comm. Pure Appl. Math.* 50.4 (1997), pp. 323–379.

[110] R. G. Ghanem and P. D. Spanos. *Stochastic finite elements: a spectral approach*. Courier Corporation, 2003.

[111] D. Gilbarg and N.S. Trudinger. *Elliptic Partial Differential Equations of Second Order*. Springer, 2015.

[112] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. Cambridge: MIT Press, 2016.

[113] J.W. Goodman. *Introduction to Fourier optics*. Roberts and Company Publishers, 2005.

[114] T. Goudon and A. Mellet. "Diffusion approximation in heterogeneous media". In: *Asymptot. Anal.* 28.3, 4 (2001), pp. 331–358.

[115] T. Goudon and A. Mellet. “Homogenization and diffusion asymptotics of the linear Boltzmann equation”. In: *ESAIM Control Optim. Calc. Var.* 9 (2003), pp. 371–398.

[116] P. Grohs, S. Koppensteiner, and M. Rathmair. “Phase retrieval: Uniqueness and stability”. In: *SIAM Rev.* 62.2 (2020), pp. 301–350.

[117] P. Grohs and M. Rathmair. “Stable Gabor phase retrieval and spectral clustering”. In: *Comm. Pure Appl. Math.* 72.5 (2019), pp. 981–1043.

[118] Y. Guo and L. Wu. “Geometric Correction in Diffusive Limit of Neutron Transport Equation in 2D Convex Domains”. In: *Arch. Ration. Mech. Anal.* 226.1 (2017), pp. 321–403.

[119] W. Hackbusch. *Hierarchical matrices: algorithms and analysis*. Vol. 49. Heidelberg: Springer, 2015.

[120] P. Hähner and T. Hohage. “New stability estimates for the inverse acoustic inhomogeneous medium problem and applications”. In: *SIAM J. Math. Anal.* 33.3 (2001), pp. 670–685.

[121] N. Halko, P.-G. Martinsson, and J. A. Tropp. “Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions”. In: *SIAM Rev.* 53.2 (2011), pp. 217–288.

[122] P. Henning, A. Målqvist, and D. Peterseim. “A localized orthogonal decomposition method for semi-linear elliptic problems”. In: *ESAIM Math. Model. Numer. Anal.* 48.5 (2014), pp. 1331–1349.

[123] S. Holman, F. Monard, and P. Stefanov. “The attenuated geodesic X-ray transform”. In: *Inverse Problems* 34.6 (2018), p. 064003.

[124] L. Hörmander. *The Analysis of Linear Partial Differential Operators III*. Springer Berlin Heidelberg, 1985.

[125] M. F. Horstemeyer. *Integrated Computational Materials Engineering (ICME) for metals: using multiscale modeling to invigorate engineering design with science*. John Wiley & Sons, 2012.

[126] J.G. Hoskins, J. Kraisler, and J.C. Schotland. “Radiative transport in quasi-homogeneous random media”. In: *JOSA A* 35.11 (2018), pp. 1855–1860.

[127] T. Y. Hou, Q. Li, and P. Zhang. “Exploring the locally low dimensional structure in solving random elliptic PDEs”. In: *Multiscale Model. Simul.* 15.2 (2017), pp. 661–695.

[128] T. Y. Hou and X.-H. Wu. “A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media”. In: *J. Comput. Phys.* 134.1 (1997), pp. 169–189.

[129] T. Y. Hou, X.-H. Wu, and Z. Cai. “Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients”. In: *Math. Comp.* 68.227 (1999), pp. 913–943.

[130] T. Y. Hou et al. “An iteratively adaptive multi-scale finite element method for elliptic PDEs with rough coefficients”. In: *J. Comput. Phys.* 336 (2017), pp. 375–400.

[131] J. Hu, S. Jin, and Q. Li. "Asymptotic-preserving schemes for multiscale hyperbolic and kinetic equations". In: *Handbook of Numerical Analysis*. Vol. 18. Elsevier, 2017, pp. 103–129.

[132] Y.Q. Huang, R. Li, and W. Liu. "Preconditioned descent algorithms for p-Laplacian". In: *J. Sci. Comput.* 32.2 (2007), pp. 343–371.

[133] A. M. Il'in. "Differencing scheme for a differential equation with a small parameter affecting the highest derivative". In: *Mathematical Notes of the Academy of Sciences of the USSR* 6.2 (1969), pp. 596–602.

[134] P. Indyk and R. Motwani. "Approximate nearest neighbors: towards removing the curse of dimensionality". In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. 1998, pp. 604–613.

[135] V. Isakov. *Inverse Problems for Partial Differential Equations*. Third. Vol. 127. Applied Mathematical Sciences. Springer, Cham, 2017, pp. xv+406.

[136] H. Ishii and P.-L. Lions. "Viscosity solutions of fully nonlinear second-order elliptic partial differential equations". In: *J. Differential Equations* 83.1 (1990), pp. 26–78.

[137] G.-S. Jiang and C.-W. Shu. "Efficient implementation of weighted ENO schemes". In: *J. Comput. Phys.* 126.1 (1996), pp. 202–228.

[138] S. Jin. "Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations". In: *SIAM J. Sci. Comput.* 21.2 (1999), pp. 441–454.

[139] W. B. Johnson and J. Lindenstrauss. "Extensions of Lipschitz mappings into a Hilbert space". In: *Contemp. Math.* 26.189-206 (1984), p. 1.

[140] A. Jollivet. "Inverse scattering at high energies for the multidimensional Newton equation in a long range potential". In: *Asymptot. Anal.* 90.1-2 (2014), pp. 105–132.

[141] A. Jollivet. "On inverse scattering at fixed energy for the multidimensional Newton equation in a non-compactly supported field". In: *J. Inverse Ill-Posed Probl.* 21.6 (2013), pp. 713–734.

[142] D. D. Joseph and T. S. Lundgren. "Quasilinear Dirichlet problems driven by positive sources". In: *Arch. Ration. Mech. Anal.* 49.4 (1973), pp. 241–269.

[143] J.B. Keller, I. Kay, and J. Shmoys. "Determination of the Potential from Scattering Data". In: *Phys. Rev.* 102 (2 Apr. 1956), pp. 557–559.

[144] Y. Khoo and L. Ying. "SwitchNet: a neural network model for forward and inverse scattering problems". In: *SIAM J. Sci. Computing* 41.5 (2019), A3182–A3201.

[145] M. C. King. "Chapter 2 - Principles of Optical Lithography". In: *VLSI Electronics: Microstructure Science*. Ed. by Norman G. Einspruch. Vol. 1. VLSI Electronics Microstructure Science. Elsevier, 1981, pp. 41–81.

[146] D. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations (ICLR)*. San Diega, CA, USA, 2015.

[147] A. Kirsch. *An Introduction to the Mathematical Theory of Inverse Problems*. Vol. 120. Springer, Jan. 2011.

[148] S. Klainerman and A. Majda. "Compressible and incompressible fluids". In: *Comm. Pure Appl. Math.* 35.5 (1982), pp. 629–651.

[149] S. Klainerman and A. Majda. "Singular limits of quasilinear hyperbolic systems with large parameters and the incompressible limit of compressible fluids". In: *Comm. Pure Appl. Math.* 34.4 (1981), pp. 481–524.

[150] A. Klar and C. Schmeiser. "Numerical passage from radiative heat transfer to nonlinear diffusion models". In: *Math. Models Methods Appl. Sci.* 11.05 (2001), pp. 749–767.

[151] A. Klar and N. Siedow. "Boundary layers and domain decomposition for radiative heat transfer and diffusion equations: applications to glass manufacturing process". In: *European J. Appl. Math.* 9.4 (1998), pp. 351–372.

[152] J. M. Klusowski and A. R. Barron. "Risk bounds for high-dimensional ridge function combinations including neural networks". In: *arXiv preprint arXiv:1607.01434* (2016).

[153] R.-Y. Lai, Q. Li, and G. Uhlmann. "Inverse problems for the stationary transport equation in the diffusion scaling". In: *SIAM J. Appl. Math.* 79.6 (2019), pp. 2340–2358.

[154] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning". In: *Nature* 521.7553 (2015), pp. 436–444.

[155] M. Lemou and L. Mieussens. "A new asymptotic preserving scheme based on micro-macro formulation for linear kinetic equations in the diffusion limit". In: *SIAM J. Sci. Comput.* 31.1 (2008), pp. 334–368.

[156] Q. Li and W. Sun. "Applications of kinetic tools to inverse transport problems". In: *Inverse Problems.* 36.3 (2020), p. 035011.

[157] S. Li, Z. Zhang, and H. Zhao. "A data-driven approach for multiscale elliptic PDEs with random coefficients based on intrinsic dimension reduction". In: *Multiscale Model. Simul.* 18.3 (2020), pp. 1242–1271.

[158] X.-A. Li, Z.-Q. J. Xu, and L. Zhang. "A multi-scale DNN algorithm for nonlinear elliptic equations with multiple scales". In: *Commun. Comput. Phys.* 28.5 (2020), pp. 1886–1906.

[159] Y. Li, X. Cheng, and J. Lu. "Butterfly-Net: Optimal function representation based on convolutional neural networks". In: *Commun. Comput. Phys.* 28.5 (2020), pp. 1838–1885.

[160] Y. E. Li and L. Demanet. "Full-waveform inversion with extrapolated low-frequency data". In: *Geophysics* 81.6 (2016), R339–R348.

[161] Z. Li et al. "Fourier neural operator for parametric partial differential equations". In: *International Conference on Learning Representations (ICLR)* (2021).

[162] P.-L. Lions. "On the existence of positive solutions of semilinear elliptic equations". In: *SIAM Rev.* 24.4 (1982), pp. 441–467.

[163] P.-L. Lions. "On the Schwarz alternating method II: Stochastic interpretation and orders properties". In: *Domain Decomposition Methods.* 1989, pp. 47–70.

[164] P.-L. Lions. "On the Schwarz alternating method. I". In: *First international symposium on domain decomposition methods for partial differential equations*. Vol. 1. Paris, France. 1988, p. 42.

[165] A. V. Little, M. Maggioni, and L. Rosasco. "Multiscale geometric methods for data sets I: Multiscale SVD, noise and curvature". In: *Appl. Comput. Harmon. Anal.* 43.3 (2017), pp. 504–567.

[166] X. Liu, E. Chung, and L. Zhang. "Iterated numerical homogenization for multiscale elliptic equations with monotone nonlinearity". In: *Multiscale Model. Simul.* 19.4 (2021), pp. 1601–1632.

[167] Z. Liu, W. Cai, and Z.-Q. J. Xu. "Multi-scale deep neural network (MscaleDNN) for solving Poisson-Boltzmann equation in complex domains". In: *Commun. Comput. Phys.* 28.5 (2020), pp. 1970–2001.

[168] Z. Long et al. "PDE-Net: Learning PDEs from data". In: *Proceedings of the 35th International Conference on Machine Learning*. PMLR. 2018, pp. 3208–3216.

[169] J. Lu and X. Yang. "Frozen Gaussian approximation for high frequency wave propagation". In: *Commun. Math. Sci.* 9 (2010), pp. 663–683.

[170] L. Lu et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature machine intelligence* 3.3 (2021), pp. 218–229.

[171] A. Målqvist and D. Peterseim. "Localization of elliptic multiscale problems". In: *Math. Comp.* 83.290 (2014), pp. 2583–2603.

[172] M. F. Modest. *Radiative Heat Transfer*. Elsevier Science, 2013.

[173] F. Monard. "Numerical implementation of geodesic X-ray transforms and their inversion". In: *SIAM J. Imaging Sci.* 7.2 (2014), pp. 1335–1357.

[174] F. Monard, P. Stefanov, and G. Uhlmann. "The geodesic ray transform on Riemannian surfaces with conjugate points". In: *Comm. Math. Phys.* 337.3 (2015), pp. 1491–1513.

[175] R. G. Mukhometov. "A problem of reconstructing a Riemannian metric". In: *Sib. Math. J.* 22 (3 1981).

[176] S. Nagayasu, G. Uhlmann, and J.-N. Wang. "Increasing stability in an inverse problem for the acoustic equation". In: *Inverse Problems* 29.2 (2013), p. 025012.

[177] F. Natterer. *The Mathematics of Computerized Tomography*. SIAM, 2001.

[178] R. G. Novikov. "Small angle scattering and X-ray transform in classical mechanics". In: *Ark. Mat.* 37.1 (1999), pp. 141–169.

[179] R. D. Oldham. "The Constitution of the Interior of the Earth, as Revealed by Earthquakes". In: *Quarterly Journal of the Geological Society* 62.1-4 (1906), pp. 456–475. ISSN: 0370-291X.

[180] M. Ovsjanikov et al. "Functional maps: a flexible representation of maps between shapes". In: *ACM Transactions on Graphics (ToG)* 31.4 (2012), pp. 1–11.

[181] H. Owhadi and L. Zhang. "Metric-based upscaling". In: *Comm. Pure Appl. Math.* 60.5 (2007), pp. 675–723.

[182] É. Pardoux. "BSDEs, weak convergence and homogenization of semilinear PDEs". In: *Nonlinear Analysis, Differential Equations and Control*. Ed. by F. H. Clarke, R. J. Stern, and G. Sabidussi. Springer Netherlands, 1999, pp. 503–549.

[183] A. Paszke et al. "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in Neural Information Processing Systems* 32 (2019), pp. 8026–8037.

[184] J. R. Pettit, A. E. Walker, and M. J. S. Lowe. "Improved Detection of Rough Defects for Ultrasonic Nondestructive Evaluation Inspections Based on Finite Element Modeling of Elastic Wave Scattering". In: *IEEE T. Ultrason. Ferr.* 62 (2015), pp. 1797–1808.

[185] R. G. Pratt. "Seismic waveform inversion in the frequency domain; Part 1: Theory and verification in a physical scale model". In: *Geophysics* 64.3 (1999), pp. 888–901.

[186] M. Presho and S. Ye. "Reduced-order multiscale modeling of nonlinear p-Laplacian flows in high-contrast media". In: *Comput. Geosci.* 19.4 (2015), pp. 921–932.

[187] J. Qian and L. Ying. "Fast Multiscale Gaussian Wavepacket Transforms and Multiscale Gaussian Beams for the Wave Equation". In: *Multiscale Model. Simul.* 8 (5 2010).

[188] M. Raissi, P. Perdikaris, and G. E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *J. Comput. Phys.* 378 (2019), pp. 686–707.

[189] N. Rawlinson, S. Pozgay, and S. Fishwick. "Seismic tomography: a window into deep Earth". English. In: *Phys. Earth Planet. Int.* 178.3-4 (Feb. 2010), pp. 101–135. ISSN: 0031-9201.

[190] D. Robert and M. Combescure. *Coherent States and Applications in Mathematical Physics*. Springer, 2012.

[191] H.-G. Roos, M. Stynes, and L. Tobiska. *Robust numerical methods for singularly perturbed differential equations: convection-diffusion-reaction and flow problems*. Vol. 24. Springer Science & Business Media, 2008.

[192] S. T. Roweis and L. K. Saul. "Nonlinear dimensionality reduction by locally linear embedding". In: *Science* 290.5500 (2000), pp. 2323–2326.

[193] L. Ryzhik, G. Papanicolaou, and J.B. Keller. "Transport equations for elastic and other waves in random media". In: *Wave motion* 24.4 (1996), pp. 327–370.

[194] S. Schochet. "The compressible Euler equations in a bounded domain: existence of solutions and the incompressible limit". In: *Comm. Math. Phys.* 104.1 (1986), pp. 49–75.

[195] H. Schomberg. "An improved approach to reconstructive ultrasound tomography". In: *J. of Phys. D: Appl. Phys.* 11.15 (Oct. 1978), pp. L181–L185.

[196] J. Sirignano and K. Spiliopoulos. "DGM: A deep learning algorithm for solving partial differential equations". In: *J. Comput. Phys.* 375 (2018), pp. 1339–1364.

[197] B. Smith, P. Bjorstad, and W. Gropp. *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*. Cambridge University Press, 2004.

[198] P. Stefanov et al. "Travel time tomography". In: *Acta Math. Sin. (Engl. Ser.)* 35.6 (2019), pp. 1085–1114.

[199] Z. Sun. "On continuous dependence for an inverse initial boundary value problem for the wave equation". In: *J. Math. Anal. Appl.* 150.1 (1990), pp. 188–204.

[200] T.L. Szabo. *Diagnostic ultrasound imaging: inside out*. Academic Press, 2004.

[201] N. M. Tanushev, J. Qian, and J. V. Ralston. "Mountain Waves and Gaussian Beams". In: *Multiscale Model. Simul.* 6.2 (2007), pp. 688–709.

[202] A. Tarantola. "Inversion of seismic reflection data in the acoustic approximation". In: *Geophysics* 49.8 (1984), pp. 1259–1266.

[203] A. Toselli and O. Widlund. *Domain decomposition methods-algorithms and theory*. Vol. 34. Springer Science & Business Media, 2006.

[204] F. Vico, L. Greengard, and M. Ferrando. "Fast convolution with free-space Green's functions". In: *J. Comput. Phys.* 323 (2016), pp. 191–203.

[205] J. Virieux and S. Operto. "An overview of full-waveform inversion in exploration Geophysics". In: *Geophysics* 74 (6 2009), WCC1–WCC26.

[206] J. Virieux et al. "6. An introduction to full waveform inversion". In: *Encyclopedia of Exploration Geophysics*. Society of Exploration Geophysicists, 2017, R1-1-R1–40.

[207] R. Viskanta and E. E. Anderson. "Heat transfer in semitransparent solids". In: *Advances in heat transfer*. Vol. 11. Elsevier, 1975, pp. 317–441.

[208] H. F. Walker and P. Ni. "Anderson acceleration for fixed-point iterations". In: *SIAM J. Numer. Anal.* 49.4 (2011), pp. 1715–1735.

[209] J.-N. Wang. "Stability estimates of an inverse problem for the stationary transport equation". In: *Ann. Inst. H. Poincaré Phys. Théor.* Vol. 70. 5. 1999, pp. 473–495.

[210] M. Wang et al. "Reduced-order deep learning for flow dynamics. The interplay between deep learning and model reduction". In: *J. Comput. Phys.* 401 (2020), p. 108939.

[211] A. D. Wheelon. *Electromagnetic Scintillation*. Vol. 1. Cambridge University Press, 2001.

[212] A. D. Wheelon. *Electromagnetic Scintillation*. Vol. 2. Cambridge University Press, 2003.

[213] K. Wu and D. Xiu. "Data-driven deep learning of partial differential equations in modal space". In: *J. Comput. Phys.* 408 (2020), p. 109307.

[214] D. Xiu. *Numerical methods for stochastic computations*. Princeton university press, 2010.

[215] D. Xiu and G. E. Karniadakis. "The Wiener–Askey polynomial chaos for stochastic differential equations". In: *SIAM J. Sci. Comput.* 24.2 (2002), pp. 619–644.

[216] Z. Xu, Y. Li, and X. Cheng. "Butterfly-Net2: Simplified Butterfly-Net and Fourier transform initialization". In: *Mathematical and Scientific Machine Learning*. PMLR. 2020, pp. 431–450.

[217] O. Yu. Imanuvilov and M. Yamamoto. "Global uniqueness and stability in determining coefficients of wave equations". In: *Comm. Partial Differential Equations* 26.7-8 (2001), pp. 1409–1425.

[218] Y. Zang et al. "Weak adversarial networks for high-dimensional partial differential equations". In: *J. Comput. Phys.* 411 (2020), p. 109409.

[219] Z. Zhang and H. Zha. "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment". In: *SIAM J. Sci. Comput* 26.1 (2004), pp. 313–338.

[220] C. Zhu et al. "Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization". In: *ACM Trans. Math. Softw.* 23.4 (Dec. 1997), pp. 550–560. ISSN: 0098-3500.