Introduction

The equations we consider in this book are primarily Fredholm integral equations of the second kind on bounded domains in the Euclidean space. These equations are used as mathematical models for a multitude of physical problems and cover many important applications, such as radiosity equations for realistic image synthesis [18, 85, 244] and especially boundary integral equations [12, 177, 203], which themselves occur as reformulations of other problems, typically originating as partial differential equations. In practice, Fredholm integral equations are solved numerically using piecewise polynomial collocation or Galerkin methods, and when the order of the coefficient matrix (which is typically full) is large, the computational cost of generating the matrix as well as solving the corresponding linear system is large. Therefore, to enhance the range of applicability of the Fredholm equation methodology, it is critical to provide alternate algorithms which are fast, efficient and accurate. This book is concerned with this challenge: designing fast multiscale methods for the numerical solution of Fredholm integral equations.

The development and use of multiscale methods for solving integral equations is a subject of recent intense study. The history of fast multiscale solutions of integral equations began with the introduction of multiscale Galerkin (Petrov–Galerkin) methods for solving integral equations, as presented in [28, 64, 68, 88, 94, 95, 202, 260, 261] and the references cited therein. Most noteworthy is the discovery in [28] that the representation of a singular integral operator by compactly supported orthonormal wavelets produces numerically sparse matrices. In other words, most of their entries are so small in absolute value that, to some degree of precision, they can be neglected without affecting the overall accuracy of the approximation. Later, the papers [94, 95] studied Petrov–Galerkin methods using periodic multiscale bases constructed from refinement equations for periodic elliptic pseudodifferential

equations, and in this restricted environment, stability, convergence and matrix compression were investigated. For a first-kind boundary integral equation, a truncation strategy for the Galerkin method using spline-based multiscale basis functions of low degree was proposed in [260]. Also, in [261], for elliptic pseudodifferential equations of order zero on a three-dimensional manifold, a Galerkin method using discontinuous piecewise linear multiscale basis functions on triangles was studied.

In another direction, a general construction of multidimensional discontinuous orthogonal and bi-orthogonal wavelets on invariant sets was presented in [200, 201]. Invariant sets include, among others, the important cases of simplices and cubes, and in the two-dimensional case L-shaped domains. A similar recursive structure was explored in [65] for multiscale function representation and approximation constructed by interpolation on invariant sets. In this regard, an essential advantage of this approach is the existence of efficient schemes for generating recursively multilevel partitions of invariant sets and their associated multiscale functions. All of these methods even extend to domains which are a finite union of invariant sets, thereby significantly expanding the range of their applicability. Therefore, the constructions given in [65, 200, 201] led to a wide variety of multiscale basis functions which, on the one hand, have desirable simple recursive structure and, on the other hand, can be used in diverse areas in which the Fredholm methodology is applied.

Subsequently, the papers [64, 68, 202] developed multiscale piecewise polynomial Galerkin, Petrov–Galerkin and discrete multiscale Petrov–Galerkin methods. An important advantage of multiscale piecewise polynomials is that their closed-form expressions are very convenient for computation. Moreover, they can easily be related to standard bases used in the conventional numerical method, thereby providing an advantage for theoretical analysis as well. Among conventional numerical methods for solving integral equations, the collocation method has received the most favorable attention in the engineering community due to its lower computational cost in generating the coefficient matrix of the corresponding discrete equations. In comparison, the implementation of the Galerkin method requires much more computational effort for the evaluation of integrals (see, for example, [19, 77] for a discussion of this point). Motivated by this issue, [69] proposed and analyzed a fast collocation algorithm for solving general multidimensional integral equations. Moreover, a matrix truncation strategy was introduced there by making a careful choice of basis functions and collocation functionals, the end result being fast, multiscale algorithms for solving the integral equations.

The development of stable, efficient and fast numerical algorithms for solving operator equations, including differential equations and integral equations,

is a main focus of research in numerical analysis and scientific computation, since such algorithms are particularly important for large-scale computation. We review the three main steps in solving an operator equation. The first is at the level of approximation theory. Here, we must choose appropriate subspaces and suitable bases for them. The second step is to discretize the operator equations using these bases and to analyze convergence properties of the approximate solutions. The end result of this step of processing is a discrete linear system and its construction is considered as a main task for the numerical solution of operator equations. The third step employs methods of numerical linear algebra to design an efficient solver for the discrete linear system. The ultimate goal is, of course, to solve the discrete linear system efficiently and obtain an accurate approximate solution to the original operator equation. Theoretical considerations and practical implementations in the numerical solution of operator equations show that these three steps of processing are closely related. Therefore, designing efficient algorithms for the discrete linear system should take into consideration the choice of subspaces and their bases, the methodologies of discretization of the operator equations and the specific characteristics and advantages of the numerical solvers used to solve the resulting discrete linear system. In this book we describe how these three steps are integrated in a multiscale environment and thereby achieve our goal of providing a wide selection of fast and accurate algorithms for the second-kind integral equations. We also describe work in progress addressing related issues of eigenvalue and eigenfunction computation as well as the solution of Fredholm equations of the first kind.

This book is organized into 12 chapters plus an appendix. Chapter 1 is devoted to a review of the Fredholm approach to solving an integral equation of the second kind. In Chapter 2 we introduce essential concepts from Fredholm integral equations of the second kind and describe a general setup of projection methods for solving operator equations which will be used in later chapters. The purpose of Chapter 3 is to describe conventional numerical methods for solving Fredholm integral equations of the second kind, including the degenerate kernel method, the quadrature method, the Galerkin method, the Petrov–Galerkin method and the collocation method. In Chapter 4, a general construction of multiscale bases of piecewise polynomial spaces, including multiscale orthogonal and interpolating bases, is presented. Chapters 5, 6 and 7 use the material from Chapter 4 to construct multiscale Galerkin, Petrov–Galerkin and collocation methods. We study the discretization schemes resulting from these methods, propose truncation strategies for building fast and accurate algorithms, and give a complete analysis for the order of convergence, computational complexity, stability and condition numbers for the

truncated schemes. In Chapter 8, two types of quadrature rule for the numerical integration required to generate the coefficient matrix are introduced and error control strategies are designed so that the quadrature errors will neither ruin the overall convergence order nor increase the overall computational complexity of the original multiscale methods. The goal of Chapter 9 is to investigate fast solvers for the discrete linear systems resulting from multiscale methods. We introduce multilevel augmentation methods and multilevel iteration methods based on direct sum decompositions of the range and domain of the operator equation. In Chapters 10, 11 and 12, the fast algorithms are applied to solving nonlinear integral equations of the second kind, ill-posed integral equations of the first kind and eigen-problems of compact integral operators, respectively. We summarize in the Appendix some of the standard concepts and results from functional analysis in a form which is used throughout the book. The appendix provides the reader with a convenient source of the background material needed to follow the ideas and arguments presented in other chapters of this book.

Most of the material in this book can only be found in research papers. This is the first time that it has been assembled into a book. Although this book is pronouncedly a research monograph, selected material from the initial chapters can be used in a semester course on numerical methods for integral equations which presents the multiscale point of view.

1

A review of the Fredholm approach

In this chapter we pay homage to Ivar Fredholm (April 7, 1866–August 17, 1927) and review his approach to solving an integral equation of the second kind. The methods employed in this chapter are classical and differ from the approach taken in the rest of the book. We include it here because those readers inexperienced in integral equations should be familiar with these important ideas. The basic tools of matrix theory and some complex analysis are needed, and we shall provide a reasonably self-contained discussion of the required material.

1.1 Introduction

We start by introducing the notation that will be used throughout this book. Let $\mathbb{C}$, $\mathbb{R}$, $\mathbb{Z}$ and $\mathbb{N}$ denote, respectively, the set of complex numbers, the set of real numbers, the set of integers and the set of positive integers. We also let $\mathbb{N}_0 := \{0\} \cup \mathbb{N}$. For the purpose of enumerating a nonempty finite set of objects we use the sets $\mathbb{N}_d := \{1, 2, \ldots, d\}$ and $\mathbb{Z}_d := \{0, 1, \ldots, d-1\}$, both of which consist of d distinct integers. For $d \in \mathbb{N}$, let $\mathbb{R}^d$ denote the d-dimensional Euclidean space and Ω a subset of $\mathbb{R}^d$. By $C(\Omega)$, we mean the linear space of all continuous real-valued functions defined on Ω. We usually denote matrices or vectors over $\mathbb{R}$ in boldface, for example, $\mathbf{A} := [A_{ij} : i, j \in \mathbb{N}_d] \in \mathbb{R}^{d\times d}$ and $\mathbf{u} := [u_j : j \in \mathbb{N}_d] \in \mathbb{R}^d$. When the vector has all integer coordinates, that is, $\mathbf{u} \in \mathbb{Z}^d$, we sometimes call it a *lattice vector*. Moreover, we usually denote integral operators by calligraphic letters. Especially, the *integral operator* with a *kernel* K will be denoted by $\mathcal{K}$, that is, for the kernel K defined on $\Omega \times \Omega$ and the function u defined on Ω, we define

$$(\mathcal{K}u)(s) := \int_\Omega K(s,t)u(t)\mathrm{d}t, \quad s \in \Omega.$$

The most direct approach to solving a second-kind integral equation merely replaces integrals by sums and thereby obtains a linear system of equations whose solution approximates the solution of the original equation. The study of the resulting linear system of equations leads naturally to the important notion of the Fredholm function and determinant, which remain a central tool in the theory of second-kind integral equations, see for example [183, 253]. We consider this direct approach when we are given a *continuous* kernel $K \in C(\Omega \times \Omega)$ on a *compact* subset Ω of $\mathbb{R}^d$ with positive Borel measure, a continuous function $f \in C(\Omega)$ and a nonzero complex number $\lambda \in \mathbb{C}$. The task is to find a function $u \in C(\Omega)$ such that, for $s \in \Omega$,

$$u(s) - \lambda \int_\Omega K(s,t)u(t)\mathrm{d}t = f(s). \tag{1.1}$$

To this end, for each positive $h > 0$, we partition Ω into nonempty compact subsets $\Omega_i, i \in \mathbb{N}_n$

$$\Omega = \bigcup_{i \in \mathbb{N}_n} \Omega_i$$

such that different subsets have no overlapping interior and

$$\operatorname{diam} \Omega_i := \max\{|x - y| : x, y \in \Omega_i\} \le h,$$

where $|x|$ is the ℓ^2-norm of the vector $x \in \mathbb{R}^d$. This partition can be constructed by first putting a large "box" around the set Ω and then decomposing this box into cubes, each of which has diameter less than or equal to h. The sets Ω_i are then formed by intersecting the set Ω with the cubes, where we discard sets of zero Borel measure. Therefore, the partition of Ω constructed in this manner is done a.e. Next, we choose *any* finite set of points $T := \{t_i : i \in \mathbb{N}_n\}$ such that, for any $i \in \mathbb{N}_n$, we have that $t_i \in \Omega_i$. With these points we now replace our integral equation (1.1) with a linear system of equations. Specifically, we choose the number $\rho := -\lambda$ and the $n \times n$ matrix $\mathbf{A}$ defined by

$$\mathbf{A} := [\operatorname{vol}(\Omega_j)K(t_i, t_j) : i, j \in \mathbb{N}_n],$$

where $\operatorname{vol}(\Omega_j)$ denotes the volume of the set Ω_j, and replace (1.1) with the system of linear equations

$$(\mathbf{I} + \rho\mathbf{A})\mathbf{u} = \mathbf{f}. \tag{1.2}$$

Here, $\mathbf{f}$ is the vector obtained by evaluating the function f on the set T.

Of course, the point of view we take here is that the vector $\mathbf{u} \in \mathbb{R}^n$ which solves equation (1.2) is an approximation to the function u on the set T. Therefore, the problem of determining the function u is replaced by the simpler one of numerically solving for the vector $\mathbf{u}$ when h is small. Certainly,

an important role is played by the determinant of the coefficient matrix of the linear system (1.2). Its properties, especially as $h \to 0^+$, will be our main concern for a significant part of this chapter. We start by studying the determinant of the coefficient matrix of the linear system (1.2) and then derive a formula for the entries of the inverse of the matrix $\mathbf{I} + \rho\mathbf{A}$ in terms of the matrix $\mathbf{A}$.

1.2 Second-kind matrix Fredholm equations

We define the minor of an $n \times n$ matrix. If $\mathbf{A} = [A_{ij} : i,j \in \mathbb{N}_n]$ is an $n \times n$ matrix, q is a non-negative integer in $\mathbb{Z}_{n+1}$, $\mathbf{i} := [i_l : l \in \mathbb{N}_q]$, $\mathbf{j} := [j_l : l \in \mathbb{N}_q]$ are lattice vectors in $\mathbb{N}_n^q$ we define the corresponding *minor* by

$$A[\mathbf{i},\mathbf{j}] := \det[A_{i_r,j_s} : r,s \in \mathbb{N}_q].$$

Sometimes, the more elaborate notation

$$A\begin{pmatrix} i_1, & i_2, & \ldots, & i_q \\ j_1, & j_2, & \ldots, & j_q \end{pmatrix} \tag{1.3}$$

is used for $A[\mathbf{i},\mathbf{j}]$. When $\mathbf{i} = \mathbf{j}$, that is, for a *principal minor* of $\mathbf{A}$, we use the simplified notation $A[\mathbf{i}]$ in place of $A[\mathbf{i},\mathbf{i}]$. For a positive integer $q \in \mathbb{N}_n$, we set

$$r_q(\mathbf{A}) := \frac{1}{q!} \sum_{\mathbf{i} \in \mathbb{N}_n^q} A[\mathbf{i}] \tag{1.4}$$

and also choose $r_0(\mathbf{A}) := 1$.

Lemma 1.1 *If $\mathbf{A}$ is an $n \times n$ matrix and $\rho \in \mathbb{C}$, then*

$$\det(\mathbf{I} + \rho\mathbf{A}) = \sum_{q \in \mathbb{Z}_{n+1}} r_q(\mathbf{A})\rho^q. \tag{1.5}$$

Before proving this lemma, we make two remarks.

Remark 1.2 *Using the extended notation for a minor as indicated in (1.3), we see that equation (1.4) is equivalent to the formula*

$$r_q(\mathbf{A}) = \frac{1}{q!} \sum_{[i_l : l \in \mathbb{N}_q] \in \mathbb{N}_n^q} A\begin{pmatrix} i_1, & \ldots, & i_q \\ i_1, & \ldots, & i_q \end{pmatrix}. \tag{1.6}$$

Certainly, if any two components of the vector $\mathbf{i} = [i_l : l \in \mathbb{N}_q]$ are equal, then the corresponding minor has a repeated row (and column) and so is zero. These terms may be neglected. Moreover, any permutation of the components of the vector $\mathbf{i}$ affects both a row and a column exchange of the determinant

appearing in (1.6), and so does not affect the value of the determinant. Since there are q! such permutations, we get that

$$r_q(\mathbf{A}) = \sum_{1 \le i_1 < i_2 < \cdots < i_q \le n} A \begin{pmatrix} i_1, & \ldots, & i_q \\ i_1, & \ldots, & i_q \end{pmatrix}. \tag{1.7}$$

Remark 1.3 *If the characteristic values of the matrix* $\mathbf{A}$ *are denoted by* $\{\lambda_j : j \in \mathbb{N}_n\}$*, then*

$$r_q(\mathbf{A}) = \sum_{1 \le i_1 < i_2 < \cdots < i_q \le n} \lambda_{i_1} \lambda_{i_2} \cdots \lambda_{i_q}. \tag{1.8}$$

The right-hand side of this equation is an elementary symmetric function of the eigenvalues of $\mathbf{A}$*, which is invariant under a similarity transformation of the matrix* $\mathbf{A}$*. This fact will be the basis of the proof of Lemma 1.1 presented below.*

We next present the proof of Lemma 1.1.

Proof By Schur's upper-triangular factorization theorem (see, for example, [142], p. 79), the matrix $\mathbf{A}$ can be factored in the form

$$\mathbf{A} = \mathbf{P}^{-1}\mathbf{T}\mathbf{P}, \tag{1.9}$$

where $\mathbf{P}$ is an orthogonal matrix and $\mathbf{T}$ is an upper-triangular matrix whose diagonal entries are the eigenvalues of $\mathbf{A}$ (chosen in any prescribed order). For an upper-triangular matrix $\mathbf{T}$ we observe from (1.8) that

$$\begin{aligned} \det(\mathbf{I} + \rho\mathbf{T}) &= \prod_{j \in \mathbb{N}_n} (1 + \rho\lambda_j) \\ &= \sum_{q \in Z_{n+1}} \rho^q \sum_{1 \le j_1 < j_2 < \cdots < j_q \le n} \lambda_{j_1} \lambda_{j_2} \cdots \lambda_{j_q} \\ &= \sum_{q \in Z_{n+1}} r_q(\mathbf{T}) \rho^q, \end{aligned}$$

thereby verifying that equation (1.5) is valid at least for an upper-triangular matrix. For a general matrix we use the reduction to the upper-triangular case by an orthogonal similarity (1.9). Since all determinants in equations (1.5) and (1.6) are unchanged under an orthogonal similarity, this comment proves the general case. □

We now add a more difficult computation, which provides an expansion of the elements of the matrix $(\mathbf{I} + \rho\mathbf{A})^{-1}$ as a rational function of ρ. To this end,

we introduce some needed constants. We define for any $k, l \in \mathbb{N}_n$ and $q \in \mathbb{Z}_{n+1}$ the constants

$$u_{lkq} := \frac{1}{q!} \sum_{\mathbf{i} \in \mathbb{N}_n^q} A \begin{bmatrix} l, & \mathbf{i} \\ k, & \mathbf{i} \end{bmatrix}. \tag{1.10}$$

In (1.10) there are (not necessarily principal) minors of order $q+1$. Moreover, when $q = n-1$ and $k, l \in \mathbb{N}_n$ with $k \neq l$ we have that $u_{lkq} = 0$, since $\mathbb{N}_n$ has only n distinct elements. Also, for the same reason, $u_{lkn} = 0$ for all $k, l \in \mathbb{N}_n$.

We shall relate these constants. But first we introduce for $k, l \in \mathbb{N}_n$ polynomials U_{kl} defined at ρ to be

$$U_{lk}(\rho) := \sum_{q \in \mathbb{Z}_{n+1}} u_{lkq} \rho^q.$$

Now, to relate all of these quantities, we start with the minor

$$A \begin{pmatrix} l, & i_1, & \dots, & i_q \\ k, & i_1, & \dots, & i_q \end{pmatrix} \tag{1.11}$$

where $l, k \in \mathbb{N}_n$ and expand it, by the *Laplace expansion* by minors, across its first row to obtain the formula

$$A \begin{pmatrix} l, & i_1, & \dots, & i_q \\ k, & i_1, & \dots, & i_q \end{pmatrix} = A_{lk} A \begin{pmatrix} i_1, & \dots, & i_q \\ i_1, & \dots, & i_q \end{pmatrix} + \sum_{m \in \mathbb{N}_q} (-1)^m A_{l i_m} A \begin{pmatrix} i_1, i_2, \dots, i_{m+1}, \dots, i_q \\ k, i_1, \dots, \widehat{i_m}, \cdots, i_q \end{pmatrix}. \tag{1.12}$$

The symbol $\widehat{i_m}$ appearing in the minor above is to be interpreted to mean that the i_mth column of A is *deleted* in that minor. We now sum both sides of this formula over all integers $i_1, i_2, \dots, i_q$ in $\mathbb{N}_n$, and interchange the summations of the second term on the right-hand side of the resulting equation to yield the formula

$$\sum_{[i_p : p \in \mathbb{N}_q] \in \mathbb{N}_n^q} A \begin{pmatrix} l, & i_1, & \dots, & i_q \\ k, & i_1, & \dots, & i_q \end{pmatrix} = \sum_{[i_p : p \in \mathbb{N}_q] \in \mathbb{N}_n^q} A_{lk} A \begin{pmatrix} i_1, & \dots, & i_q \\ i_1, & \dots, & i_q \end{pmatrix} + \sum_{m \in \mathbb{N}_q} \sum_{[i_p : p \in \mathbb{N}_q] \in \mathbb{N}_n^q} (-1)^m A_{l i_m} A \begin{pmatrix} i_1, & i_2, & \dots, & i_{m+1}, & \dots, & i_q \\ k, & i_1, & \dots, & \widehat{i_m}, & \cdots, & i_q \end{pmatrix}. \tag{1.13}$$

The first term on the right-hand side of equation (1.13) is clearly $A_{lk} q! r_q(\mathbf{A})$, which follows from our definition (1.6). The value for the second term requires more explanation.

For the second term we point out that there are two sums over $i_m \in \mathbb{N}_n$. The outer sum already appears in the right-hand side of equation (1.13) and the inner one appears in the sum over all indices $i_1, i_2, \ldots, i_q$ in $\mathbb{N}_n$. In the inner sum we first fix i_m and then sum over all the other indices $i_1, \ldots, i_{m-1}, i_{m+1}, \ldots, i_q \in \mathbb{N}_n$. This leads us to the expression

$$\sum_{[i_1,\ldots,i_{m-1},i_{m+1},\ldots,i_q]\in\mathbb{N}_n^{q-1}} (-1)^m A\begin{pmatrix} i_1, & \ldots, & i_q \\ k, i_1, \ldots, i_{m-1}, & i_{m+1}, \ldots, & i_q \end{pmatrix}.$$

We now locate the i_mth row in the minor of A and move it forward to the first row. This requires $m-1$ row exchanges and gives us the expression

$$-\sum_{[i_1,\ldots,i_{m-1},i_{m+1},\ldots,i_q]\in\mathbb{N}_n^{q-1}} A\begin{pmatrix} i_m, i_1, \ldots, i_{m-1}, & i_{m+1}, \ldots, & i_q \\ k, i_1, \ldots, i_{m-1}, & i_{m+1}, \ldots, & i_q \end{pmatrix}.$$

Next, we multiply this expression by A_{li_m} and compute the (first) sum of it over $i_m \in \mathbb{N}_n$. This yields the quantity

$$-\sum_{r\in\mathbb{N}_n} A_{lr} \sum_{[i_p:p\in\mathbb{N}_{q-1}]\in\mathbb{N}_n^{q-1}} A\begin{pmatrix} r, & i_1, & \ldots, & i_{q-1} \\ k, & i_1, & \ldots, & i_{q-1} \end{pmatrix}.$$

But this quantity is *independent* of i_m and so appears q times in the other (second) sum over i_m. So, in summary, we get the equation

$$\sum_{[i_p:p\in\mathbb{N}_q]\in\mathbb{N}_n^q} A\begin{pmatrix} l, & i_1, & \ldots, & i_q \\ k, & i_1, & \ldots, & i_q \end{pmatrix} = A_{lk} \sum_{[i_p:p\in\mathbb{N}_q]\in\mathbb{N}_n^q} A\begin{pmatrix} i_1, & \ldots, & i_q \\ i_1, & \ldots, & i_q \end{pmatrix} - q\sum_{r\in\mathbb{N}_n} A_{lr} \sum_{[i_p:p\in\mathbb{N}_{q-1}]\in\mathbb{N}_n^{q-1}} A\begin{pmatrix} r, & i_1, & \ldots, & i_{q-1} \\ k, & i_1, & \ldots, & i_{q-1} \end{pmatrix},$$

which is equivalent to the formula

$$\sum_{\mathbf{i}\in\mathbb{N}_n^q} A\begin{bmatrix} l, & \mathbf{i} \\ k, & \mathbf{i} \end{bmatrix} = A_{lk} \sum_{\mathbf{i}\in\mathbb{N}_n^q} A[\mathbf{i}] - q\sum_{r\in\mathbb{N}_n} A_{lr} \sum_{\mathbf{j}\in\mathbb{N}_n^{q-1}} A\begin{bmatrix} r, & \mathbf{j} \\ k, & \mathbf{j} \end{bmatrix}.$$

When $q = 0$ the second term on the right is zero while the first sum on the right is set to one. Likewise, the expression on the left is set to A_{lk} and so this formula is still true when $q = 0$. We now multiply both sides by $\frac{\rho^q}{q!}$ and sum over $q \in \mathbb{Z}_{n+1}$. Upon simplification we conclude, for $l, k \in \mathbb{N}_n$, that

$$U_{lk}(\rho) = A_{lk} \det(\mathbf{I} + \rho\mathbf{A}) - \rho \sum_{r\in\mathbb{N}_n} A_{lr} U_{rk}(\rho). \tag{1.14}$$