Towards a high-performance method for the biharmonic Dirichlet problem

Jingrui Zhang

April 2004

Abstract

The biharmonic Dirichlet problem appears in many applications. Therefore, there is a lot of interest in developing high-performance methods for its solution. Two important components of performance are accuracy and efficiency. In this research, we combine a sixth order discretization method and a Fast Fourier Transform (FFT)-based solver for the solution of the biharmonic Dirichlet problem.

The discretization of the problem uses the collocation methodology and quartic splines. Quartic Spline Collocation (Quartic SC) methods that produce sixth order approximations have been recently developed for the solution of fourth order Boundary Value Problems (BVPs). In this paper, we apply an adjustment to the quartic spline basis functions so that the Quartic SC matrix has more favorable properties. Particular attention is given to the one-dimensional harmonic problem. We study the properties of two Quartic SC matrices, more specifically, one that corresponds to Dirichlet and Neumann conditions (the standard harmonic problem) and another that corresponds to Dirichlet and second derivative conditions. The latter is solvable by FFTs and used as preconditioner for the first. We develop formulae for the eigenvalues and eigenvectors of the preconditioned matrix. We show that three iterations suffice to obtain convergence for the GMRES preconditioned method. Numerical experiments verify the effectiveness of the solver and preconditioner. We also discuss the application of the preconditioned GMRES method to the two-dimensional biharmonic Dirichlet problem.

Contents

Chapter 1

Introduction

Fourth-order boundary value problems (BVPs) are widely used to model physical phenomena [27]. The biharmonic Dirichlet problem, a fourth-order two-dimensional partial differential equation (PDE) problem that arises in some applications, e.g. flow problems in fluid mechanics and elasticity problems in solid mechanics [12], is described by

$$\Delta^2 u = f \quad \text{in} \quad \Omega, \tag{1.1}$$

$$u = g_1 \quad \text{on} \quad \partial\Omega, \tag{1.2}$$

$$\frac{\partial u}{\partial n} = g_2 \quad \text{on} \quad \partial\Omega, \tag{1.3}$$

where Δ denotes the Laplacian, $\Omega = (a,b) \times (a,b)$, $\partial\Omega$ is the boundary of Ω, $\partial/\partial n$ is the outer normal derivative on $\partial\Omega$, and f, g_1 and g_2 are given functions. We remind the reader that, for a two-dimensional problem, $\Delta u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$ and $\Delta^2 u \equiv \frac{\partial^4 u}{\partial x^4} + 2\frac{\partial^4 u}{\partial x^2 y^2} + \frac{\partial^4 u}{\partial y^4}$.

Various numerical methods have been developed for the BVP (1.1)-(1.3). These methods can be divided essentially into two classes: (a) methods that transform the problem into an equivalent 2×2 system of coupled second order PDEs, and (b) methods that treat the biharmonic PDE as one entity and apply discretization methods for the fourth order derivatives.

For the coupled approach, an auxiliary unknown $v \equiv \Delta u$ is introduced, and the fourth order PDE (1.1) is split into two coupled Poisson PDEs,

$$\Delta u = v, \tag{1.4}$$

$$\Delta v = f. \tag{1.5}$$

Note that the boundary conditions (1.2)-(1.3) refer only to u, therefore the boundary conditions of the second unknown, v, are undefined, and the resulting 2×2 PDE system is not uniquely solvable. This difficulty is overcome by techniques for approximating the boundary conditions for v on two opposite sides of the domain. Various discretization techniques can be used for solving the 2×2 PDE system, including finite difference and finite element methods.

Several authors (please see [10], [11], [18] and other references for detailed information) use standard five-point finite differences, resulting second order approximations. The arising linear systems can be solved by fast Poisson solvers, however, the approximation of boundary conditions for v can involve substantial computational cost.

Other researchers have developed solution techniques based on finite element discretization of the 2×2 Poisson system. The techniques developed in

([7], [6], [12] and [20]) are based on finite element discretization and iterative solution techniques. In [13], a Galerkin discretization methodology based on Hermite piecewise bicubics is used. The approximate boundary conditions for v are obtained by solving a Schur complement system by a preconditioned conjugate gradient (PCG) method. It is conjectured and verified by experiments that the number of iterations is almost independent of the problem size. The solution of the Poisson problems is obtained by fast Fourier transforms (FFTs). The discretization error is of fourth order and the complexity of the resulting solver is $O(N^2 \log_2 N)$ on an $N \times N$ uniform partition.

Relatively recently, some researchers have developed and analyzed orthogonal bicubic spline collocation methods for the biharmonic problem ([16], [23], and [9]). The global errors of these methods are of fourth order. Moreover, in [2], a solution technique similar to that in [13] and based on FFTs and PCG solution of a Schur complement system is presented for the Hermite piecewise bicubic collocation equations arising from the pair of coupled second order problems. The complexity of the resulting solver is $O(N^2 \log_2 N)$.

For the second approach, that treats the biharmonic as one fourth order PDE, several finite difference discretization and associated linear solvers have been proposed. Some methods are based on a thirteen-point stencil discretization, leading to second order approximations. Some algorithms for solving the resulting linear system have been proposed in [4] and [3]. Bjørstad's method [3] uses an auxiliary problem for the solution of (1.1)-(1.3). More specifically, it considers the problem that consists of the biharmonic equation (1.1), the boundary conditions (1.2) and the boundary conditions

$$\frac{\partial u}{\partial y} = g_{21} \quad \text{on} \quad \partial\Omega_1, \quad \frac{\partial^2 u}{\partial x^2} = g_{22} \quad \text{on} \quad \partial\Omega_2, \tag{1.6}$$

where $\partial\Omega_1$ are the two horizontal boundary lines of Ω and $\partial\Omega_2$ the two vertical ones. This problem can be solved by FFTs at $O(N^2 \log_2 N)$ cost or by multigrid solvers at $O(N^2)$ cost. To obtain the solution of the biharmonic Dirichlet problem (1.1)-(1.3), the solution of (1.1)-(1.2)-(1.6) is combined with the solution of two one-dimensional problems along the horizontal boundary lines solved by a PCG method with number of iterations independent of N. The resulting method in [3] is of optimal complexity, that is, on an $N \times N$ partition, the complexity is $O(N^2)$.

Certain second and fourth order finite difference approximations for the biharmonic Dirichlet problem on a 9-point compact cell are given in [1], [14], and [22]. Moreover, Altas et al. also proposed several multigrid techniques which exhibit grid-independent convergence and solve the biharmonic equation in time proportional to the number of unknowns [1].

Most recently, in [27], Zhu proposed quartic spline collocation (Quartic SC) methods for the solution of fourth order two-point BVPs that produce sixth order approximations. These methods can be extended to two-dimensional problems. The purpose of this paper is to formulate the Bi-Quartic spline collocation equations and present a fast solver based on the preconditioned GMRES method and fast Fourier transforms for the linear systems arising from the Bi-Quartic SC method. In order to study the properties of the Bi-Quartic SC equations, we draw attention to the one-dimensional harmonic problem, which is defined by

$$\mathbf{L}u \equiv u^{(4)} + u^{(2)} + u = g \qquad \text{on} \quad I = (a, b), \tag{1.7}$$

subject to boundary conditions

$$u = g_1 \quad \text{on} \quad \partial\Omega, \tag{1.8}$$

$$u_n = g_2 \quad \text{on} \quad \partial\Omega. \tag{1.9}$$

An outline of this paper is as follows. Quartic SC concepts and the discretization results of the harmonic problems are given in Chapter 2. In Chapter 3, a variety of GMRES algorithms are introduced and theorems about the convergence analysis are presented. The properties of the preconditioned matrix are discussed in Chapter 4 along with the convergence analysis of the preconditioned GMRES method for the problem (1.7)-(1.9). In Chapter 5, we derive the Bi-Quartic SC matrices for the biharmonic problem. The numerical results are presented in Chapter 6. We conclude our work and mention some future work in Chapter 7.

Chapter 2

Quartic Spline Collocation Method for Harmonic Problem

In this chapter, we adopt the quartic spline collocation (Quartic SC) method to solve the fourth order derivative equation (1.7) with different boundary conditions.

2.1 Quartic Splines

We first introduce quartic splines and some notation. Let $\Delta \equiv \{a = x_0 < x_1 < \cdots < x_N = b\}$ be the uniform partition of the interval $[a, b]$ with the mesh size $h = \frac{1}{N}$. Please note that, in the following, without explicit mention, Δ represents a partition of some interval. Throughout we denote by $\tau_i \equiv \frac{x_{i-1}+x_i}{2}, i = 1, \cdots, N$, the midpoints of Δ. For convenience, we extend the notation so that $\tau_0 \equiv x_0, \tau_{N+1} \equiv x_N$.

We denote by $S_{4,\Delta}$ the quartic spline space with respect to Δ, that is, the approximating space of quartic piecewise polynomials with respect to Δ and with continuity $C^3(I)$. Noticing that $S_{4,\Delta}$ has $N+4$ dimensions, in order to represent any quartic spline $S \in S_{4,\Delta}$, we choose a set of quartic spline basis functions, the quartic B-splines, $\Phi_i(x), i = -1, \cdots, N+2$. These basis functions are defined based on a model quartic spline function:

$$\Phi(x) = \frac{1}{24}\begin{cases} x^4 & \text{for } 0 \le x \le 1 \\ x^4 - 5(x-1)^4 & \text{for} 1 \le x \le 2 \\ x^4 - 5(x-1)^4 + 10(x-2)^4 & \text{for} 2 \le x \le 3 \\ x^4 - 5(x-1)^4 + 10(x-2)^4 - 10(x-3)^4 & \text{for} 3 \le x \le 4 \\ x^4 - 5(x-1)^4 + 10(x-2)^4 - 10(x-3)^4 + 5(x-4)^4 & \text{for} 4 \le x \le 5 \\ 0 & \text{elsewhere.} \end{cases} \tag{2.1}$$

Then the set

$$\{\Phi_i(x) = \Phi(\frac{x-a}{h} - i + 3), i = -1, \cdots, N+2\} \tag{2.2}$$

forms the B-spline basis functions of $S_{4,\Delta}$. Any quartic spline $S \in S_{4,\Delta}$ can be written as $S(x) = \sum_{i=-1}^{N+2} c_i \Phi_i(x)$

When we apply the Quartic SC method to BVPs, we adjust the standard basis functions so that the approximate space satisfies exactly the boundary conditions and the resulting Quartic SC equations are of a smaller size.

2.2 Harmonic Problem with Dirichlet and Second Derivative Boundary Conditions

Consider the problem

$$\mathbf{L}u \equiv u^{(4)} + u^{(2)} + u = g \qquad \text{on} \quad I = (a, b), \tag{2.3}$$

subject to boundary conditions

$$u(a) = u(b) = 0, \tag{2.4}$$

$$u''(a) = u''(b) = 0. \tag{2.5}$$

A detailed discussion about using the quartic spline collocation method to solve the problem (2.3) can be found in [27]. Here we briefly describe it for our next discussion.

To satisfy the boundary conditions (2.4)-(2.5) by construction, the standard quartic spline basis functions $\Phi_i, i = -1, \cdots, N+2$ are adjusted as follows:

$$\begin{aligned} \approx{\Phi}_1 &= \Phi_1 - \Phi_0, & \approx{\Phi}_2 &= \Phi_2 - \Phi_{-1}, \\ \approx{\Phi}_i &= \Phi_i, & i &= 3, \cdots, N, \\ \approx{\Phi}_{N-1} &= \Phi_{N-1} - \Phi_{N+2}, & \approx{\Phi}_N &= \Phi_N - \Phi_{N+1}. \end{aligned} \tag{2.6}$$

The application of the quartic spline collocation method with the basis functions $\overset{\approx}{\Phi}_i, i = 1, \cdots, N$ to the differential equation problem (2.3) generates the system of discrete equations:

$$K\bar{\theta} = (Q_4 + Q_2 + Q_0)\bar{\theta} = \bar{g}, \tag{2.7}$$

where $\bar{g} = (g(\tau_1), g(\tau_2), \cdots, g(\tau_N))^T$, and Q_i, $i = 0, 2, 4$, arise from the terms $u^{(i)}, i = 0, 2, 4$, respectively. The forms of $Q_i, i = 0, 2, 4$, are as follows:

$$Q_4 = \frac{1}{h^4} \begin{bmatrix} 10 & -5 & 1 & & & & \\ -5 & 6 & -4 & 1 & & & \\ 1 & -4 & 6 & -4 & 1 & & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \\ & & 1 & -4 & 6 & -4 & 1 \\ & & & 1 & -4 & 6 & -5 \\ & & & & 1 & -5 & 10 \end{bmatrix}, \tag{2.8}$$

$$Q_2 = \frac{1}{8h^2} \begin{bmatrix} -14 & 3 & 1 & & & & \\ 3 & -10 & 4 & 1 & & & \\ 1 & 4 & -10 & 4 & 1 & & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \\ & & 1 & 4 & -10 & 4 & 1 \\ & & & 1 & 4 & -10 & 3 \\ & & & & 1 & 3 & -14 \end{bmatrix}, \tag{2.9}$$

$$Q_0 = \frac{1}{384}\begin{bmatrix} 154 & 75 & 1 & & & & \\ 75 & 230 & 76 & 1 & & & \\ 1 & 76 & 230 & 76 & 1 & & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \\ & & 1 & 76 & 230 & 76 & 1 \\ & & & 1 & 76 & 230 & 75 \\ & & & & 1 & 75 & 154 \end{bmatrix}. \tag{2.10}$$

It is worth noticing that Q_4, Q_2, Q_0 have same eigenvectors, and the equation (2.7) can be solved by a Fast Fourier Transform (FFT) algorithm[27].

2.3 Harmonic Problem with Dirichlet and Neumann Boundary Conditions

In this section, we consider the Quartic SC method to discretize the BVP equation (2.3) with boundary conditions

$$u(a) = u(b) = 0, \tag{2.11}$$

$$u'(a) = u'(b) = 0. \tag{2.12}$$

We first adjust the quartic spline basis functions so that they satisfy the boundary conditions (2.11)-(2.12) by construction. The basis functions Φ_i,

$i = -1, \cdots, N+2$, of (2.2) are adjusted by

$$\begin{aligned}
\widetilde{\Phi}_1 &= \Phi_1 + \frac{33}{4}\Phi_{-1} - \frac{7}{4}\Phi_0, & \widetilde{\Phi}_2 &= \Phi_2 + \frac{7}{4}\Phi_{-1} - \frac{1}{4}\Phi_0, \\
\widetilde{\Phi}_i &= \Phi_i, & i &= 3, \cdots, N, \\
\widetilde{\Phi}_{N-1} &= \Phi_{N-1} + \frac{7}{4}\Phi_{N+2} - \frac{1}{4}\Phi_{N+1}, & \widetilde{\Phi}_N &= \Phi_N + \frac{33}{4}\Phi_{N+2} - \frac{7}{4}\Phi_{N+1}.
\end{aligned} \tag{2.13}$$

The space $\widetilde{S}_{4,\Delta}$ spanned by $\{\widetilde{\Phi}_i, i = 1, \ldots, N\}$ is the space of quartic splines that satisfy (1.8)-(1.9). Any $u_\Delta(x) = \sum_{i=1}^N \tilde{c}_i \tilde{\Phi}_i(x)$ is a quartic spline and satisfies the boundary conditions (1.8)-(1.9) by construction of the $\tilde{\Phi}_i$'s.

Applied to the boundary value problem (1.7) with boundary conditions (1.8) and (1.9), the standard quartic spline collocation method determines an approximation $u_\Delta(x) \in \widetilde{S}_{4,\Delta}$ to u, that satisfies (1.7) on the set $T \equiv \{\tau_i, i = 1, \cdots, N\}$. Let $u_\Delta(x) = \sum_{i=1}^N \tilde{c}_i \tilde{\Phi}_i(x)$. The collocation equations

$$u_\Delta^{(4)}(\tau_i) + u_\Delta^{(2)}(\tau_i) + u_\Delta(\tau_i) = g(\tau_i), i = 1, \cdots, N, \tag{2.14}$$

lead to an $N \times N$ linear system

$$Q_{DN}\bar{c} = (Q_{DN4} + Q_{DN2} + Q_{DN0})\bar{c} = \bar{g}, \tag{2.15}$$

where $\bar{c} = (\tilde{c}_1, \tilde{c}_2, \cdots, \tilde{c}_N)^T$, $\bar{g} = (g(\tau_1), g(\tau_2), \cdots, g(\tau_N))^T$, and the matrices Q_{DN4}, Q_{DN2} and Q_{DN0} arise from the terms $u^{(4)}$, $u^{(2)}$ and $u^{(0)}$ of (1.7), respectively.

In the following, we give the forms of $Q_{DNi}, i = 0, 2, 4$:

$$Q_{DN4} = \frac{1}{4h^4}\begin{bmatrix} 85 & -5 & 4 & & & & \\ -23 & 23 & -16 & 4 & & & \\ 4 & -16 & 24 & -16 & 4 & & \\ & . & . & . & . & . & \\ & & 4 & -16 & 24 & -16 & 4 \\ & & & 4 & -16 & 23 & -23 \\ & & & & 4 & -5 & 85 \end{bmatrix}, \tag{2.16}$$

$$Q_{DN2} = \frac{1}{32h^2}\begin{bmatrix} -35 & 19 & 4 & & & & \\ 9 & -41 & 16 & 4 & & & \\ 4 & 16 & -40 & 16 & 4 & & \\ & . & . & . & . & . & \\ & & 4 & 16 & -40 & 16 & 4 \\ & & & 4 & 16 & -41 & 9 \\ & & & & 4 & 19 & -35 \end{bmatrix}, \tag{2.17}$$

$$Q_{DN0} = \frac{1}{1536}\begin{bmatrix} 421 & 235 & 4 & & & & \\ 297 & 919 & 304 & 4 & & & \\ 4 & 304 & 920 & 304 & 4 & & \\ & . & . & . & . & . & \\ & & 4 & 304 & 920 & 304 & 4 \\ & & & 4 & 304 & 919 & 297 \\ & & & & 4 & 235 & 421 \end{bmatrix}. \tag{2.18}$$

2.4 Notes

The boundary conditions (2.4), (2.5), (2.11), and (2.12) considered in this chapter are all homogeneous. This may seem at first a restriction to the applicability of the Quartic SC method and of any solution technique for the arising matrices. We would like to emphasize that this is not the case.

If the boundary conditions are non-homogeneous, for example as in (1.8)-(1.9), the quartic spline space $\widetilde{S}_{4,\Delta}$ and the associated basis functions $\{\widetilde{\Phi}_i, i = 1, \ldots, N\}$, cannot be used to represent the approximate solution. In this case, the space $S_{4,\Delta}$ and the associated basis functions $\{\Phi_i, i = -1, \ldots, N+2\}$, are used. The basis functions $\{\Phi_i, i = -1, \ldots, N+2\}$, give rise to matrices which differ slightly from the matrices Q_{DN4}, Q_{DN2} and Q_{DN0} presented in (2.16), (2.17) and (2.18), respectively, and therefore the collocation equations lead to a linear system that differs slightly from the one in (2.15). In particular, the linear system arising from $\{\Phi_i, i = -1, \ldots, N+2\}$, has $N+4$ equations and the approximation has $N+4$ degrees of freedom, with the four extra equations and associated degrees of freedom corresponding to the boundary conditions.

With appropriate row operations, we can transform the linear system arising from $\{\Phi_i, i = -1, \ldots, N+2\}$ with non-homogeneous boundary conditions to an equivalent one with a matrix such as in (2.15). More specifically, we eliminate the boundary degrees of freedom and appropriately adjust the right-side vector. Once the resulting linear system is solved and the interior degrees of freedom are computed, the boundary degrees of freedom are recovered from the boundary equations. We choose not to give the details of this procedure, because the focus of our work is the study of the properties of the matrix in (2.15).

Another thing that needs to be mentioned is that, though we discuss only the standard Quartic SC method for the harmonic problems, the solution techniques developed in this paper are applicable to the optimal three-step Quartic SC method described in [27], that achieves sixth order approximations. Notice that the three-step sixth order method in [27] proceeds in three steps, each step involving the solution of a linear system with the same matrix and different right side. The matrix resulting from each step of the optimal Quartic SC method is the same as (2.15). We focus our attention on the study of the matrix properties, this is why we do not present how the different right side vectors are formed.

Chapter 3

Generalized Minimal Residual(GMRES) Method

Before we start our discussion about solving the linear system (2.15), we will briefly introduce the Generalized Minimal Residual (GMRES) method, an iterative method that is often applied to solving linear systems $Ax = b$, where A is nonsymmetric. Detailed discussions about GMRES can be found in [21]. We will also prove two theorems which characterize the convergence performance of the GMRES method.

3.1 Basic GMRES

GMRES is designed to solve nonsymmetric linear systems. Starting with a given approximate solution, it generates a sequence of orthogonal vectors and updates the approximation by the linear combination of these vectors. The following is the basic GMRES algorithm.

Algorithm: Basic GMRES
Choose $x^{(0)}$, Compute $r^{(0)} = b - Ax^{(0)}$, $\beta = \left\|r^{(0)}\right\|_2$
If $\beta \leq \epsilon$, set $x = x^{(0)}$ and exit **EndIf**
Compute $v^{(1)} = r^{(0)}/\beta$
For $k = 1$ to maxit **Do**
Compute $w^{(k)} = Av^{(k)}$
For $i = 1$ to k **Do**
$h_{ik}^{(k)} = (w^{(k)}, v^{(i)})$ $w^{(k)} = w^{(k)} - h_{ik}^{(k)}v^{(i)}$
EndFor

$h_{k+1,k}^{(k)} = \left\|w^{(k)}\right\|_2$
    If $h_{k+1,k}^{(k)} == 0$ break **EndIf**
    $v^{(k+1)} = w^{(k)}/h_{k+1,k}^{(k)}$

EndFor
Compute the minimizer y_k of $\left\|\beta e_1^{(k)} - \bar{H}_k y\right\|_2$
Compute $x^{(k)} = x^{(0)} + V_k y_k$, $r^{(k)} = b - Ax^{(k)}$, $\alpha = \left\|r^{(k)}\right\|_2$
If $\alpha \leq \epsilon$, set $x = x^{(k)}$ and exit successfully

In the above:

- $e_1^{(k)}$ is the first unit vector of size $k+1$.
- $\bar{H}_k$ is an $(k+1) \times k$ matrix with $\bar{H}_k = \{h_{il}^{(k)}\}_{1 \leq i \leq k+1, 1 \leq l \leq k}$.

- V_k is an $n \times k$ orthogonal matrix with columns $\{v^{(i)}\}_{1 \le i \le k}$.

3.2 Convergence Properties of Basic GMRES

It is trivial to see that in exact arithmetic the basic GMRES method will terminate at the solution in no more than n steps. However, when the distribution of the eigenvalues of A has certain favorable features, the algorithm will locate the solution in many fewer than n iterations. To show this we start with the following lemma [21].

Lemma 3.2.1 *Let x_{k+1} be the approximate solution obtained from the $(k+1)$-st step of the GMRES algorithm, and let $r_{k+1} = b - Ax_{k+1}$. Then, x_{k+1} is of the form*

$$x_{k+1} = x_0 + q_{k+1}(A)r_0$$

and

$$\|r_{k+1}\|_2^2 = \|(I - Aq_{k+1}(A))r_0\|_2^2 = \min_{q \in P_k} \|(I - Aq(A))r_0\|_2^2, \qquad (3.2)$$

where q_{k+1} is a polynomial of degree k, and P_k is the set of polynomials of degree k.

Proof. See Saad([21], p.193). □

To make this lemma more applicable, it is convenient to recast it in terms of eigenvectors and eigenvalues of the matrix A. We will prove two theorems that relate the convergence of GMRES with the distribution of eigenvalues of A. The technique used here is similar to that used in ([19], p.114). In the following discussion, without explicit mention, we assume that the eigenvectors of A are linearly independent.

Let $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of A, and let $v_1, v_2, \cdots, v_n$ be the corresponding eigenvectors. Since these eigenvectors are linearly independent, they span the whole space $\mathbf{R}^n$ and r_0 can be written as follows:

$$r_0 = \sum_{i=1}^{n} \xi_i v_i, \tag{3.3}$$

for some coefficients ξ_i. It is trivial to see that any eigenvector of A is also an eigenvector of $P_k(A)$, so we have

$$P_k(A)v_i = P_k(\lambda_i)v_i, \quad i = 1, 2, \cdots, n. \tag{3.4}$$

Substituting (3.3) and (3.4) into (3.2), we obtain

$$\begin{aligned} \|r_{k+1}\|_2^2 &= \min_{q \in P_k} \left\| \sum_{i=1}^{n} (1 - \lambda_i q(\lambda_i)) \xi_i v_i \right\|_2^2 \\ &\leq \min_{q \in P_k} \left\| \sum_{i=1}^{n} |1 - \lambda_i q(\lambda_i)| \times |\xi_i v_i| \right\|_2^2 . \end{aligned}$$

Extracting the largest of the terms $|1 - \lambda_i q(\lambda_i)|, \quad i = 1, 2, \cdots, n$, from the above expression, we have

$$\|r_{k+1}\|_2^2 \leq \min_{q \in P_k} \max_{1 \leq i \leq n} (1 - \lambda_i q(\lambda_i))^2 \left\| \sum_{i=1}^{n} |\xi_i v_i| \right\|_2^2 .$$

With $\kappa_0 = \|\sum_{i=1}^{n} |\xi_i v_i|\|_2^2$, we have

$$\|r_{k+1}\|_2^2 \leq \min_{q \in P_k} \max_{1 \leq i \leq n} (1 - \lambda_i q(\lambda_i))^2 \kappa_0. \tag{3.5}$$

Now the convergence rate of the GMRES method can be quantified if we can estimate the nonnegative scalar quantity

$$\min_{q \in P_k} \max_{1 \le i \le n} (1 - \lambda_i q(\lambda_i))^2.$$

By constructing appropriate polynomials $q(x)$ explicitly, we prove two theorems that characterize the convergence properties of GMRES from the distribution of the eigenvalues of A.

Theorem 3.2.2 *If A has eigenvalues $0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$, then we have*

$$\|r_{k+1}\|_2^2 \le \left(\frac{\lambda_{n-k} - \lambda_1}{\lambda_{n-k} + \lambda_1}\right)^2 \kappa_0, \tag{3.6}$$

where κ_0 is a constant.

Proof. To prove (3.6), we define a polynomial Q_{k+1} of degree $k+1$ as follows:

$$Q_{k+1}(\lambda) = \frac{(-1)^{k+1}}{\lambda_n \lambda_{n-1} \cdots \lambda_{n-k+1} \frac{\lambda_1 + \lambda_{n-k}}{2}} (\lambda - \lambda_n)(\lambda - \lambda_{n-1}) \cdots (\lambda - \lambda_{n-k+1})(\lambda - \frac{\lambda + \lambda_{n-k}}{2}).$$

Noting that $Q_{k+1}(\lambda_i) = 0$ for $i = n, n-1, \cdots, n-k+1$, $Q_{k+1}(\frac{\lambda_1 + \lambda_{n-k}}{2}) = 0$, and $Q_{k+1}(0) = 1$, we can deduce that $1 - Q_{k+1}(\lambda)$ is a polynomial of degree $k+1$ with a root at $\lambda = 0$, so the function $\bar{q}$ defined by $\bar{q} = (1 - Q_{k+1}(\lambda))/\lambda$ is a polynomial of degree k. For $\bar{q}$, we have

$$\begin{aligned}
\max_{1 \le i \le n} |1 - \lambda \bar{q}(\lambda_i)| &= \max_{1 \le i \le n} Q_{k+1}(\lambda_i) \\
&= Q_{k+1}(\lambda_1) \\
&= (1 - \frac{\lambda_1}{\lambda_n})(1 - \frac{\lambda_1}{\lambda_{n-1}}) \cdots (1 - \frac{\lambda_1}{\lambda_{n-k+1}})(1 - \frac{\lambda_1}{\frac{\lambda_1 + \lambda_{n-k}}{2}}) \\
&< \frac{\lambda_{n-k} - \lambda_1}{\lambda_{n-k} + \lambda_1}.
\end{aligned} \tag{3.7}$$

¿From (3.5) and (3.7), we can obtain (3.6). $\square$

Theorem 3.2.3 *If A has only k distinct positive eigenvalues, then the GMRES method will identify the solution in at most k iterations.*

Proof. Let $0 < \sigma_1 < \sigma_2 < \cdots < \sigma_k$ be the distinct positive eigenvalues of A. We define a polynomial Q_k of degree k as follows:

$$Q_k(\lambda) = \frac{(-1)^k}{\sigma_1 \sigma_2 \cdots \sigma_k}(\lambda - \sigma_1)(\lambda - \sigma_2)\cdots(\lambda - \sigma_k).$$

Since $Q_k(0) = 1$, $1 - Q_k(\lambda)$ is a polynomial of degree k with a root at $\lambda = 0$. Let $\tilde{q}_{k-1}(\lambda) = (1 - Q_k(\lambda))/\lambda$, then $\tilde{q}_{k-1}(\lambda)$ is a polynomial of degree $k-1$, and we have

$$0 \leq \max_{1 \leq i \leq k}(1 - \sigma_i \tilde{q}_{k-1}(\sigma_i))^2 = \max_{1 \leq i \leq k}(Q_k(\sigma_i))^2 = 0. \tag{3.8}$$

Combining (3.8) and (3.5), we have $r_k = 0$, that is, x_k is the solution. $\square$

3.3 Basic GMRES with Preconditioning

We have seen that the convergence performance of the basic GMRES method depends on the eigenvalue distribution of A. In practice, a process known as *preconditioning* is often employed to improve the eigenvalue distribution so that the GMRES method converges in fewer iterations. The general idea of preconditioning is to find a matrix P which satisfies the following requirements:

- P is easily solvable, or its (approximate) inverse is known and sparse.

- the matrix $P^{-1}A$ has a more desirable eigenvalue distribution (that is, the eigenvalues are clustered), so that the GMRES method applied to $P^{-1}Ax = P^{-1}b$ converges faster than when applied to $Ax = b$.

P is called the **preconditioner**, and $P^{-1}A$ is called the **preconditioned matrix**.

Now we integrate preconditioning with the basic GMRES algorithm. The pseudocode for the basic GMRES algorithm with preconditioner P is given as follows:

Algorithm: Basic GMRES with preconditioning
Choose $x^{(0)}$, Compute $r^{(0)} = b - Ax^{(0)}$, solve $Pq^{(0)} = r^{(0)}$, $\beta = \left\|q^{(0)}\right\|_2$
If $\beta \leq \epsilon$, set $x = x^{(0)}$ and exit **EndIf**
Compute $v^{(1)} = q^{(0)}/\beta$
For $k = 1$ to maxit **Do**
Compute $w^{(k)} = Av^{(k)}$, solve $P\bar{w}^{(k)} = w^{(k)}$
For $i = 1$ to k **Do**
$h_{ik}^{(k)} = (\bar{w}^{(k)}, v^{(i)})$, $\bar{w}^{(k)} = \bar{w}^{(k)} - h_{ik}^{(k)} v^{(i)}$
EndFor
$h_{k+1,k}^{(k)} = \left\|\bar{w}^{(k)}\right\|_2$
If $h_{k+1,k}^{(k)} == 0$ break **EndIf**
$v^{(k+1)} = \bar{w}^{(k)}/h_{k+1,k}^{(k)}$
EndFor
Compute the minimizer y_k of $\left\|\beta e_1^{(k)} - \bar{H}_k y\right\|_2$
Compute $x^{(k)} = x^{(0)} + V_k y_k$, $r^{(k)} = b - Ax^{(k)}$, solve $Pq^{(k)} = r^{(k)}$, $\beta = \left\|q^{(k)}\right\|_2$
If $\beta \leq \epsilon$, set $x = x^{(k)}$ and exit successfully **EndIf**

3.4 Restarted GMRES: GMRES(m)

The GMRES method needs to store all previously computed vectors in the orthogonal sequence, and involves solving a minimization problem. When k is large, the solution of the minimization problem becomes impractical because of the growth of memory and computational requirements. One widely used remedy is *Restarted GMRES*.

The basic idea of Restarted GMRES is as follows: Let m be the maximum number of steps that are tolerable. If the approximate solution obtained by the Basic GMRES x_m is not satisfactory, then $x^{(m)}$ can be used as the initial guess for the next run of GMRES. The following algorithm is the Restarted GMRES with preconditioning.

Algorithm: GMRES(m) with preconditioning

Choose $x^{(0)}$

For $l = 1$ to maxoutit **Do**

Compute $r^{(0)} = b - Ax^{(0)}$, solve $Pq^{(0)} = r^{(0)}$, $\beta = \left\|q^{(0)}\right\|_2$

If $\beta \leq \epsilon$, set $x = x^{(0)}$ and exit **EndIf**

Compute $v^{(1)} = q^{(0)}/\beta$

For $k = 1$ to m **Do**

Compute $w^{(k)} = Av^{(k)}$, solve $P\bar{w}^{(k)} = w^{(k)}$

For $i = 1$ to k **Do**

$h_{ik}^{(k)} = (\bar{w}^{(k)}, v^{(i)})$, $\bar{w}^{(k)} = \bar{w}^{(k)} - h_{ik}^{(k)} v^{(i)}$

EndFor

$h_{k+1,k}^{(k)} = \left\|\bar{w}^{(k)}\right\|_2$

If $h_{k+1,k}^{(k)} == 0$ break **EndIf**

$v^{(k+1)} = \bar{w}^{(k)}/h_{k+1,k}^{(k)}$

EndFor

Compute the minimizer y_m of $\left\|\beta e_1^{(m)} - \bar{H}_m y\right\|_2$

Compute $x^{(m)} = x^{(0)} + V_m y_m$, $r^{(m)} = b - Ax^{(m)}$, solve $Pq^{(m)} = r^{(m)}$, $\beta = \left\|q^{(m)}\right\|_2$

If $\beta \leq \epsilon$, set $x = x^{(m)}$ and Exit

Else $x^{(0)} = x^{(m)}$ **EndIf**

EndFor

Chapter 4

Preconditioned GMRES for Quartic SC Equations

In this chapter, we apply the preconditioned basic GMRES method to solving the equations (2.15). We discuss the properties of the preconditioned matrix, and based on them, analyze the algorithm's convergence.

4.1 Preconditioner

Notice that the only difference between problems (1.7) -(1.9) and (2.3)-(2.5) is two of the four boundary conditions. Therefore, their matrices after discretization, that is, Q_{DN} and K, differ only in the top left and bottom right 2×2 submatrices. Additionally, as Zhu pointed out, the matrices Q_4, Q_2, Q_0 have the same eigenvectors, and the equation (2.7) can be solved by a Fast

Fourier Transform (FFT) algorithm[27]. These observations suggest that the matrix K can serve as a suitable preconditioner when the preconditioned GMRES method is applied to the DE problem (2.15).

From the forms of $Q_{DNi}, i = 0, 2, 4$, we can see that, for sufficiently small h, the entries of Q_{DN4} dominate those of Q_{DN0} and Q_{DN2}. A similar property holds for $Q_i, i = 0, 2, 4$. So we will concentrate on analyzing the matrices associated with the $u^{(4)}$ term, that is, Q_{DN4} and Q_4.

4.2 Eigenvalues and Eigenvectors of the Preconditioned Matrix

In this section, we derive expressions for the eigenvalues and eigenvectors of the preconditioned matrix, so that we can analyze the eigenvalue distribution of the preconditioned matrix.

We need a formal definition of *the reverse vector* for our discussion.

Definition 4.2.1 *For two vectors* $\mathbf{v} = [v_1, v_2, \ldots, v_n]^T$ *and* $\mathbf{u} = [u_1, u_2, \ldots, u_n]^T$, $\mathbf{u}$ *is said to be the reverse of* $\mathbf{v}$ *if*

$$u_i = v_{n+1-i}, \quad \textit{for} \quad 1 \leq i \leq n.$$

Let Q_{DN4} and Q_4 be $n \times n$ matrices as in (2.16) and (2.8) respectively.

Let $D = Q_{DN4} - Q_4$. Then from the forms of Q_{DN4} and Q_4 , we get

$$D = \frac{1}{4h^2} \begin{bmatrix} \frac{45}{4} & \frac{15}{4} & 0 & & & & \\ -\frac{3}{4} & -\frac{1}{4} & 0 & 0 & & & \\ 0 & 0 & 0 & 0 & 0 & & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \\ & & 0 & 0 & 0 & 0 & 0 \\ & & & 0 & 0 & -\frac{1}{4} & -\frac{3}{4} \\ & & & & 0 & \frac{15}{4} & \frac{45}{4} \end{bmatrix}. \tag{4.1}$$

It is worth noticing the special structure of D. There are only eight entries in D that are non-zeros, more specifically, four elements at the upper left corner, and four elements at the bottom right corner. Additionally, the two entries in the first column are three times those in the second column, and a similar property holds for the last two columns.

Let P_{DN} be the preconditioned matrix, that is, $P_{DN} = Q_4^{-1}Q_{DN4}$. Then we have

$$P_{DN} = Q_4^{-1}Q_{DN4} = I + Q_4^{-1}D.$$

Let $\mathbf{v}_i$ denote the ith column vector of $Q_4^{-1}D$. From the above discussion of D, it is easy to show the relations

$$\begin{aligned} \mathbf{v}_1 &= 3\mathbf{v}_2 \\ \mathbf{v}_i &= \mathbf{0}, \ i = 3, 4, \cdots, n-3 \\ \mathbf{v}_n &= 3\mathbf{v}_{n-1} \end{aligned}$$

so $P_{DN} = I + [3\mathbf{v}_2, \mathbf{v}_2, \mathbf{0}, \cdots, \mathbf{0}, \mathbf{v}_{n-1}, 3\mathbf{v}_{n-1}]$.

In addition, another important property of $Q_4^{-1}D$ is that $\mathbf{v}_2$ is the reverse of $\mathbf{v}_{n-1}$. Since Q_4 and D are both centrosymmetric, $Q_4^{-1}D$ is centrosymmetric (see Appendix A). From the definition of centrosymmetric matrices, we know that $\mathbf{v}_2$ is the reverse of $\mathbf{v}_{n-1}$. If $\mathbf{v}_2 = [v_1, v_2, \cdots, v_{n-1}, v_n]^T$, then $\mathbf{v}_{n-1} = [v_n, v_{n-1}, \cdots, v_2, v_1]^T$.

Let $\boldsymbol{\mu} = [\mu_1, \mu_2, \cdots, \mu_{n-1}, \mu_n]^T$ be an eigenvector of P_{DN} and λ be the corresponding eigenvalue. Then we have

$$
\begin{aligned}
P_{DN}\boldsymbol{\mu} &= (I + Q_4^{-1}D)\boldsymbol{\mu} \\
&= \boldsymbol{\mu} + Q_4^{-1}D\boldsymbol{\mu} \\
&= \lambda\boldsymbol{\mu}
\end{aligned}
$$

Hence $Q_4^{-1}D\boldsymbol{\mu} = (\lambda - 1)\boldsymbol{\mu}$, that is,

$$
\begin{pmatrix}
3v_1 & v_1 & 0 & \cdots & 0 & v_n & 3v_n \\
3v_2 & v_2 & 0 & \cdots & 0 & v_{n-1} & 3v_{n-1} \\
\vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\
3v_i & v_i & 0 & \cdots & 0 & v_{n-i+1} & 3v_{n-i+1} \\
\vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\
3v_{n-1} & v_{n-1} & 0 & \cdots & 0 & v_2 & 3v_2 \\
3v_n & v_n & 0 & \cdots & 0 & v_1 & 3v_1
\end{pmatrix}
\begin{pmatrix}
\mu_1 \\ \mu_2 \\ \vdots \\ \mu_i \\ \vdots \\ \mu_{n-1} \\ \mu_n
\end{pmatrix}
= (\lambda-1)
\begin{pmatrix}
\mu_1 \\ \mu_2 \\ \vdots \\ \mu_i \\ \vdots \\ \mu_{n-1} \\ \mu_n
\end{pmatrix}
$$

Then we get the equations

$$
\begin{aligned}
3v_1\mu_1 + v_1\mu_2 + v_n\mu_{n-1} + 3v_n\mu_n &= (\lambda-1)\mu_1 && (4.2)\\
3v_2\mu_1 + v_2\mu_2 + v_{n-1}\mu_{n-1} + 3v_{n-1}\mu_n &= (\lambda-1)\mu_2 && (4.3)\\
3v_i\mu_1 + v_i\mu_2 + v_{n-i+1}\mu_{n-1} + 3v_{n-i+1}\mu_n &= (\lambda-1)\mu_i, && (4.4)\\
& \quad i = 3, \cdots, n-2, && \\
3v_{n-1}\mu_1 + v_{n-1}\mu_2 + v_2\mu_{n-1} + 3v_2\mu_n &= (\lambda-1)\mu_{n-1}, && (4.5)\\
3v_{n-1}\mu_1 + v_{n-1}\mu_2 + v_2\mu_{n-1} + 3v_2\mu_n &= (\lambda-1)\mu_{n-1}. && (4.6)
\end{aligned}
$$

We now discuss the solutions to the equations (4.2)-(4.6). Our goal is to get explicit forms for λ and μ_i, $i = 1, 2, \cdots, n$.

By adding (4.2) and (4.6), we have

$$3(v_1+v_n)(\mu_1+\mu_n) + (v_1+v_n)(\mu_2+\mu_{n-1}) = (\lambda-1)(\mu_1+\mu_n),$$

that is,

$$(v_1+v_n)(\mu_2+\mu_{n-1}) = [\lambda - (1+3(v_1+v_n)](\mu_1+\mu_n). \tag{4.7}$$

Similarly, by adding (4.3) and (4.5), we get

$$3(v_2+v_{n-1})(\mu_1+\mu_n) = [\lambda - (1+v_2+v_{n-1})](\mu_2+\mu_{n-1}). \tag{4.8}$$

From (4.7) and (4.8), we see that

$$
\begin{aligned}
& 3(v_1+v_n)(v_2+v_{n-1})(\mu_1+\mu_n)(\mu_2+\mu_{n-1}) \\
& = [\lambda - (1+3(v_1+v_n)][\lambda - (1+v_2+v_{n-1})](\mu_1+\mu_n)(\mu_2+\mu_{n-1}).
\end{aligned} \tag{4.9}
$$

To make (4.9) hold, it is sufficient to have

$$3(v_1 + v_n)(v_2 + v_{n-1}) = [\lambda - (1 + 3(v_1 + v_n)][\lambda - (1 + v_2 + v_{n-1})]. \quad (4.10)$$

The solutions of λ in (4.10) are

$$\lambda_1 = 1, \quad \lambda_2 = 1 + (3v_1 + v_2) + (3v_n + v_{n-1}). \quad (4.11)$$

By subtracting (4.6) from (4.2), and (4.5) from (4.3), we have

$$\begin{aligned} (v_1 - v_n)(\mu_2 - \mu_{n-1}) &= [\lambda - (1 + 3(v_1 - v_n)](\mu_1 - \mu_n), \quad (4.12) \\ 3(v_2 - v_{n-1})(\mu_1 - \mu_n) &= [\lambda - (1 + v_2 - v_{n-1})](\mu_2 - \mu_{n-1}). \quad (4.13) \end{aligned}$$

From (4.12) and (4.13), we get

$$\begin{aligned} &3(v_1 - v_n)(v_2 - v_{n-1})(\mu_1 - \mu_n)(\mu_2 - \mu_{n-1}) \\ &= [\lambda - (1 + 3(v_1 - v_n)][\lambda - (1 + v_2 - v_{n-1})](\mu_1 - \mu_n)(\mu_2 - \mu_{n-1}). \end{aligned} \quad (4.14)$$

Similarly, to satisfy (4.14), it is sufficient to have

$$3(v_1 - v_n)(v_2 - v_{n-1}) = [\lambda - (1 + 3(v_1 - v_n)][\lambda - (1 + v_2 - v_{n-1})]. \quad (4.15)$$

By solving (4.15), we get one more solution for λ

$$\lambda_3 = 1 + (3v_1 + v_2) - (3v_n + v_{n-1}). \quad (4.16)$$

So far, we have found three candidates, namely λ_1, λ_2 in (4.11) and λ_3 in (4.16), for eigenvalues of the preconditioned matrix P_{DN}. Next, we will derive the eigenvectors corresponding to these candidates.

Case 1 $\lambda = 1$. Then the right hand side of the equations (4.2)-(4.6) are zeros. To make the left hand side of the equations equal zeros, there are two possibilities:

(i) μ_1, μ_2, μ_{n-1}, μ_n are all zero. Then μ_i, $i = 3, \cdots, n-2$, can take on any value to make the equations (4.2) -(4.6) valid. Let $\mathbf{e}_i$ denote an unit vector, whose elements are all 0 except the ith one which is 1. It is easy to verify that $\mathbf{e}_i$, $i = 3, \cdots, n-2$, are eigenvectors of P_{DN} with the associated common eigenvalue $\lambda = 1$.

(ii) μ_1, μ_2, μ_{n-1}, μ_n are not all zeros. It is easy to see that $\boldsymbol{\mu}_1 = (1, -3, 0, \cdots, 0)^T$ and $\boldsymbol{\mu}_n = (0, \cdots, 0, -3, 1)^T$ are eigenvectors of P_{DN} with associated common eigenvalue $\lambda = 1$. Notice that $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_n$ are linearly independent of each other and linearly independent of $\mathbf{e}_i$, $i = 3, \cdots, n-2$.

So in total we find $n-2$ linearly independent eigenvectors of P_{DN} with 1 as the corresponding eigenvalue, which means that P_{DN} has $n-2$ identical eigenvalues of 1.

Case 2 $\lambda = 1 + (3v_1 + v_2) + (3v_n + v_{n-1})$. By substituting this equation into the equations (4.2) -(4.6), we have

$$\begin{aligned}
\mu_1 &= \mu_n = v_1 + v_n, \\
\mu_2 &= \mu_{n-1} = v_2 + v_{n-1}, \\
\mu_i &= v_i + v_{n-i+1}, \quad i = 3, \cdots, n-2.
\end{aligned}$$

Case 3 $\lambda = 1 + (3v_1 + v_2) - (3v_n + v_{n-1})$. By substituting this equation into the equations (4.2) -(4.6), we have

$$\begin{aligned}
\mu_1 &= v_1 - v_n, \\
\mu_2 &= v_2 - v_{n-1}, \\
\mu_i &= v_i - v_{n-i+1}, \quad i = 3, \cdots, n-2, \\
\mu_{n-1} &= v_{n-1} - v_2, \\
\mu_n &= v_n - v_1.
\end{aligned}$$

4.3 Linear Independence of Eigenvectors

We have found three eigenvalues and n eigenvectors. In this section, we show that the eigenvalues are the only distinct eigenvalues of P_{DN}, and that the eigenvectors of P_{DN} are linearly independent, so that from Theorem 3.2.3 we can claim that the GMRES method will stop in three iterations.

To prove that the two eigenvectors obtained from **Case 2** and **Case 3** are linearly independent from those in **Case 1**, we only need to show that

$$\lambda_1 \neq \lambda_2 \neq \lambda_3,$$

that is,

$$(3v_1 + v_2) + (3v_n + v_{n-1}) \neq (3v_1 + v_2) - (3v_n + v_{n-1}) \neq 0. \tag{4.17}$$

To show this is true, we first derive the expressions of v_1, v_2, v_{n-1}, and v_n.

Let T_{-2}^{hd} be a special type of matrices that is defined by

$$T_{-2}^{hd} = \begin{bmatrix} -3 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & \cdot & \cdot & \cdot & & \\ & & 1 & -2 & 1 & \\ & & & & 1 & -3 \end{bmatrix}$$

As shown in [8], the eigenvalues λ_i^{hd}, $i = 1, \cdots, n$, of T_{-2}^{hd} are given by

$$\lambda_i^{hd} = -4\sin^2\frac{i\pi}{2n},$$

and its normalized eigenvectors δ_i^{hd}, $i = 1, \cdots, n$, are

$$\delta_{ij}^{hd} = k_i \sin\frac{(2j-1)i\pi}{2n}, j = 1, \cdots, n$$

where

$$k_i = \sqrt{\frac{2}{n}},\ i = 1, \cdots, n-1, \text{and } k_n = \sqrt{\frac{1}{n}}.$$

By observing that $Q_4 = T_{-2}^{hd} T_{-2}^{hd}/(4h^4)$, we see that the eigenvalues of Q_4 are

$$\lambda_i^q = \frac{1}{4h^4}(\lambda_i^{hd})^2 = \frac{16}{4h^4}\sin^4\frac{i\pi}{2n},\ i = 1, \cdots, n,$$

and its eigenvectors are the same as those of T_{-2}^{hd}. Then Q_4 can be written as

$$Q_4 = [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}] \begin{bmatrix} \lambda_1^q & & \\ & \ddots & \\ & & \lambda_n^q \end{bmatrix} [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}]^{-1}$$

Recall that δ_i^{hd}, $i = 1, 2, \cdots, n$ are normalized eigenvectors of T_{-2}^{hd}, then

$$[\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}]^{-1} = [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}]^T$$

Hence,

$$Q_4^{-1} = [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}] \begin{bmatrix} 1/\lambda_1^q & & \\ & \ddots & \\ & & 1/\lambda_n^q \end{bmatrix} \begin{bmatrix} (\delta_1^{hd})^T \\ \vdots \\ (\delta_n^{hd})^T \end{bmatrix} \tag{4.18}$$

¿From (4.1) and (4.18), we have

$$
Q_4^{-1}D = \quad \left[\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}\right] \begin{bmatrix} 1/(\lambda_1^{hd})^2 & & \\ & \ddots & \\ & & 1/(\lambda_n^{hd})^2 \end{bmatrix} \times
$$

$$
\begin{bmatrix} (\delta_1^{hd})^T \\ \vdots \\ (\delta_n^{hd})^T \end{bmatrix}
\begin{bmatrix}
45/4 & 15/4 & 0 & & & & \\
-3/4 & -1/4 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & 0 & & \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \\
 & & 0 & 0 & 0 & 0 & 0 \\
 & & & 0 & 0 & -1/4 & -3/4 \\
 & & & & 0 & 15/4 & 45/4
\end{bmatrix}
$$

$$
\begin{aligned}
&= [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}] \begin{bmatrix} \frac{1}{(\lambda_1^{hd})^2} & & \\ & \ddots & \\ & & \frac{1}{(\lambda_n^{hd})^2} \end{bmatrix} \times \\
&\begin{bmatrix}
\frac{45}{4}\delta_{1,1}^{hd} - \frac{3}{4}\delta_{1,2}^{hd} & \frac{15}{4}\delta_{1,1}^{hd} - \frac{1}{4}\delta_{1,2}^{hd} & 0 & \cdot & 0 & \frac{15}{4}\delta_{1,n}^{hd} - \frac{1}{4}\delta_{1,n-1}^{hd} & \frac{45}{4}\delta_{1,n}^{hd} - \frac{3}{4}\delta_{1,n-1}^{hd} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{45}{4}\delta_{i,1}^{hd} - \frac{3}{4}\delta_{i,2}^{hd} & \frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd} & 0 & \cdot & 0 & \frac{15}{4}\delta_{i,n}^{hd} - \frac{1}{4}\delta_{i,n-1}^{hd} & \frac{45}{4}\delta_{i,n}^{hd} - \frac{3}{4}\delta_{i,n-1}^{hd} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{45}{4}\delta_{n,1}^{hd} - \frac{3}{4}\delta_{n,2}^{hd} & \frac{15}{4}\delta_{n,1}^{hd} - \frac{1}{4}\delta_{n,2}^{hd} & 0 & \cdot & 0 & \frac{15}{4}\delta_{n,n}^{hd} - \frac{1}{4}\delta_{n,n-1}^{hd} & \frac{45}{4}\delta_{n,n}^{hd} - \frac{3}{4}\delta_{n,n-1}^{hd}
\end{bmatrix} \\
&= [\delta_1^{hd}, \delta_2^{hd}, \cdots, \delta_{n-1}^{hd}, \delta_n^{hd}] \times \\
&\begin{bmatrix}
\frac{\frac{45}{4}\delta_{1,1}^{hd} - \frac{3}{4}\delta_{1,2}^{hd}}{(\lambda_1^{hd})^2} & \frac{\frac{15}{4}\delta_{1,1}^{hd} - \frac{1}{4}\delta_{1,2}^{hd}}{(\lambda_1^{hd})^2} & 0 & \cdot & 0 & \frac{\frac{15}{4}\delta_{1,n}^{hd} - \frac{1}{4}\delta_{1,n-1}^{hd}}{(\lambda_1^{hd})^2} & \frac{\frac{45}{4}\delta_{1,n}^{hd} - \frac{3}{4}\delta_{1,n-1}^{hd}}{(\lambda_1^{hd})^2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\frac{45}{4}\delta_{i,1}^{hd} - \frac{3}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2} & \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2} & 0 & \cdot & 0 & \frac{\frac{15}{4}\delta_{i,n}^{hd} - \frac{i}{4}\delta_{i,n-1}^{hd}}{(\lambda_i^{hd})^2} & \frac{\frac{45}{4}\delta_{i,n}^{hd} - \frac{3}{4}\delta_{i,n-1}^{hd}}{(\lambda_i^{hd})^2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\frac{45}{4}\delta_{n,1}^{hd} - \frac{3}{4}\delta_{n,2}^{hd}}{(\lambda_n^{hd})^2} & \frac{\frac{15}{4}\delta_{n,1}^{hd} - \frac{1}{4}\delta_{n,2}^{hd}}{(\lambda_n^{hd})^2} & 0 & \cdot & 0 & \frac{\frac{15}{4}\delta_{n,n}^{hd} - \frac{1}{4}\delta_{n,n-1}^{hd}}{(\lambda_n^{hd})^2} & \frac{\frac{45}{4}\delta_{n,n}^{hd} - \frac{3}{4}\delta_{n,n-1}^{hd}}{(\lambda_n^{hd})^2}
\end{bmatrix} \\
&= [\mathbf{v}_1 \quad \mathbf{v}_2 \quad 0 \quad \cdots \quad 0 \quad \mathbf{v}_{n-1} \quad \mathbf{v}_n],
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{v}_1 &= 3\mathbf{v}_2, \\
\mathbf{v}_2 &= \left[\sum_{i=1}^n \delta_{i,1}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2} \cdots \sum_{i=1}^n \delta_{i,j}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2} \cdots \sum_{i=1}^n \delta_{i,n}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2} \right]^T, \\
\mathbf{v}_{n-1} &= \left[\sum_{i=1}^n \delta_{i,1}^{hd} \frac{\frac{15}{4}\delta_{i,n}^{hd} - \frac{1}{4}\delta_{i,n-1}^{hd}}{(\lambda_i^{hd})^2} \cdots \sum_{i=1}^n \delta_{i,j}^{hd} \frac{\frac{15}{4}\delta_{i,n}^{hd} - \frac{1}{4}\delta_{i,n-1}^{hd}}{(\lambda_i^{hd})^2} \cdots \sum_{i=1}^n \delta_{i,n}^{hd} \frac{\frac{15}{4}\delta_{i,n}^{hd} - \frac{1}{4}\delta_{i,n-1}^{hd}}{(\lambda_i^{hd})^2} \right]^T, \\
\mathbf{v}_n &= 3\mathbf{v}_{n-1}.
\end{aligned}
$$

It is easy to verify that in the above equations, $\mathbf{v}_2$ is the reverse of $\mathbf{v}_{n-1}$, which is one of the properties of $Q_4^{-1}D$ we have previously discussed.

Now we get the explicit forms of v_1, v_2, v_{n-1}, v_n as follows:

$$v_1 = \sum_{i=1}^{n} \delta_{i,1}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2}, \tag{4.19}$$

$$v_2 = \sum_{i=1}^{n} \delta_{i,2}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2}, \tag{4.20}$$

$$v_{n-1} = \sum_{i=1}^{n} \delta_{i,n-1}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2}, \tag{4.21}$$

$$v_n = \sum_{i=1}^{n} \delta_{i,n}^{hd} \frac{\frac{15}{4}\delta_{i,1}^{hd} - \frac{1}{4}\delta_{i,2}^{hd}}{(\lambda_i^{hd})^2}. \tag{4.22}$$

With these explicit expressions, we are able to evaluate $3v_1+v_2$ and $3v_n+v_{n-1}$.

$$\begin{aligned}
3v_1 + v_2 &= \sum_{i=1}^{n} \frac{\frac{45}{4}(\delta_{i,1}^{hd})^2 + 3\delta_{i,1}^{hd}\delta_{i,2}^{hd} - \frac{1}{4}(\delta_{i,2}^{hd})^2}{(\lambda_i^{hd})^2} \\
&= \frac{1}{4}\sum_{i=1}^{n} \frac{(15\delta_{i,1}^{hd} - \delta_{i,2}^{hd})(3\delta_{i,1}^{hd} + \delta_{i,2}^{hd})}{(\lambda_i^{hd})^2} \\
&= \frac{1}{4}\left[\sum_{i=1}^{n-1} \frac{2}{n} \cdot \frac{(15\sin\frac{i\pi}{2n} - \sin\frac{3i\pi}{2n})(3\sin\frac{i\pi}{2n} + \sin\frac{3i\pi}{2n})}{16\sin^4\frac{i\pi}{2n}} + \frac{1}{n}\cdot\frac{(15+1)(3-1)}{16}\right] \\
&= \frac{1}{32n}\left[\sum_{i=1}^{n-1} \frac{(15\sin\frac{i\pi}{2n} - 3\sin\frac{i\pi}{2n} + 4\sin^3\frac{i\pi}{2n})(6\sin\frac{i\pi}{2n} - 4\sin^3\frac{i\pi}{2n})}{\sin^4\frac{i\pi}{2n}} + 16\right] \\
&= \frac{1}{4n}\left[\sum_{i=1}^{n-1}(\frac{9}{\sin^2\frac{i\pi}{2n}} - 3 - 2\sin^2\frac{i\pi}{2n}) + 2\right] \\
&= \frac{1}{4n}\left[\sum_{i=1}^{n-1}(\frac{18}{1-\cos\frac{i\pi}{n}} - 3 - 1 + \cos\frac{i\pi}{n}) + 2\right] \\
&= \frac{9}{2n}\sum_{i=1}^{n-1}\frac{1}{1-\cos\frac{i\pi}{n}} + \frac{1}{4n}\sum_{i=1}^{n-1}\cos\frac{i\pi}{n} - \frac{2n-3}{2n},
\end{aligned} \tag{4.23}$$

$$
\begin{aligned}
&3v_n + v_{n-1} \\
&= \frac{1}{4}\sum_{i=1}^{n} \frac{(15\delta_{i,1}^{hd} - \delta_{i,2}^{hd})(3\delta_{i,n}^{hd} + \delta_{i,n-1}^{hd})}{(\lambda_i^{hd})^2} \\
&= \frac{1}{4}\sum_{i=1}^{n} (-1)^{i+1} k_i^2 \cdot \frac{(15\sin\frac{i\pi}{2n} - \sin\frac{3i\pi}{2n})(3\sin\frac{i\pi}{2n} + \sin\frac{3i\pi}{2n})}{16\sin^4\frac{i\pi}{2n}} \\
&= \frac{1}{4}\sum_{i=1}^{n} (-1)^{i+1} k_i^2 \cdot \frac{(15\sin\frac{i\pi}{2n} - 3\sin\frac{i\pi}{2n} + 4\sin^3\frac{i\pi}{2n})(6\sin\frac{i\pi}{2n} - 4\sin^3\frac{i\pi}{2n})}{16\sin^4\frac{i\pi}{2n}} \\
&= \frac{1}{8}\sum_{i=1}^{n} (-1)^{i+1} k_i^2 \cdot (\frac{9}{\sin^2\frac{i\pi}{2n}} - 3 - 2\sin^2\frac{i\pi}{2n}) \\
&= \frac{1}{4n}\left[\sum_{i=1}^{n-1} (-1)^{i+1} (\frac{18}{1-\cos\frac{i\pi}{n}} - 3 - 1 + \cos\frac{i\pi}{n}) + 2(-1)^{n+1}\right] \\
&= \frac{9}{2n}\sum_{i=1}^{n-1} (-1)^{i+1} \frac{1}{1-\cos\frac{i\pi}{n}} + \frac{1}{n}\sum_{i=1}^{n-1}(-1)^{i} + \frac{1}{4n}\sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n} + \frac{(-1)^{n+1}}{2n}.
\end{aligned}
\tag{4.24}
$$

Notice that both $3v_1 + v_2$ and $3v_n + v_{n-1}$ have summation terms involving $\cos\frac{i\pi}{n}$. These summations can be evaluated with the help of *Chebyshev Polynomials*. We postpone the detailed evaluation process to the next section, and directly substitute the results here to make our derivation fluent. From (4.23) and (4.24), it can be shown that

$$
(3v_1 + v_2) + (3v_n + v_{n-1}) = \frac{9}{4}n - 1, \tag{4.25}
$$

$$
(3v_1 + v_2) - (3v_n + v_{n-1}) = \frac{3}{4}n - 1, \tag{4.26}
$$

hence, the inequalities (4.17) hold.

Consequently, we know that the preconditioned matrix P_{DN} has three distinct positive eigenvalues and its eigenvectors are linearly independent. By Theorem 3.2.3, we know the GMRES method will stop in three iterations.

4.4 Summation Evaluation

In this section, we present a detailed discussion about how the summation terms in (4.23) and (4.24) are evaluated and show that the equations (4.25) and (4.26) hold. We will intensively use the definitions of *Chebyshev Polynomials* (see Appendix B). Let $T_n(x)$, $U_n(x)$, $V_n(x)$ and $W_n(x)$ be the n-th degree Chebyshev polynomials of the first, second, third and fourth kind.

Lemma 4.4.1 *For $n \in \mathbb{Z}^+$, $n > 1$,*

$$\sum_{i=1}^{n-1} \frac{1}{1 - \cos \frac{i\pi}{n}} = \frac{n^2 - 1}{3}. \tag{4.27}$$

Proof. It is trivial to show that this formula holds when $n = 2$. So in the following proof, we assume $n \geq 3$.

By the definition (B.20), we have

$$U_{n-1}(x) = 2^{n-1} \prod_{i=1}^{n-1} \left[x - \cos \frac{i\pi}{n} \right], \tag{4.28}$$

and from the definition (B.12), we also have

$$U_{n-1}(x) = \sum_{m=0}^{\lceil (n-1)/2 \rceil} \binom{n}{2m+1} x^{n-1-2m} (x^2 - 1)^m. \tag{4.29}$$

From the formula (4.28), we obtain

$$\ln U_{n-1}(x) = (n-1) \ln 2 + \sum_{i=1}^{n-1} \ln \left[x - \cos \frac{k\pi}{n} \right],$$

then,

$$[\ln U_{n-1}]'(x) = \sum_{i=1}^{n-1} \frac{1}{x - \cos \frac{i\pi}{n}}.$$

Hence,

$$[\ln U_{n-1}]'(1) = \sum_{i=1}^{n-1} \frac{1}{1-\cos\frac{i\pi}{n}}. \tag{4.30}$$

On the other hand, by the formula (4.29), we get

$$\begin{aligned}
[\ln U_{n-1}]'(x) &= \frac{U'_{n-1}(x)}{U_{n-1}(x)} \\
&= \frac{\left[\binom{n}{1}x^{n-1} + \binom{n}{3}x^{n-3}(x^2-1)\right]' + \left[\sum_{m=2}^{\lceil (n-1)/2 \rceil}\binom{n}{2m+1}x^{n-1-2m}(x^2-1)^m\right]'}{\binom{n}{1}x^{n-1} + \sum_{m=1}^{\lceil (n-1)/2 \rceil}\binom{n}{2m+1}x^{n-1-2m}(x^2-1)^m}.
\end{aligned}$$

Note that when $x = 1$, we have

$$\left[\sum_{m=2}^{\lceil (n-1)/2 \rceil}\binom{n}{2m+1}x^{n-1-2m}(x^2-1)^m\right]' = 0,$$

$$\sum_{m=1}^{\lceil (n-1)/2 \rceil}\binom{n}{2m+1}x^{n-1-2m}(x^2-1)^m = 0,$$

it follows that

$$\begin{aligned}
[\ln U_{n-1}]'(1) &= \frac{\binom{n}{1}(n-1) + 2\binom{n}{3}}{\binom{n}{1}} \\
&= \frac{n^2-1}{3}.
\end{aligned} \tag{4.31}$$

From (4.30) and (4.31), we obtain the equation (4.27). $\square$

Lemma 4.4.2 *For $n \in \mathbb{Z}^+$, $n > 1$,*

$$\sum_{i=1}^{n-1} \cos\frac{i\pi}{n} = 0. \tag{4.32}$$

Proof. Since

$$\sum_{i=1}^{n-1} \cos\frac{i\pi}{n} = \sum_{i=1}^{n-1} \cos\frac{(n-i)\pi}{n},$$

we have

$$\begin{aligned}
\sum_{i=1}^{n-1} \cos\frac{i\pi}{n} &= \frac{1}{2}\left(\sum_{i=1}^{n-1} \cos\frac{i\pi}{n} + \sum_{i=1}^{n-1} \cos\frac{(n-i)\pi}{n}\right) \\
&= \frac{1}{2}\sum_{i=1}^{n-1}\left(\cos\frac{i\pi}{n} + \cos\frac{(n-i)\pi}{n}\right) \\
&= 0.
\end{aligned}$$

$\square$

By substituting (4.27) and (4.32) into (4.23), we deduce that

$$3v_1 + v_2 = \frac{9}{2n} \times \frac{n^2-1}{3} - \frac{2n-3}{2n} = \frac{3}{2}n - 1. \tag{4.33}$$

Lemma 4.4.3 *For $n \in \mathbb{Z}^+, n > 1$,*

$$\sum_{i=1}^{n-1}(-1)^{i+1}\frac{1}{1-\cos\frac{i\pi}{n}} = \frac{n^2+2}{6}, \quad n \text{ even}, \tag{4.34}$$

$$\sum_{i=1}^{n-1}(-1)^{i+1}\frac{1}{1-\cos\frac{i\pi}{n}} = \frac{n^2-1}{6}, \quad n \text{ odd}. \tag{4.35}$$

Proof. It is easy to show that these formulae hold when $n = 2,\ 3$. So we only consider the cases that $n > 3$.

(i) n is even. Let $n = 2k$, $k \in \mathbb{Z}^+$, $k > 1$. Then

$$\begin{aligned}
\sum_{i=1}^{n-1}(-1)^{i+1}\frac{1}{1-\cos\frac{i\pi}{n}} &= \sum_{i=1}^{k}\frac{1}{1-\cos\frac{(2i-1)\pi}{2k}} - \sum_{i=1}^{k-1}\frac{1}{1-\cos\frac{2i\pi}{2k}} \\
&= \sum_{i=1}^{k}\frac{1}{1-\cos\frac{(2i-1)\pi}{2k}} - \sum_{i=1}^{k-1}\frac{1}{1-\cos\frac{i\pi}{k}} \\
&= \sum_{i=1}^{k}\frac{1}{1-\cos\frac{(2i-1)\pi}{2k}} - \frac{k^2-1}{3}.
\end{aligned}$$

From the definition (B.19), it is easy to show that

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{(2i-1)\pi}{2k}} = [\ln T_k]'(1). \tag{4.36}$$

On the other hand, by the definition (B.11), we have

$$\begin{aligned}
[\ln T_k]'(x) &= \frac{T_k'(x)}{T_k(x)} \\
&= \frac{\left[\binom{k}{0}x^k + \binom{k}{2}x^{k-2}(x^2-1)\right]' + \left[\sum_{m=2}^{\lfloor k/2\rfloor}\binom{k}{2m}x^{k-2m}(x^2-1)^m\right]'}{\binom{k}{0}x^k + \sum_{m=1}^{\lfloor k/2\rfloor}\binom{k}{2m}x^{k-2m}(x^2-1)^m}.
\end{aligned}$$

Since when $x = 1$,

$$\left[\sum_{m=2}^{\lfloor k/2\rfloor}\binom{k}{2m}x^{k-2m}(x^2-1)^m\right]' = 0,$$

$$\sum_{m=1}^{\lfloor k/2\rfloor}\binom{k}{2m}x^{k-2m}(x^2-1)^m = 0,$$

it follows that

$$[\ln T_k]'(1) = \frac{k+2\binom{k}{2}}{1} = k^2,$$

that is,

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{(2i-1)\pi}{2k}} = k^2.$$

Hence,

$$\sum_{i=1}^{n-1}(-1)^{i+1}\frac{1}{1-\cos\frac{i\pi}{n}} = k^2 - \frac{k^2-1}{3} = \frac{2k^2+1}{3} = \frac{n^2+2}{6}, \quad \text{for } n = 2k.$$

(ii) n is odd. Let $n = 2k+1$, $k \in \mathbb{Z}^+$, $k > 1$. Then

$$\sum_{i=1}^{n-1}(-1)^{i+1}\frac{1}{1-\cos\frac{i\pi}{n}} = \sum_{i=1}^{k}\frac{1}{1-\cos\frac{(2i-1)\pi}{2k+1}} - \sum_{i=1}^{k}\frac{1}{1-\cos\frac{2i\pi}{2k+1}}.$$

From the definitions (B.21) and (B.22), it is trivial to show that

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{(2i-1)\pi}{2k+1}} = [\ln V_k]'(1), \tag{4.37}$$

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{2i\pi}{2k+1}} = [\ln W_k]'(1). \tag{4.38}$$

On the other hand, by the formula (B.13), we have

$$\begin{aligned}
[\ln V_k]'(1) &= \frac{U_k'(1) - U_{k-1}'(1)}{U_k(1) - U_{k-1}(1)} \\
&= \frac{\binom{k+1}{1}k + 2\binom{k+1}{3} - \binom{k}{1}(k-1) - 2\binom{k}{3}}{\binom{k+1}{1} - \binom{k}{1}} \\
&= k^2 + k,
\end{aligned} \tag{4.39}$$

$$\begin{aligned}
[\ln W_k]'(1) &= \frac{U_k'(1) + U_{k-1}'(1)}{U_k(1) + U_{k-1}(1)} \\
&= \frac{\binom{k+1}{1}k + 2\binom{k+1}{3} + \binom{k}{1}(k-1) + 2\binom{k}{3}}{\binom{k+1}{1} + \binom{k}{1}} \\
&= \frac{k^2 + k}{3},
\end{aligned} \tag{4.40}$$

that is,

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{(2i-1)\pi}{2k+1}} = k^2 + k, \tag{4.41}$$

$$\sum_{i=1}^{k} \frac{1}{1-\cos\frac{2i\pi}{2k+1}} = \frac{k^2 + k}{3}. \tag{4.42}$$

Hence,

$$\sum_{i=1}^{n-1} (-1)^{i+1} \frac{1}{1-\cos\frac{i\pi}{n}} = k^2 + k - \frac{k^2 + k}{3} = \frac{2(k^2 + k)}{3} = \frac{n^2 - 1}{6}. \tag{4.43}$$

$\square$

Lemma 4.4.4 *For $n \in \mathbb{Z}^+, n > 1$,*

$$\sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n} = 0, \quad n \text{ even}, \tag{4.44}$$

$$\sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n} = 1, \quad n \text{ odd}. \tag{4.45}$$

Proof. It is trivial to show that the formulae hold for $n = 2, 3$, so we only consider $n > 3$.

Let $\theta = \frac{\pi}{n}$, we have

$$\cos i\theta = 2\cos\theta\cos(i-1)\theta - \cos(i-2)\theta, \quad i = 3, \cdots, n.$$

Then,

$$(-1)^{i+1}\cos i\theta = 2(-1)^{i+1}\cos\theta\cos(i-1)\theta - (-1)^{i+1}\cos(i-2)\theta,\ i = 3, \cdots, n.$$

Hence,

$$\sum_{i=3}^{n-1}(-1)^{i+1}\cos i\theta = -2\cos\theta\sum_{i=2}^{n-2}(-1)^{i+1}\cos i\theta - \sum_{i=1}^{n-3}(-1)^{i+1}\cos i\theta.$$

Let $y = \sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n}$, then the left hand side of the above equation is

$$\sum_{i=3}^{n-1}(-1)^{i+1}\cos i\theta = y + \cos\frac{2\pi}{n} - \cos\frac{\pi}{n},$$

and the right hand side changes into

$$
\begin{aligned}
&-2\cos\theta\sum_{i=2}^{n-2}(-1)^{i+1}\cos i\theta-\sum_{i=1}^{n-3}(-1)^{i+1}\cos i\theta \\
&=-2\cos\frac{\pi}{n}\left[y-\cos\frac{\pi}{n}-(-1)^n\cos\frac{n-1}{n}\pi\right]-\left[y-(-1)^{n-1}\cos\frac{n-2}{n}\pi-(-1)^n\cos\frac{n-1}{n}\pi\right] \\
&=-2\cos\frac{\pi}{n}\left[y-\cos\frac{\pi}{n}+(-1)^n\cos\frac{\pi}{n}\right]-\left[y+(-1)^{n-1}\cos\frac{2\pi}{n}+(-1)^n\cos\frac{\pi}{n}\right] \\
&=\begin{cases} -(2\cos\frac{\pi}{n}+1)y+\cos\frac{2\pi}{n}-\cos\frac{\pi}{n}, & \text{when } i \text{ is even} \\ -(2\cos\frac{\pi}{n}+1)y+\cos\frac{2\pi}{n}+\cos\frac{\pi}{n}+2, & \text{when } i \text{ is odd} \end{cases}
\end{aligned}
$$

Thus, when n is even, we have

$$y+\cos\frac{2\pi}{n}-\cos\frac{\pi}{n}=-(2\cos\frac{\pi}{n}+1)y+\cos\frac{2\pi}{n}-\cos\frac{\pi}{n},$$

that is,

$$2(\cos\frac{\pi}{n}+1)y=0.$$

So

$$\sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n}=y=0,\ \text{when } n \text{ is even}$$

when n is odd, we have

$$2(\cos\frac{\pi}{n}+1)y=2(\cos\frac{\pi}{n}+1).$$

Then,

$$\sum_{i=1}^{n-1}(-1)^{i+1}\cos\frac{i\pi}{n}=y=1,\ \text{when } n \text{ is odd}$$

□

By substituting the results of Lemma 4.4.3 and 4.4.4 into (4.24), we have

$$3v_n+v_{n-1}=\frac{3}{4}n. \tag{4.46}$$

From (4.33) and (4.46), we get the equations (4.25) and (4.26).

Chapter 5

Preconditioned GMRES for Bi-Quartic SC Equations

We have discussed the preconditioned basic GMRES method for the Quartic SC equations arising from the one-dimensional harmonic problem. In this chapter, we discuss the solution of the Bi-Quartic SC equations arising from the discretization of the two-dimensional harmonic problem, referred to as biharmonic problem.

5.1 Biharmonic Problem with Dirichlet and Neumann Boundary Conditions

Consider the biharmonic equation described by the operator equation

$$\mathbf{L}u \equiv u_{xxxx} + 2u_{xxyy} + u_{yyyy} = g \quad \text{for} \quad (x, y) \in \Omega \equiv (0,1) \times (0,1), \tag{5.1}$$

subject to Dirichlet and Neumann boundary conditions

$$u = 0, \quad u_x = 0 \quad \text{on} \quad x = 0, x = 1, \quad \text{for} \quad 0 \leq y \leq 1 \tag{5.2}$$

$$u = 0, \quad u_y = 0 \quad \text{on} \quad y = 0, y = 1, \quad \text{for} \quad 0 \leq x \leq 1 \tag{5.3}$$

For the discretization of the biharmonic problem (5.1)-(5.3) we use a discretization strategy similar to the one used for the one-dimensional harmonic problem (1.7) with Dirichlet (1.8) and Neumann (1.9) boundary conditions.

Let $\Delta_x \equiv \{x_i = i/M, i = 0, \cdots, M\}$ and $\Delta_y \equiv \{y_j/N, j = 0, \cdots, N\}$ be uniform partitions of $(0,1)$ with step-sizes $h_x = \frac{1}{M}$ and $h_y = \frac{1}{N}$, respectively. Let $\tau_i^x = (x_{i-1} + x_i)/2$, $i = 1, \cdots, M$ and $\tau_j^y = (y_{j-1} + y_j)/2$, $j = 1, \cdots, N$ be the midpoints of the partitions Δ_x and Δ_y, respectively.

We denote by $\widetilde{S}_{\Delta_x}$ and $\widetilde{S}_{\Delta_y}$ the quartic spline spaces with respect to partitions Δ_x and Δ_y, respectively, constructed so that the splines satisfy the boundary conditions (5.2) and (5.3), respectively. We choose as basis functions for $\widetilde{S}_{\Delta_x}$ and $\widetilde{S}_{\Delta_x}$ the sets $\{\widetilde{\Phi}_i^x(x)\}_{i=1}^M$ and $\{\widetilde{\Phi}_j^y(y)\}_{j=1}^N$, respectively. These two sets are generated as $\{\widetilde{\Phi}_i(x)\}_{i=1}^N$ was generated in Chapter 2, from the model quartic spline function (2.1) through transformations as in (2.2) and adjustments as in (2.13).

Let $\widetilde{S}_{\Delta_{xy}} \equiv \widetilde{S}_{\Delta_x} \otimes \widetilde{S}_{\Delta_y}$ be the approximating space for the BVP (5.1)-(5.3). Note that any $u_\Delta \in \widetilde{S}_{\Delta_{xy}}$ satisfies the boundary conditions by construction. The set of basis functions for $\widetilde{S}_{\Delta_{xy}}$ is chosen to be the tensor product $\{\widetilde{\Phi}_i^x(x)\widetilde{\Phi}_j^y(y)\}_{i=1,j=1}^{M,N}$ of the basis functions of $\widetilde{S}_{\Delta_x}$ and $\widetilde{S}_{\Delta_y}$.

The Quartic SC method determines an approximation $u_\Delta \in S_{\Delta_{xy}}$ to u

that satisfies (5.1) on the set $T \equiv \{(\tau_i^x, \tau_j^y), i = 1, \cdots, M, j = 1, \cdots, N\}$ of collocation points, that is,

$$\mathbf{L}u_\Delta = g \quad \text{on} \quad T. \tag{5.4}$$

The linear equations resulting from (5.4), if ordered first bottom-up then left-to-right, give rise to an $MN \times MN$ linear system

$$A\bar{\theta} = \bar{g}, \tag{5.5}$$

where $\bar{g}$ is a vector of values of $g(x, y)$ at the collocation points, $\bar{\theta}$ is the vector of unknown coefficients (degrees of freedom) of the bi-quartic spline approximation u_Δ, and A is a block-pentadiagonal matrix with pentadiagonal $N \times N$ blocks, which takes the tensor product form

$$A \equiv Q_{DN4}^x \otimes Q_{DN0}^y + 2Q_{DN2}^x \otimes Q_{DN2}^y + Q_{DN0}^x \otimes Q_{DN4}^y, \tag{5.6}$$

where Q_{DNi}^x and Q_{DNi}^y have the same structures as Q_{DNi} in (2.16)-(2.18) do, $i = 0, 2, 4$. Note that the matrices with superscript x are $M \times M$, involving the x-direction grid size $1/h_x$, and those with superscript y are $N \times N$, involving the y-direction grid size $1/h_y$.

5.2 Preconditioner for Bi-Quartic SC Equations

In order to construct a preconditioner for the Bi-Quartic SC equations (5.5) arising from the biharmonic problem (5.1)-(5.3), we consider an auxiliary problem which differs from (5.1)-(5.3) only in some of the boundary conditions. It will be seen that, as in the 1D case, the Bi-Quartic SC equations arising from this auxiliary problem can serve as the preconditioner for the preconditioned restarted GMRES method in solving the equations (5.5).

Consider the BVP problem (5.1) with the following boundary conditions

$$u = 0, \quad u_{xx} = 0 \quad \text{on} \quad x = 0, x = 1, \quad \text{for} \quad 0 \le y \le 1 \tag{5.7}$$

$$u = 0, \quad u_{y} = 0 \quad \text{on} \quad y = 0, y = 1, \quad \text{for} \quad 0 \le x \le 1 \tag{5.8}$$

For this problem, we choose the same partitions Δ_x and Δ_y, and the spline collocation point set T, that are used in the discretization of (5.1)-(5.3). Since the boundary conditions (5.8) and (5.3) are the same, the quartic spline space with respect to the partition Δ_y is still $\widetilde{S}_{\Delta_y}$. Notice that the boundary conditions (5.7) along the x direction are same as those of the one-dimensional harmonic problem (2.3)-(2.5). Thus, we construct the quartic spline space with respect to Δ_x in a similar way as for (2.3)-(2.5).

The linear system resulting from the above discretization strategy is of the form

$$A'\bar{\theta}' = \bar{g}', \tag{5.9}$$

where $\bar{g}'$ is the vector of values of $g(x, y)$ at the collocation points, $\bar{\theta}'$ is the vector of unknown coefficients (degrees of freedom) of the bi-quartic spline approximation u'_Δ, and A' is a block-pentadiagonal matrix with pentadiagonal $N \times N$ blocks, which takes the tensor product form

$$A' \equiv Q_4^x \otimes Q_{DN0}^y + 2Q_2^x \otimes Q_{DN2}^y + Q_0^x \otimes Q_{DN4}^y, \tag{5.10}$$

where Q^y_{DNi}, $i = 0, 2, 4$ are same matrices as in (5.6), and Q^x_i has the same structures as Q_i in (2.8)-(2.10) do, $i = 0, 2, 4$. Note that the matrices with the superscript x are $M \times M$, involving the x-direction grid size $1/h_x$, and those with the superscript y are $N \times N$, involving the y-direction grid size $1/h_y$.

From [8] and [27], we know that every Q^x_i, $i = 0, 2, 4$ can be diagonalized by the inverse of the Discrete Sine Transform II (DST-II) [15] matrix S^M, thus, the matrix A' of (5.10) can be block-diagonalized by the inverse of $S^M \otimes I^N$. That is,

$$A' = ((S^M)^{-1} \otimes I^N)B(S^M \otimes I^N), \tag{5.11}$$

where

$$B \equiv \Lambda_4 \otimes Q^y_{DN0} + 2\Lambda_2 \otimes Q^y_{DN2} + \Lambda_0 \otimes Q^y_{DN4}, \tag{5.12}$$

and Λ_i, $i = 0, 2, 4$, are $M \times M$ diagonal matrices with the eigenvalues of Q^x_i, $i = 0, 2, 4$, respectively, on the diagonal. Note that the superscripts of S^M and I^N denote the respective orders. Also note that the block-diagonal matrix B consists of M pentadiagonal blocks of size $N \times N$.

In [5], an algorithm based on the Fast Fourier Transform (FFT) is presented for computing the solution to the two-dimensional quadratic SC equations. The algorithm applies Fast Sine Transforms II (FST-II) in one-dimension to block diagonalize the matrix, and solves the block-diagonal matrix by Gauss elimination. A similar algorithm can be applied to solve the equations (5.9). Before describing the algorithm, we adopt a convenient notation from

[5]. For any $MN \times 1$ vector g, let $g_{N\times M}$ denote an $N \times M$ matrix with entries the components of g laid out in N rows and M columns, column by column.

Algorithm FFTSC(M, N, $\bar{g}'$)

Step 1: Perform **FST-II** of size M to each of the N columns of $(\bar{g}'_{N\times M})^T$ to obtain $g^{(1)}_{M\times N} = S^M(g'_{N\times M})^T$, or equivalently, $g^{(1)} = (S^M \otimes I^N)g'$

Step 2: Solve the block-diagonal system $Bg^{(2)} = g^{(1)}$, where B is given in (5.12).

Step 3: Perform **iFST-II** of size M to each of the N columns of $(g^{(2)}_{N\times M})^T$ to obtain $\bar{\theta}'_{M\times N} = (S^M)^{-1}(g^{(2)}_{N\times M})^T$, or equivalently, $\bar{\theta}' = ((S^M)^{-1} \otimes I^N)g^{(2)} = ((S^M)^{-1} \otimes I^N)B^{-1}(S^M \otimes I^N)\bar{g}' = A'^{-1}\bar{g}'$.

The block-diagonal matrix B consists of M pentadiagonal blocks of size $N \times N$, therefore **Step 2** consists of solving M pentadiagonal systems of size $N \times N$, i.e., it requires approximately $20MN$ real single flops, of which $10MN$ are attributed to LU factorization and $10MN$ to back-and-forward substitutions. Hence, the FFTSC algorithm requires approximately $5MN\log_2(M)+20MN$ real single flops.

Note that the only difference between Algorithm FFTSC and the algorithm in [5] is that the block-diagonal matrix in [5] consists of tridiagonal blocks, while B in FFTSC consists of pentadiagonal blocks.

The fact that the equations (5.9) can be solved by the algorithm FFTSC based on FFT suggests that the matrix A' (5.10) can serve as the preconditioner when the preconditioned GMRES method is applied to equations

(5.5). As will be seen in the numerical results the resulting iterative method does not have convergence rate independent of the problem size, but the number of iterations increases sublinearly with the number of subintervals in one dimension. Therefore, we choose to use the preconditioned restarted GMRES method so that the number of past directions is controlled. The numerical results are presented in next chapter.

Chapter 6

Numerical Results

In this chapter, we present numerical results that demonstrate the convergence performance of the preconditioned GMRES method. Our numerical experiments consist of two parts: the preconditioned basic GMRES method for the Quartic SC equations, and the preconditioned restarted GMRES method for the Bi-Quartic SC equations.

6.1 GMRES for Quartic SC Equations

The sixth order Quartic SC method [27] is used for the discretization of the problems. The method proceeds in 3 steps: at each step a linear system is solved with the same matrix and different right side vectors. The right side vectors of the 2nd and 3rd steps are appropriate perturbations of the right sides of the previous steps.

The preconditioned basic GMRES method is used for solving the linear system at each step. The stopping criterion is the relative Euclidean norm of the residual and the tolerance is set to 10^{-9}. The initial guess for the solution is the zero vector. All experiments are implemented with MATLAB in double precision.

All problems considered involve the boundary conditions (1.8)-(1.9). The preconditioners arise from the same differential operator, but with the boundary conditions

$$u = g_1 \quad \text{on} \quad \partial\Omega, \tag{6.1}$$

$$u'' = g_2 \quad \text{on} \quad \partial\Omega. \tag{6.2}$$

We select some exact solution functions first, then determine the functions g, g_1, and g_2 from the differential equation, the boundary conditions, and the selected exact solutions. After getting g, g_1, and g_2, we compute the approximations.

Problem 1: Consider the problem (1.7)-(1.9) with an exact solution $u(x) = x^{19/2} - 3x^{17/2} + 3x^{15/2} - x^{13/2}$ on $\Omega = (0, 1)$.

Problem 1 has homogeneous boundary conditions, that is, g_1 and g_2 are all zero. Table 6.1 shows results from the application of the preconditioned basic GMRES method to the Quartic SC system arising from the discretization of Problem 1. The error and convergence results are presented to verify the performance of the sixth order Quartic SC method, as it is described in [27]. For each N shown in Table 6.1, the error "on grid points" corresponds to the maximum in absolute value error of the Quartic SC approximation on the discretization grid and is expected to be sixth order, while the "global" error corresponds to the error on an uniform grid of 1003 points, of which none is a mesh point, and is expected to be fifth order. The "global" error is taken as an approximation to the uniform norm of the error. The iteration numbers at each step agree with our theoretical analysis.

Problem 2: Consider the problem (1.7)-(1.9) with an exact solution $u(x) = e^x$ on $\Omega = (0, 3)$.

The boundary conditions for Problem 2 are non-homogeneous. As we said in Chapter 2, the system arising from the discretization of Problem 2 by the optimal Quartic SC method can also be solved by the preconditioned GMRES method. The numerical results are shown in Table 6.2.

Problem 3: Consider the problem (1.7)-(1.9) with an exact solution $u(x) = x^2(1-x)^2e^x$ on $\Omega = (0, 1)$.

The boundary conditions for Problem 3 are homogeneous. The numerical results are shown in Table 6.3.

Table 6.1: Errors on the grid points, respective orders of convergence, number of iterations for convergence of the preconditioned GMRES method, corresponding to Problem 1 discretized by the Quartic SC method, for several gridsizes N.

$u(x) = x^{19/2} - 3x^{17/2} + 3x^{15/2} - x^{13/2}$, $\Omega = (0, 1)$					
	Step 1				
	on grid points		global		number of
N	error	order	error	order	iterations
8	7.4247e-004		7.6927e-004		3
16	1.7394e-004	2.0937	1.7395e-004	2.1449	3
32	4.2043e-005	2.0487	4.2046e-005	2.0486	3
64	1.0416e-005	2.0131	1.0417e-005	2.0131	3
N	Step 2				
8	1.1550e-004		1.1550e-004		3
16	2.8993e-006	5.3160	2.9164e-006	5.3076	3
32	2.0302e-007	3.8360	2.0312e-007	3.8438	3
64	1.2698e-008	3.9989	1.2703e-008	3.9990	3
N	Step 3				
8	2.3242e-004		2.3468e-004		3
16	3.4653e-006	6.0676	3.4852e-006	6.0733	3
32	6.9343e-008	5.6431	7.0140e-008	5.6348	3
64	1.2666e-009	5.7748	1.2898e-009	5.7650	3

Table 6.2: Errors on the grid points, respective orders of convergence, number of iterations for convergence of the preconditioned GMRES method, corresponding to Problem 2 discretized by the Quartic SC method, for several gridsizes N.

$u(x) = e^x$, $\Omega = (0,3)$					
	Step 1				
	on grid points		global		number of
N	error	order	error	order	iterations
8	6.9247e-003		7.0183e-003		3
16	1.7261e-003	2.0042	1.7349e-003	2.0162	3
32	4.3244e-004	1.9970	4.3256e-004	2.0039	3
64	1.2097e-004	2.0009	1.0807e-004	2.0009	3
N	Step 2				
8	1.7643e-004		1.8596e-004		3
16	9.8807e-006	4.1583	1.0321e-005	4.1713	3
32	6.2069e-007	3.9927	6.3358e-007	4.0259	3
64	4.0088e-008	3.9526	4.0533e-008	3.9664	3
N	Step 3				
8	8.1680e-005		8.9792e-005		3
16	1.4661e-006	5.7999	1.7911e-006	5.6476	3
32	2.7067e-008	5.7593	3.8845e-008	5.5270	3
64	1.4042e-009	4.2687	1.9703e-009	4.3012	3

Table 6.3: Errors on the grid points, respective orders of convergence, number of iterations for convergence of the preconditioned GMRES method, corresponding to Problem 3 discretized by the Quartic SC method, for several gridsizes N.

$u(x) = x^2(1-x)^2 e^x$, $\Omega = (0,1)$					
	Step 1				
	on grid points		global		number of
N	error	order	error	order	iterations
8	1.0954e-003		1.1048e-003		3
16	2.6867e-004	2.0276	2.7049e-004	2.0301	3
32	6.7268e-005	1.9978	6.7271e-005	2.0075	3
64	1.6795e-005	2.0019	1.6796e-005	2.0019	3
N	Step 2				
8	2.9811e-005		3.1369e-005		3
16	1.8946e-006	3.9759	1.9523e-006	4.0061	3
32	1.1893e-007	3.9937	1.2088e-007	4.0135	3
64	7.4546e-009	3.9958	7.5139e-009	4.0079	3
N	Step 3				
8	5.8956e-006		7.5853e-006		3
16	1.1890e-007	5.6318	1.8012e-007	5.3961	3
32	2.1503e-009	5.7891	4.4070e-009	5.3530	3
64	3.8265e-011	5.8124	1.3549e-010	5.0235	3

6.2 GMRES for Bi-Quartic SC Equations

In this section, we present the numerical results that show the convergence performance of the preconditioned restarted GMRES method for solving the linear equations (5.5).

The experiment settings, such as the stopping criterion, the initial guess for the solution, and the system precision, are the same as for the preconditioned basic GMRES method for one-dimensional harmonic equations. The maximum number of previous iteration vectors that GMRES take into account in one iteration, i.e., the number to restart, is set to 20. The testing process is also same as the one-dimensional case, that is, we select some exact solution functions first that satisfy the boundary conditions (5.2)-(5.3), then determine the functions g from the selected exact solutions. From g, we can compute the approximations to these exact solutions. We consider the following problems:

Problem 4: Consider the problem (5.1)-(5.3) with an exact solution $u(x, y) = (x^{19/2} - 3x^{17/2} + 3x^{15/2} - x^{13/2})(y^{19/2} - 3y^{17/2} + 3y^{15/2} - y^{13/2})$ on $\Omega = (0, 1) \times (0, 1)$.

Problem 5: Consider the problem (5.1)-(5.3) with an exact solution $u(x, y) = (x^{13/2} - 2x^{11/2} + x^{9/2})(y^{13/2} - 2y^{11/2} + y^{9/2})$ on $\Omega = (0, 1) \times (0, 1)$.

Problem 6: Consider the problem (5.1)-(5.3) with an exact solution $u(x, y) = (7x^{10} - 8x^9 + x^2)(7y^{10} - 8y^9 + y^2)$ on $\Omega = (0, 1) \times (0, 1)$.

The experimental results of the preconditioned restarted GMRES method are shown in Table (6.4). It can be observed that the number of iterations increases sublinearly with the gridsizes N in one dimension, i.e. with the square root of the number of equations. To get a clearer picture about how the number of iterations varies with the size of the linear system, we also apply the preconditioned basic GMRES method to these problems. The results are shown in Table 6.5. Figure 6.1 suggests that the relation between the number of iterations and the size of the linear systems might be approximated with a function $i = \beta(\log n)^{\alpha}$, where i is the number of iterations, n is the size of the linear system, and α and β are two parameters.

- For Problem 4, $\alpha = 1.0739$, and $\beta = 2.5835$, so the function is $i = 2.5835(\log n)^{1.0739}$
- For Problem 5, $\alpha = 1.3194$, and $\beta = 1.8062$, so the function is $i = 1.8062(\log n)^{1.3194}$
- For Problem 6, $\alpha = 1.3194$, and $\beta = 1.8062$, so the function is $i = 1.8062(\log n)^{1.3194}$

Table 6.4: Number of iterations for convergence of the preconditioned restarted GMRES method for three biharmonic problems, discretized by the Bi-Quartic SC method, for several gridsizes $N \times N$

	Number of iterations for each problem		
N	Problem 4	Problem 5	Problem 6
8	12	12	12
16	16	17	17
32	21	26	28
64	30	37	38

Table 6.5: Number of iterations for convergence of the preconditioned basic GMRES method for three biharmonic problems, discretized by the Bi-Quartic SC method, for several gridsizes $N \times N$

	Number of iterations for each problem		
N	Problem 4	Problem 5	Problem 6
8	12	12	12
16	16	17	17
32	21	23	23
64	25	30	30

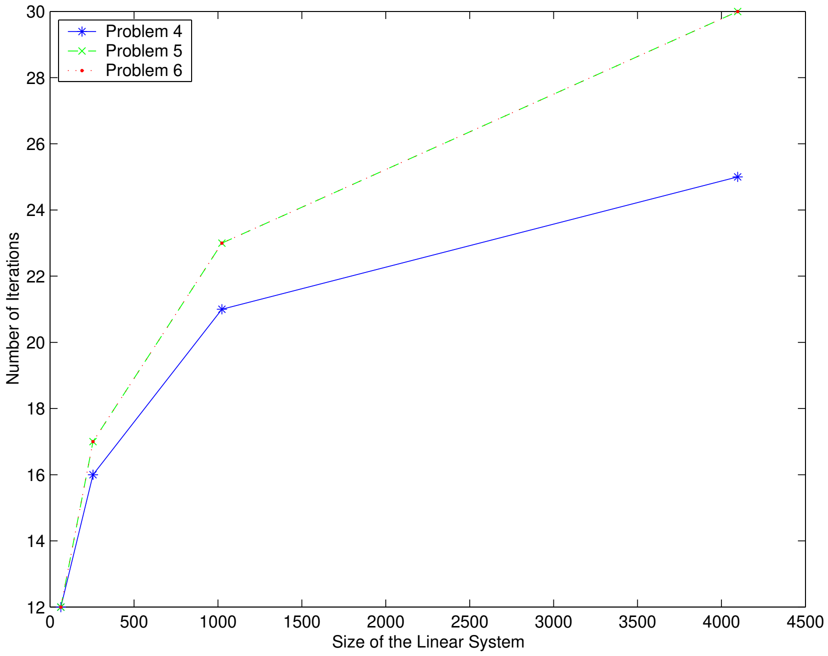

Figure 6.1: Plot of the numbers of iterations required for the solution of the Bi-Quartic SC equations for Problem 4, 5, and 6, with the preconditioned basic GMRES method versus the size of the linear system

Chapter 7

Conclusions and Future Work

In this paper, we present a Fast Fourier Transform (FFT)-based solver for the solution of the system arising from a sixth order discretization method for the biharmonic Dirichlet problem. In order to motivate our solution technique, particular attention is given to the one-dimensional harmonic problem.

The problem is discretized by using the collocation methodology and quartic splines. We apply an adjustment to the quartic spline basis functions so that the Quartic SC matrix has more favorable properties. We introduce the GMRES method, an iterative method for solving linear systems $Ax = b$, where A is a nonsymmetric matrix. Three different versions of GMRES algorithms, namely, the basic GMRES, the preconditioned basic GMRES, and the preconditioned restarted GMRES, are presented. In particular, for the basic GMRES method, we prove two theorems which relate its convergence performance with the eigenvalue distribution of the matrix A. We study the properties of two Quartic SC matrices, more specifically, one that corresponds to Dirichlet and Neumann conditions (the standard harmonic problem) and the other one that corresponds to Dirichlet and second derivative conditions. The latter is solvable by FFTs and used as the preconditioner for the first. We develop formulae for the eigenvalues and eigenvectors of the preconditioned matrix. With the help of the two theorems on the convergence properties of GMRES, we show that three iterations suffice to obtain convergence for the preconditioned GMRES method. Numerical experiments verify the effectiveness of the solver and preconditioner.

We also apply the preconditioned restarted GMRES method to the Bi-Quartic SC equations arising from the two-dimensional biharmonic Dirichlet problem. Although the numerical results show that the convergence performance is promising, a precise theoretical analysis about the convergence properties is still needed. Therefore, for possible future research, the distribution of the eigenvalues of the preconditioned Bi-Quartic SC matrix can be studied, so that the convergence of the GMRES method can be predicted.

Another issue that needs to be mentioned is that, unlike the preconditioned basic GMRES for the one-dimensional harmonic problem, the preconditioned restarted GMRES method for the two-dimensional biharmonic problem is not able to identify the solution in number of iterations independent of the problem size. In [3], Bjørstad proposes an algorithm that uses a thirteen-point finite difference stencil and the preconditioned conjugate gradient method for the solution of the biharmonic Dirichlet problem. The number of iterations of PCG is independent of the matrix size. The basic idea of Bjørstad's algorithm is to reduce the solution of the linear system arising from the thirteen-point finite difference discretization to the solution of two auxiliary biharmonic problems that can be solved by $O(N^2)$ solvers and two one-dimensional problems along two opposite sides of the domain. The latter are solved by the preconditioned conjugate gradient method in number of iterations independent of N. However, the discretization method used in [3] is standard second order finite differences. Therefore, another interesting future problem is to explore whether the Bi-Quartic SC equations, which give sixth order approximations, can be solved by a technique similar to the technique in [3].

Appendix A

Centrosymmetric Matrix

An elegant discussion about *centrosymmetric matrices* can be found in [24].

Definition A.0.1 *An $n \times n$ matrix P over the real field is said to be centrosymmetric if*

$$P_{i,j} = P_{n-i+1,\,n-j+1},\ \textit{for } 1 \leq i,\ j \leq n.$$

There are two propositions about centrosymmetric matrices [24].

Proposition A.0.2 *A matrix P is centrosymmetric if and only if $P = JPJ$, where J is the permutation matrix with ones on the cross diagonal (bottom left to top right) and zeros elsewhere.*

Proposition A.0.3 *If P is a nonsingular centrosymmetric matrix, then P^{-1} and P^T are also centrosymmetric; if A and B are both centrosymmetric, then AB is centrosymmetric too.*

The proofs of these propositions are presented in [24].

Appendix B

Chebyshev Polynomials

J.C. Mason and D.C. Handscomb present detailed discussions on different topics of Chebyshev polynomials in their book [17]. Here we only introduce some definitions that are needed for our analysis.

B.1 Trigonometric definitions and recurrences

Definition B.1.1 *The first-, second-, third- and fourth-kind Chebyshev polynomials $T_n(x)$, $U_n(x)$, $V_n(x)$ and $W_n(x)$ are polynomials of degree n in x defined, respectively, by*

$$T_n(x) = \cos(n\theta) \tag{B.1}$$

$$U_n(x) = \sin(n\theta)/\sin(\theta) \tag{B.2}$$

$$V_n(x) = \cos(n+\frac{1}{2})\theta/\cos\frac{1}{2}\theta \tag{B.3}$$

$$W_n(x) = \sin(n+\frac{1}{2})\theta/\sin\frac{1}{2}\theta \tag{B.4}$$

when $x = \cos\theta$.

With Definition B.1.1, we obtain the fundamental recurrence relations

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \tag{B.5}$$

$$U_n(x) = 2xU_{n-1}(x) - U_{n-2}(x) \tag{B.6}$$

$$V_n(x) = 2xV_{n-1}(x) - V_{n-2}(x) \tag{B.7}$$

$$W_n(x) = 2xW_{n-1}(x) - W_{n-2}(x) \tag{B.8}$$

for $n = 2,\ 3, \ldots$.

Two further relationships that may be derived from the definitions are

$$V_n(x) = U_n(x) - U_{n-1}(x) \tag{B.9}$$

$$W_n(x) = U_n(x) + U_{n-1}(x) \tag{B.10}$$

where $n > 1$.

B.2 Definitions in terms of sums

The polynomials $T_n(x)$ and $U_n(x)$ can also defined in terms of the sums ([25], [26])

$$T_n(x) = \sum_{m=0}^{\lfloor n/2 \rfloor} \binom{n}{2m} x^{n-2m}(x^2-1)^m. \tag{B.11}$$

$$U_n(x) = \sum_{m=0}^{\lceil n/2 \rceil} \binom{n+1}{2m+1} x^{n-2m}(x^2-1)^m \tag{B.12}$$

Then, we deduce that $V_n(x)$ and $W_n(x)$ can also be written as

$$\begin{aligned} V_n(x) &= U_n(x) - U_{n-1}(x) \\ &= \sum_{m=0}^{\lceil n/2 \rceil} \binom{n+1}{2m+1} x^{n-2m}(x^2-1)^m - \sum_{m=0}^{\lceil (n-1)/2 \rceil} \binom{n}{2m+1} x^{n-1-2m}(x^2-1)^m, \end{aligned} \tag{B.13}$$

and

$$\begin{aligned} W_n(x) &= U_n(x) + U_{n-1}(x) \\ &= \sum_{m=0}^{\lceil n/2 \rceil} \binom{n+1}{2m+1} x^{n-2m}(x^2-1)^m - \sum_{m=0}^{\lceil (n-1)/2 \rceil} \binom{n}{2m+1} x^{n-1-2m}(x^2-1)^m. \end{aligned} \tag{B.14}$$

B.3 Chebyshev polynomial zeros

From the formula (B.1), the zeros for x in $[-1,1]$ of $T_n(x)$ have to correspond to the zeros for θ in $[0,\pi]$ of $\cos n\theta$, which are

$$\theta_i = \frac{2i-1}{2n}\pi, \quad i = 1,\ 2, \ldots, n.$$

Hence, the zeros of $T_n(x)$ are

$$x = \cos \frac{2i-1}{2n}\pi, \quad i = 1,\ 2, \ldots, n. \tag{B.15}$$

The zeros of $U_n(x)$, $V_n(x)$ and $W_n(x)$ can be determined in a similar way. The zeros of $U_n(x)$ occur at

$$x = \cos \frac{i\pi}{n+1}, \quad i = 1,\ 2, \ldots, n. \tag{B.16}$$

and the zeros of $V_n(x)$ occur at

$$x = \cos \frac{2i-1}{2n+1}\pi, \quad i = 1,\ 2, \ldots, n. \tag{B.17}$$

while the zeros of $W_n(x)$ occur at

$$x = \cos \frac{2i\pi}{2n+1}, \quad i = 1,\ 2, \ldots, n. \tag{B.18}$$

B.4 Definitions in terms of products

It is known that the x^n coefficients of $T_n(x)$ is 2^{n-1}, while the x^n coefficients of $U_n(x)$, $V_n(x)$ and $W_n(x)$ are all 2^n [17].

With the zeros and x^n coefficients, we can easily obtain the polynomials' definitions in terms of products

$$\begin{aligned}
T_n(x) &= 2^{n-1} \prod_{i=1}^{n} \left[x - \cos \frac{(2i-1)\pi}{2n} \right], && \text{(B.19)} \\
U_n(x) &= 2^{n} \prod_{i=1}^{n} \left[x - \cos \frac{i\pi}{n+1} \right], && \text{(B.20)} \\
V_n(x) &= 2^{n} \prod_{i=1}^{n} \left[x - \cos \frac{(2i-1)\pi}{2n+1} \right], && \text{(B.21)} \\
W_n(x) &= 2^{n} \prod_{i=1}^{n} \left[x - \cos \frac{2i\pi}{2n+1} \right] && \text{(B.22)}
\end{aligned}$$

Bibliography

[1] Irfan Altas, Jonathan Dym, Murli M. Gupta, and Ram P. Manohar. Multigrid solution of automatically generated high-order discretizations for the biharmonic equation. *SIAM Journal on Scientific Computing*, 19(5):1575–1585, 1998.

[2] Bernard Bialecki. A fast solver for the orthogonal spline collocation solution of the biharmonic Dirichlet problem on rectangles. *Journal of Computational Physics*, 191:601–621, 2003.

[3] P. Bjørstad. Fast numerical solution of the biharmonic Dirichlet problem on rectangles. *SIAM Journal on Numerical Analysis*, 20:59–71, 1983.

[4] B. L. Buzbee and F. W. Dorr. The direct solution of the biharmonic equation on rectangular regions and the Poisson equation on irregular regions. *SIAM Journal on Numerical Analysis*, 11:753–763, 1974.

[5] Christina C. Christara and Kit Sun Ng. Fast Fourier transform solvers and preconditioners for quadratic spline collocation. *BIT*, 42(4):702–739, 2002.

[6] P.G. Ciarlet and R. Glowinski. Dual iterative techniques for solving a finite element approximation of the biharmonic equation. *Computational Methods in Applied Mechanics and Engineering*, 5:277–295, 1975.

[7] P.G. Ciarlet and P. A. Raviart. A mixed finite element method for the biharmonic equation. In C. de Boor, editor, *Proceedings of Symposium on Mathematical Aspects of Finite Elements in PDE*, pages 125–145. Academic Press, 1974.

[8] Athanasia Constas. Fast Fourier transform solvers for quadratic spline collocation. Master's thesis, University of Toronto, Toronto, Ontario, Canada, 1996.

[9] K. D. Cooper and P. M. Prenter. A coupled double slitting ADI scheme for first biharmonic using collocation. *Numerical Methods for Partial Differential Equations*, 6:321–333, 1990.

[10] L. W. Ehrlich. Solving the biharmonic equation as coupled finite difference equations. *SIAM Journal on Numerical Analysis*, 8:278–287, 1971.

[11] L. W. Ehrlich. Solving the biharmonic equation in a square: a direct versus semidirect method. *Communications of the ACM*, 16(11):711–714, 1973.

[12] R. Glowinski and O. Pironneau. Numerical methods for the first biharmonic equation and for the two-dimensional Stokes problem. *SIAM Review*, 21:167–212, 1979.

[13] D. B. Knudson. *A piecewise Hermite bicubic finite element Galerkin method for the biharmonic Dirichlet problem.* PhD thesis, Colorado School of Mines, Golden, Colorado, U.S.A., 1997.

[14] Y. Kwon, R. Manohar, and J. W. Stephenson. Single cell fourth order methods for the biharmonic equation. *Congressus Numerantium*, 34:475–482, 1982.

[15] C. Van Loan. *Computational Frameworks for the Fast Fourier Transform.* Society for Industrial and Applied Mathematics, 1992.

[16] Zhou-Ming Lou, Bernard Bialecki, and Graeme Fairweather. Orthogonal spline collocation methods for biharmonic problems. *Numerische Mathematik*, 80:267–303, 1998.

[17] J.C. Mason and D.C. Handscomb. *Chebyshev Polynomials.* Chapman & Hall/CRC, 2003.

[18] J. W. McLaurin. A general coupled equation approach for solving the biharmonic boundary value problem. *SIAM Journal on Numerical Analysis*, 11:14–33, 1974.

[19] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization.* Springer, 1999.

[20] P. Peisker. On the numerical solution of the first biharmonic equation. *Mathematical Modelling and Numerical Analysis*, 22:655–676, 1988.

[21] Yousef Saad. *Iterative methods for sparse linear systems.* PWS, 1996.

[22] J. W. Stephenson. Single cell discretizations of order two and four for biharmonic problems. *Journal of Computational Physics*, 55:65–80, 1984.

[23] W. Sun. Orthogonal collocation solution of biharmonic equations. *International Journal of Computer Mathematics*, 49:21–232, 1993.

[24] James R. Weaver. Centrosymmetric(cross-symmetric) matrices, their basic properties, eigenvalues, and eigenvectors. *The American Mathematical Monthly*, 92(10):711–717, December 1985.

[25] Eric W. Weisstein. Chebyshev polynomial of the first kind. *MathWorld.* `http://mathworld.wolfram.com/ChebyshevPolynomialoftheFirstKind.html`.

[26] Eric W. Weisstein. Chebyshev polynomial of the second kind. *MathWorld.* `http://mathworld.wolfram.com/ChebyshevPolynomialoftheSecondKind.html`.

[27] Ying Zhu. Quartic-spline collocation methods for fourth-order two-point boundary value problems. Master's thesis, University of Toronto, Toronto, Ontario, Canada, 2001.