2022 HAWAII UNIVERSITY INTERNATIONAL CONFERENCES
SCIENCE, TECHNOLOGY & ENGINEERING, ARTS, MATHEMATICS & EDUCATION JUNE 7 - 9, 2022
PRINCE WAIKIKI RESORT, HONOLULU, HAWAII

PRECONDITIONED ITERACTIVE SOLVERS FOR THE 2D HELMHOLTZ EQUATION VIA RADIAL BASIS FUNCTIONS

LIU, JIA
CARBREANA, JOHNNY
KUO, LIE HSIN
DEPARTMENT OF MATHEMATICS AND STATISTICS
UNIVERSITY OF WEST FLORIDA
PENSACOLA, FLORIDA
WU, LINA
DEPARTMENT OF MATHEMATICS
BOROUGH OF MANHATTAN COMMUNITY COLLEGE
THE CITY UNIVERSITY OF NEW YORK
NEW YORK, NEW YORK

Preconditioned Iterative Solvers for the 2D Helmholtz Equation Via Radial Basis Functions

Johnny Carbreana[a], Lie Ksin Kuo[a], Lina Wu[b], Jia Liu[a]

[a] *Department of Mathematics and Statistics, University of West Florida, Pensacola 32514, U.S.A.*
[b] *Department of Mathematics, Borough of Manhattan Community College, The City University of New York, New York 32514, U.S.A.*

Abstract

In this paper, we discuss the solution solver for the linear system of equations arising from the discretized 2D Helmholtz equation using the radial basis functions. The coefficient matrix A that is generated from the discretization is dense and often ill-conditioned. This paper uses preconditioned iterative methods such as the General Minimal Residual method (GMRES) to solve the linear system. Different preconditioners are compared. Numerical experiments are conducted in order to provide a comparison of the convergence rates among various preconditioned linear systems. A further observation is made into the eigenvalue distributions and the choice of the shaping parameters during the meshless discretization. We conclude the triangular preconditioner and the diagonal preconditioner are both good choices for the discretized 2D Helmholtz equation when we use the meshless discretizations.

Keywords: Iterative Methods, Preconditioning, Helmholtz equations, Radial Basis Functions

Email addresses: jcarbreana@uwf.edu (Johnny Carbreana), lkuo@uwf.edu (Lie Ksin Kuo), lwu@bmcc.cuny.edu (Lina Wu), jliu@uwf.edu (Jia Liu)

1. Introduction

Consider the following 2D steady-state Helmholtz in homogeneous Boundary Value Problem (BVP) with Dirichlet boundary conditions

$$\Delta u(x,y) + k^2 u(x,y) = f(x,y), \quad x,y \in \Omega = [0,1]^2, \tag{1}$$

$$u(x,y) = g(x,y), \quad x,y \in \partial\Omega = [0,1]^2. \tag{2}$$

Where $f(x,y)$ represents the forcing term and $g(x,y)$ represents the Dirichlet boundary condition. The parameter $k = \frac{\omega}{c}$ is called the *wavenumber*. Here $\omega = 2\pi f$ is the angular frequency, f is the frequency in hertz, and c is the local speed of sound (in m/s). The Helmholtz equation has applications in many important fields, such as in aeroacoustics, under-water acoustics, etc. In addition, the Helmhotz equation (1) is also used to describe electromagnetics, where it can be derived from Maxwell's equation; see [6].

Formulated by Hermann Ludwig Ferdinand von Helmholtz, the Helmholtz equation results from applying the separation of variables method to the wave equation. Consider the wave equation below

$$\Delta\Psi(\vec{x},t) = \frac{1}{c^2}\frac{\partial^2\Psi(\vec{x},t)}{\partial t^2}, \tag{3}$$

where $\Psi(\vec{x},t)$ denotes the wave function and c represents the wave velocity. Setting equation (1.3) equal to zero and factoring out the wave function, we arrive to the following

$$\Big(\Delta - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\Big)\Psi(\vec{x},t) = 0. \tag{4}$$

Let us assume that the wave function is separable, i.e. $\Psi(\vec{x},t) = A(\vec{x})\,T(t)$. Substituting this into equation (1.4) and simplifying, we obtain

$$\frac{\Delta A(\vec{x})}{A(\vec{x})} = \frac{1}{c^2 T(t)}\frac{\partial^2 T(t)}{\partial t^2}. \tag{5}$$

Notice that the expression on the left depends solely on and the expression on the right depends on t. Therefore, equation (1.5) is valid if and only if both sides

of the equation are equal to the same constant value. Let us choose the constant value to be the expression $-k^2$, so that we have the following two equations

$$\frac{\Delta A(\vec{x})}{A(\vec{x})} = -k^2, \tag{6}$$

and

$$\frac{1}{c^2 T(t)} \frac{\partial^2 T(t)}{\partial t^2} = -k^2. \tag{7}$$

Rearranging equations (1.6) and (1.7), we obtain the Helmholtz equation and a second-order ODE in time respectively

$$\Delta A(\vec{x}) + k^2 A(\vec{x}) = 0, \tag{8}$$

$$\frac{\partial^2 T(t)}{\partial t^2} + k^2 c^2 T(t) = 0. \tag{9}$$

Clearly we can see that the Helmholtz equation represents a time-independent form of the wave equation.

2. Discretization

It is common to use a five point finite difference discretization or a finite element discretization for (1) and (2). In this paper, we will use another discretization which is known as the meshless method. dsMeshless methods have gained much attention in recent years. Independence from a mesh is a great advantage since mesh generation is one of the most time consuming parts of any mesh-based numerical simulation. Element-free Galerkin method (EFG), meshless local Petrov-Galerkin method (MLPG), and smoothed particle hydrodynamics (SPH) are all examples of meshless methods. In this paper we will discuss a meshfree approach for discretization using radial basis functions. Let R^n be n-dimensional Euclidean space. A function $\Phi : R^n \to R$ is called a radial basis function if

$$\Phi(\vec{x}) = \Phi(\vec{y}), \quad \text{whenever} \quad \|\vec{x}\| = \|\vec{y}\|, \quad \vec{x}, \vec{y} \in R^n, \tag{10}$$

where $\|\cdot\|$ represents the Euclidean norm on R^n [5].

Ed Kansa proposed a non-symmetric method for the solution of elliptic PDEs utilizing RBFs in the 1990s. In the context of scattered data interpolation we are given data $\{\vec{x}_i, f_i\}$, $i = 1, \ldots, N$, $\vec{x}_i \in R^n$, where we can think of the values f_i being sampled from a function $f : R^n \to R$. The goal is to find an interpolant of the form

$$\mathcal{P}_f(\vec{x}) = \sum_{j=1}^{N} \alpha_j \Phi\left(\|\vec{x} - \vec{x}_j\|_2\right), \quad \vec{x} \in R^n, \tag{11}$$

such that

$$\mathcal{P}_f(\vec{x}_i) = f_i, \quad i = 1, \ldots, N. \tag{12}$$

The solution of this problem leads to a linear system $A\alpha = f$ with entries of A given by

$$A_{ij} = \Phi\left(\|\vec{x}_i - \vec{x}_j\|_2\right), \quad i, j = 1, \ldots, N. \tag{13}$$

We now switch to the collocation solution of partial differential equations. Assume we are given a domain $\Omega \subset R^n$, and a linear elliptic partial differential equation with Dirichlet boundary conditions of the form

$$\mathcal{L}u(\vec{x}) = f(\vec{x}), \quad \vec{x} \in \Omega, \tag{14}$$

$$u(\vec{x}) = g(\vec{x}), \quad \vec{x} \in \partial\Omega. \tag{15}$$

For Kansa's collocation method, we choose to represent the approximate solution $\tilde{u}$ by a radial basis function expansion analogous to that used for scattered data interpolation

$$\tilde{u}(\vec{x}) = \sum_{j=1}^{N} \alpha_j \Phi\left(\|\vec{x} - \xi_j\|_2\right), \quad \vec{x} \in R^n, \tag{16}$$

where $\Xi = \{\xi_1, \ldots, \xi_N\}$ act as *centers* and $\chi = \{\vec{x}_1, \ldots, \vec{x}_N\}$ are the *collocation points* of the RBF numerical solution [5]. The collocation matrix that arises

when matching the differential equation and the boundary condition at the collocation points is of the form

$$A = \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix}, \tag{17}$$

where the two block matrices $\tilde{A}_{\mathcal{L}}$ and $\tilde{A}$ are created as follows

$$\left(\tilde{A}_{\mathcal{L}}\right)_{ij} = \mathcal{L}\Phi\left(\|\vec{x}_i - \xi_j\|_2\right), \quad \vec{x}_i \in \mathcal{I},\, \xi_j \in \Xi, \tag{18}$$

$$\tilde{A}_{ij} = \Phi\left(\|\vec{x}_i - \xi_j\|_2\right), \quad \vec{x}_i \in \mathcal{B},\, \xi_j \in \Xi. \tag{19}$$

The total dataset χ of collocation points is split into a set $\mathcal{I}$ of interior points and a set $\mathcal{B}$ of boundary points. The problem is well-posed if the linear system $A\alpha = \vec{y}$, with $\vec{y}$ a vector consisting of entries $f\left(\vec{x}_i\right),\, \vec{x}_i \in \mathcal{I}$, followed by $g\left(\vec{x}_i\right),\, \vec{x}_i \in \mathcal{B}$, has an unique solution [5]. In this paper we will use Kansa's approach with Hardy's MultiQuadratic (MQ) defined below

$$\Phi\left(r\right) = \sqrt{r^2 + c^2}, \tag{20}$$

where r represents the radial distance and c denotes the shape parameter. Using the following equations (1.21) and (1.22) for the interior and boundary points respectively

$$\sum_{j=1}^{N} \left(\frac{\partial^2 \Phi\left(r_j\right)}{\partial x^2} + \frac{\partial^2 \Phi\left(r_j\right)}{\partial y^2} \right) \alpha_j = f\left(x_i, y_i\right), \quad i = 1, \ldots, N_i, \tag{21}$$

$$\sum_{j=1}^{N} \Phi\left(r_j\right) \alpha_j = g\left(x_i, y_i\right), \quad i = N_i + 1, \ldots, N, \tag{22}$$

leads to the discretized linear system $A\alpha = \vec{y}$ shown below

$$\begin{bmatrix} \mathcal{L}\Phi\left(\|\vec{x}_1 - \xi_1\|_2\right) & \mathcal{L}\Phi\left(\|\vec{x}_1 - \xi_2\|_2\right) & \cdots & \mathcal{L}\Phi\left(\|\vec{x}_1 - \xi_N\|_2\right) \\ \vdots & \vdots & \vdots & \vdots \\ \mathcal{L}\Phi\left(\|\vec{x}_{N_i} - \xi_1\|_2\right) & \mathcal{L}\Phi\left(\|\vec{x}_{N_i} - \xi_2\|_2\right) & \dots & \mathcal{L}\Phi\left(\|\vec{x}_{N_i} - \xi_N\|_2\right) \\ \Phi\left(\|\vec{x}_{N_{i+1}} - \xi_1\|_2\right) & \Phi\left(\|\vec{x}_{N_{i+1}} - \xi_2\|_2\right) & \cdots & \Phi\left(\|\vec{x}_{N_{i+1}} - \xi_N\|_2\right) \\ \vdots & \vdots & \vdots & \vdots \\ \Phi\left(\|\vec{x}_N - \xi_1\|_2\right) & \Phi\left(\|\vec{x}_N - \xi_2\|_2\right) & \cdots & \Phi\left(\|\vec{x}_N - \xi_N\|_2\right) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_{N_i} \\ \alpha_{N_{i+1}} \\ \vdots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} f\left(\vec{x}_1\right) \\ \vdots \\ f\left(\vec{x}_{N_i}\right) \\ g\left(\vec{x}_{N_{i+1}}\right) \\ \vdots \\ g\left(\vec{x}_N\right) \end{bmatrix}. \tag{23}$$

3. Iterative Solvers and Preconditioners

After discretization of (1), we obtain the linear system

$$A\mathbf{u} = \mathbf{f}$$

where A is an N-by-N matrix, $\mathbf{u}$ is the solution and $\mathbf{f}$ is the right hand side. Direct methods such as Gaussian Elimination is one of the oldest and most popular solvers for the linear system above. Direct methods compute an approximate solution to a problem in a finite amount of steps.The drawback with direct solvers is the superlinear complexities they have in time and storage requirements [6]. Direct methods ensure higher accuracy but this in turns produces higher cost.

In contrast to direct solvers, iterative solvers solve the solution approximately with a relative lower accuracy. Iterative solvers have become a core area of research in numerical analysis [6]. The central issue in designing an iterative solver is finding an approximation of A^{-1}. It is particularly difficult to design a fast iterative solver for the Helmholtz equation; see review papers [1, 2, 4]. To tackle the special difficulties of the Helmholtz equation, many techniques have been developed and integrated into three major frameworks: incomplete factorization, (algebraic) multigrid, and domain decomposition [6].

A general orthogonal projection method for solving the linear system

$$A\vec{x} = \vec{b}, \tag{24}$$

retrieves an approximate solution $\vec{x}_n$ from a subspace $\vec{x}_0 + \mathcal{K}_n$ of dimension n by applying the Petrov-Galerkin condition

$$\vec{b} - A\vec{x}_n \perp \mathcal{L}_n, \tag{25}$$

where $\mathcal{L}_n$ is another subspace of dimension n and $\vec{x}_0$ represents an arbitrary initial guess to the solution [7]. Here, $\mathcal{K}_n$ represents the Krylov subspace which is defined as

$$\mathcal{K}_n\left(A, \vec{r}_0\right) = span\{\vec{r}_0, A\vec{r}_0, A^2\vec{r}_0, \ldots, A^{n-1}\vec{r}_0\}, \tag{26}$$

where $\vec{r}_0 = \vec{b} - A\vec{x}_0$ is the residual vector associated with $\vec{x}_0$. The approximations obtained from a Krylov subspace method are of the form

$$A^{-1}\vec{b} \approx \vec{x}_n = \vec{x}_0 + q_{n-1}\left(A\right)\vec{r}_0, \tag{27}$$

where q_{n-1} is a polynomial of degree $n-1$ [7]. If $\vec{x}_0 = 0$, then

$$A^{-1}\vec{b} \approx q_{n-1}\left(A\right)\vec{b}. \tag{28}$$

Unfortunately most iterative algorithms converge slowly and the central issue in designing an good iterative solver is to find a good approximation of the coefficient matrix A, which is called a *preconditioner*. A preconditioner is any form of implicit or explicit modification of an original linear system which makes it simpler to solve by a given iterative method [7]. Suppose we wish to solve an $n \times n$ nonsingular linear system

$$A\vec{x} = \vec{b}. \tag{29}$$

For any nonsingular $n \times n$ matrix M, the system

$$M^{-1}A\vec{x} = M^{-1}\vec{b}, \tag{30}$$

has the same solution. For the idea of preconditioning to be useful, it must be possible to compute the operation represented by the product $M^{-1}A$ efficiently.

Equation (2.24) describes the idea behind preconditioning the system on the left side. If we would like to precondition the system on the right side

$$AM^{-1}\vec{y} = \vec{b}, \quad \vec{x} = M^{-1}\vec{y}, \tag{31}$$

or, if M is available in split form $M = M_1 M_2$, to use

$$M_1^{-1} A M_2^{-1} \vec{y} = M_1^{-1} \vec{b}, \quad \vec{x} = M_2^{-1} \vec{y}. \tag{32}$$

In this paper, three different preocnditioners are considered:

3.1. Diagonal Preconditioning of A

Let preconditioner M_1 be the diagonal elements of A as shown below

$$M_1 = \begin{bmatrix} a_{11} & & \\ & \ddots & \\ & & a_{nn} \end{bmatrix}. \tag{33}$$

Where $a_{i,j}$ are the diagonal elements of the coefficient matrix.

3.2. Upper Triangular Preconditioning of A

Let preconditioner M_2 be the upper triangular elements of A as shown below

$$M_2 = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ & \ddots & \vdots \\ & & a_{nn} \end{bmatrix}. \tag{34}$$

3.3. Incomplete LU Preconditioning of A

Let preconditioner M_3 be the incomplete LU factorization of A as shown below

$$M_3 = \begin{bmatrix} \hat{l}_{11} & & \\ \vdots & \ddots & \\ \hat{l}_{n1} & \cdots & \hat{l}_{nn} \end{bmatrix} \begin{bmatrix} \hat{u}_{11} & \cdots & \hat{u}_{1n} \\ & \ddots & \vdots \\ & & \hat{u}_{nn} \end{bmatrix}. \tag{35}$$

An ILU preconditioner can be constructed by performing Gaussian elimination and removing elements based on certain criteria [3]. Removing all elements except for those in the same diagonals as the original matrix, describes the ILU(0) preconditioner. ILU(p) allows fill-in for p additional diagonals. Removing elements smaller than a specified value, lends itself to the ILU(tol). [2] provides

a discussion on preconditioners involving Algebraic ILU and Analytical ILU. In applications involving M-matrices, this class of preconditioners is effective. However, these preconditioners are not effective for general indefinite problems [3].

4. Numerical Experiments

Consider the following 2D Helmholtz in homogeneous boundary value problem (BVP) with Dirichlet boundary conditions (1) and (2). Let $f(x, y)$ represent the right hand side function and $g(x, y)$ represent the analytical solution defined below

$\mathrm{f}(x, y) = \pi^2 \sinh(y) \cos(\pi x) - \pi^2 \cosh(y) \sin(\pi x) + \cosh(y) \sin(\pi x) - \sinh(y) \cos(\pi x) + k^2 (\sin(\pi x) \cosh(y) - \cos(\pi x) \sinh(y))$,

$$g(x, y) = \sin(\pi x) \cosh(y) - \cos(\pi x) \sinh(y). \tag{36}$$

Numerical simulations utilizing GMRES are conducted with constant wave number $k = 10$ and constant shape parameter $c = 0.01$. We choose the desired tolerance to be 10^{-6} and the maximum number of iterations to be 1000. The number of the discretization grid are given by N. FLAG 1 means the divergence and FLAG 0 means convergence. RELERES denotes the residual and ITER denotes the iterative number.

Table 1: **Diagonal Preconditioner**

N	**FLAG**	**RELRES**	**ITER**
10	1	0.4646	1000
100	1	0.7194	1000
200	1	0.7529	1000
400	1	0.7911	876

Table 2: **Upper Triangular Preconditioner**

N	FLAG	RELRES	ITER
10	0	1.0079E-14	1
100	0	7.2723E-16	1
200	0	1.0760E-15	1
400	0	1.6820E-16	1

Table 3: **ILU Factorization with setup.droptol = 0.5**

N	FLAG	RELRES	ITER
10	0	3.5319E-14	7
100	0	7.0764E-07	42
200	0	8.8037E-07	57
400	0	8.1610E-07	76

Table1 -3 shows the numerical experiments of the preconditioned GMRES methods with different preconditioners. Table 1 is the iteration number for the GRMRE with a diagonal preconditioner; Table 2 is the interation number for the GRMRE with a uppper triangular preconditioner; and the able 3 is the interation number for the GRMRE with an incomplete LU preconditioner with the drop tolerance 0.5. As we can see from the table, the diagonal preconditioner fails to converge, and the upper triangular preconditioner converges within one step. The ILU preconditioner converges within 80 iterations. We suspect that the upper triangular preconditioner could give the best performance if we lower the accuracy. Tables reftable:tri1-reftable:tri3 show the iteration numbers of the upper triangular preconditioner with different mesh sizes and tolerances: We have tested the mesh sizes from 32 to 1028 and the tolerances from 10^{-4} to 10^{-6}. All tables shows that the upper triangular preconditioned GRMES is robust with respect to the mesh sizes and the tolerances.

Table 4: **Upper Triangular Preconditioner with tolerance** 10^{-6}

N	FLAG	RELRES	ITER
32	0	5.60E-08	16
64	0	3.60E-07	24
128	0	8.30E-07	30
256	0	6.20E-07	36
512	0	7.90E-07	42
1028	0	7.70E-07	52

Table 5: **Upper Triangular Preconditioner with tolerance** 10^{-5}

N	FLAG	RELRES	ITER
32	0	1.0079E-5	14
64	0	2.30E-06	23
128	0	2.40E-06	28
256	0	3.20E-06	35
512	0	5.40E-06	36
1028	0	9.50E-06	42

Table 6: **Upper Triangular Preconditioner with tolerance** 10^{-4}

N	FLAG	RELRES	ITER
32	0	1.00E-05	14
64	0	2.50E-05	19
128	0	6.60E-05	24
256	0	9.80E-05	18
512	0	7.20E-05	20
1028	0	1.00E-04	23

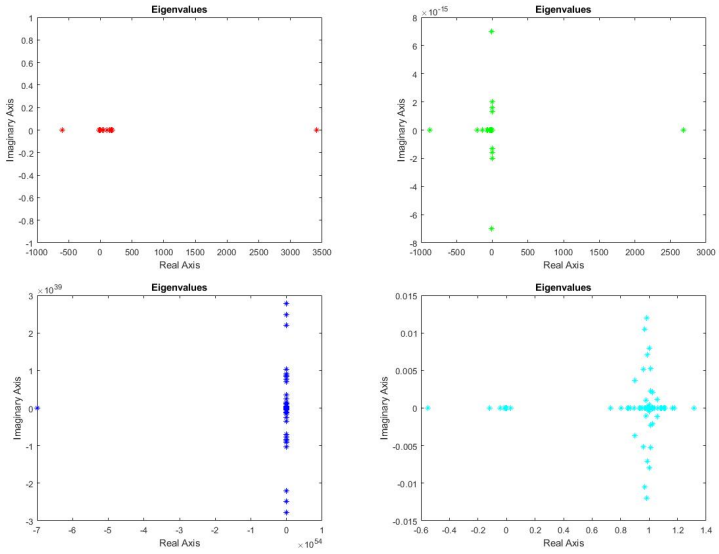

Figure 1: The eigenvalue distributions of the coefficient matrix A and the preconditioned matrices with the diagonal preconditioner, triangular preconditioner and the ILU preconditioner.

4.1. Further Observation

The eigenvalue distributions of the coefficient matrix A and the preconditioned matrices with the diagonal preconditioner, triangular preconditioner and the ILU preconditioner are presented in Figure 1. Moreover, the shaping parameter in the Radial Basis Function plays an important role for the conditioning of the coefficient matrix. Figure 2 shows the choice of the shape parameter with respect to different grid sizes. We choice the shaping parameter with the lowest conditioner number during the numerical experiments. Table 7shows the condition number with the optimal shape parameter.

5. Conclusion

In this paper we considered the steady state 2D Helmholtz equation with Dirichlet boundary conditions. A discretization was performed using a meshless approach involving Kansa's method along with Hardy's MQ as our radial basis function. The linear system arising from the discretization was then solved

by a type of Krylov subspace method known as GMRES. To better condition the problem for numerical simulations, we considered various types of preconditioners. We compared three common preconditioners: Diagonal preconditioner, upper triangular preconditioner, and incomplete LU preconditioner. The upper triangular preconditioenr performed the best among all preconditioners by converging to a solution in one iteration for all cases. From the eigenvalue plots, one can see the nice clustering of eigenvalues for both upper triangular preconditioner and ILU preconditioner. This is a strong indicator of why those preconditioned systems converged quickly.

Iterative methods for large Helmholtz problems are quite challenging especially when they are discretized by the finite element methods or finite difference methods. The resulting linear system is a large sparse system. Most preconditioned iterative methods are quite challenging in this case. However, in this paper, we found the preconditioned GMRES could have robust results with the meshless discretization. The computation cost of solving the triangular matrix is quite low. In the next study, we plan to optimize the discretization step and study the behavior of the preconditioned GMRES. We hope with the new implementation strategy, we could improve the accuracy o the solution.

Table 7: **Optimal Shape Parameter**

N	**c***	**Condition of A**
10	0.0319	682.8995
100	0.0041	9.1728E+04
200	0.0026	2.7594E+05
400	0.0015	8.5358E+05

6. Acknowledgement

The authors would like to thank the support from the Hal Marcus College of Science and Engineering at the University of West Florida and the support provided by a PSC-CUNY Award, jointly funded by the Professional Staff Congress

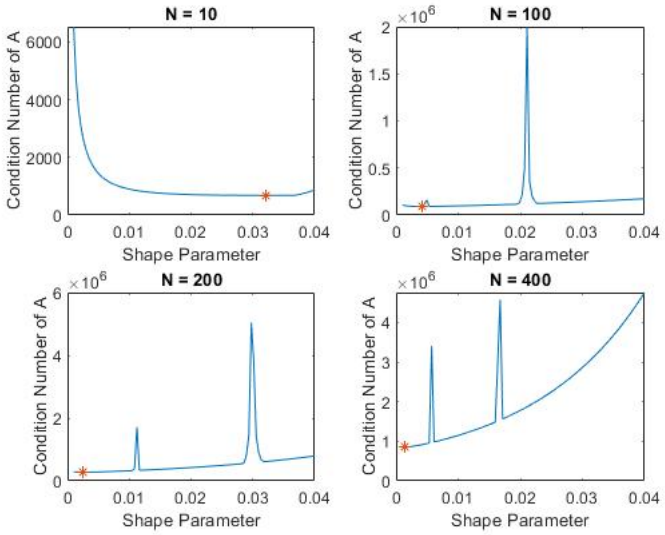

Figure 2: Optimal shape parameter for various sizes of N

and The City University of New York.

References

[1] Xavier Antoine and Marion Darbas. Integral equations and iterative schemes for acoustic scattering problems, 2016.

[2] Yogi A Erlangga. Advances in iterative methods and preconditioners for the helmholtz equation. *Archives of Computational Methods in Engineering*, 15(1):37–66, 2008.

[3] Yogi A Erlangga, Cornelis Vuik, and Cornelis Willebrordus Oosterlee. On a class of preconditioners for solving the helmholtz equation. *Applied Numerical Mathematics*, 50(3-4):409–425, 2004.

[4] Oliver G Ernst and Martin J Gander. Why it is difficult to solve helmholtz problems with classical iterative methods. In *Numerical analysis of multiscale problems*, pages 325–363. Springer, 2012.

[5] Gregory E Fasshauer. *Meshfree approximation methods with MATLAB*, volume 6. World Scientific, 2007.

[6] Martin J Gander and Hui Zhang. A class of iterative solvers for the helmholtz equation: Factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized schwarz methods. *SIAM Review*, 61(1):3–76, 2019.

[7] Yousef Saad. *Iterative methods for sparse linear systems*, volume 82. Siam, 2003.