INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 13, Number 1, Pages 166–178

AN AUGMENTED IIM FOR HELMHOLTZ/POISSON EQUATIONS ON IRREGULAR DOMAINS IN COMPLEX SPACE

SIDONG M. ZHANG AND ZHILIN LI

Abstract. In this paper, an augmented immersed interface method has been developed for Helmholtz/Poisson equations on irregular domains in complex space. One of motivations of this paper is for simulations of wave scattering in different geometries. This paper is the first immersed interface method in complex space. The new method utilizes a combination of methodologies including the immersed interface method, a fast Fourier transform, augmented strategies, least squares interpolations, and the generalized minimal residual method (GMRES) for a Schur complement system, all in complex space. The new method is second order accurate in the L^∞ norm and requires $O(N\log(N))$ operations. Numerical examples are provided for a variety of real or complex wave numbers.

Key words. Helmholtz equation, complex space, irregular domain, augmented immerse interface method, fast Poisson solver in complex space.

1. Introduction

In this paper, we consider two-dimensional Helmholtz equations in complex space on irregular domains,

$$\begin{aligned} &\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+ku=f(x,y), \qquad (x,y)\in\Omega\subset R^2,\\ &u(x,y)\Big|_{\partial\Omega}=u_0(x,y),\\ &u(x,y):\Omega\to\mathbb{C};\quad f(x,y):\Omega\to\mathbb{C};\quad k\in\mathbb{C}, \end{aligned} \tag{1}$$

see Fig. 1 for an illustration, where the domain Ω is a half circle with two parts of the boundary $\partial\Omega_1$ and $\partial\Omega_2$. One particular application is numerical simulations of wave scattering when $k\geq 0$ on an uneven surface, see for example, [2, 10] and the references therein. In reality of the electromagnetic field models, the domains are discontinuous media with general interfaces and different complex wave numbers which represents both electric and magnetic charges. The irregularity in the domain presents extra challenges for researchers and engineers do develop fast and accurate numerical methods.

One difficulty with the problem is to deal with complex numbers. An intuitive approach is to separate the problem as the real and imaginary parts. If we define $u=v+iw$, $k=k_1+k_2i$, $f=f_1+if_2$, where $i=\sqrt{-1}$ is the imaginary unit, then

Received by the editors February 28, 2014 and, in revised form, January 20, 2015.

2000 *Mathematics Subject Classification.* 65N06, 65N22, 65N50, 65F35.

The second author is partially supported by an NIH grant 5R01GM96195-2, an NSF grant DMS-1522768, and CNSF grants 11371199 and 11471166, and BK201411443.

the original partial differential equation (PDE) becomes a system of PDEs

$$\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + k_1 v - k_2 w = f_1(x, y), \tag{2}$$

$$\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + k_2 v + k_1 w = f_2(x, y). \tag{3}$$

It is not straightforward to such a system efficiently. A naive iterative method between the two system will lead to slow convergence or divergence. While it is possible to solve the coupled system of PDEs, it is more expensive than to solve it directly using the original form in complex number space.

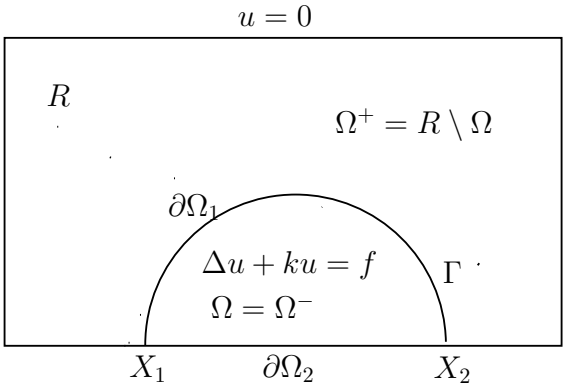


FIGURE 1. A diagram of the set-up of the problem. The Helmholtz equation is defined inside the half circle that is embedded in a rectangle R. We denote the domain inside R and outside of Ω as $\Omega^+ = R \setminus \Omega$. We also denote the original domain Ω as Ω^- for convenience. The part of boundary $\partial\Omega_1$ excluding the two points $\mathbf{X}_1$ and $\mathbf{X}_2$ becomes an interface that is denoted as Γ.

Our strategy is to solve the problem directly using complex numbers based on the augmented immersed interface method (AIIM) [4, 7]. The immersed interface method [5] is based on Cartesian grid method similar to the immersed boundary (IB) method [11, 12]. While there may be different methods in the literature, in this paper, we present a fast second order accurate finite difference method based on the immersed interface method (IIM) in complex number space.

In our approach, we use an imbedding technique to put the irregular domain Ω into a rectangular domain R so that a uniform Cartesian mesh can be used, and a fast Helmholtz/Poisson solver such as one that based on a discrete fast Fourier transform (FFT) can be utilized. We also extend the source term $f(x, y)$ to the entire rectangular domain by zero. Then the irregular boundary will become in interface. We set zero boundary condition at the auxiliary rectangular domain. Thus the original problem can be treated as an interface problem if we know the jump conditions in the solution and the flux. We set the jump in the solution as an augmented variable $[u] = Q$ and let $[u_n] = [\frac{\partial u}{\partial n}] = 0$. The augmented variable should be chosen such that the Dirichlet boundary condition is satisfied along $\partial\Omega$. The augmented variable has co-dimension one compared with that of the solution.

If we can solve the augmented variable in an efficient way, then we can get the solution to the original problem using one fast Poisson/Helmholtz solver. In this sense, the augmented approach presented here is similar to a boundary integral approach in which the source strength is solved through an integral equation. In our method, while we introduce an augmented variable along the boundary, no Green function is needed. We use the GMRES method to solve the augmented variable without explicitly form the coefficient matrix. Each GMRES iteration involves two main steps: one is to solve the Helmholtz equation on the rectangular domain with a known augmented variable in the complex number space; the second step is to use a least square interpolation scheme to compute the residual of the boundary condition.

This is the first time that the IIM has been developed for problems in complex space. We have also developed a fast Poisson/Helmholtz solver, and a GMRES solver in complex space to solve the resulting linear system of equations from the finite difference discretization, and the Schur complement system of equations, respectively. Our strategy is the same as the augmented immersed interface method (AIIM) for irregular domains, see for example, [7] except that we need to implement the AIIM in complex space.

The rest of paper is organized as follows. In the next section. We explain the problem set up and discretization process. In Section 3, we explain our fast Poisson/Helmholtz solver suing the ZFFT transformation. In Section 4, we present some numerical results followed by the conclusion and acknowledgment section.

2. Outline of the algorithm

We use an embedding technique to embed the irregular domain into a large rectangle so that a fast Poisson solver based the fast Fourier transform (FFT) can be utilized. We embed the domain Ω into a rectangle $R : \{(x, y), a \le x \le b;\ c \le y \le d\}$; and extend the PDE and the source term to the rectangle R:

$$\Delta u + ku = \begin{cases} f(x,y) & \text{if } (x,y) \in \Omega^-, \\ 0 & \text{if } (x,y) \in \Omega^+. \end{cases} \tag{4}$$

Across the interior boundary $\partial\Omega_1$, now is denoted as the interface Γ, the jump conditions are defined as

$$[u]_\Gamma = Q, \tag{5}$$

$$[u_n]_\Gamma = 0, \tag{6}$$

where $[u_\Gamma]$ is the jump of the solution of u from Ω^+ to Ω^- defined as

$$[u](\mathbf{X}) = \lim_{\mathbf{x}\to\mathbf{X},\mathbf{x}\in\Omega^+} u(\mathbf{x}) - \lim_{\mathbf{x}\to\mathbf{X},\mathbf{x}\in\Omega^-} u(\mathbf{x}), \qquad \mathbf{x} = (x, y),\ \mathbf{X} = (X, Y),$$

and so on, $u_n = \frac{\partial u}{\partial n}$ is the normal derivative along the normal direction of the boundary $\partial\Omega_1$ pointing outward. Thus, the Helmholtz equation (1) becomes

$$\Delta u + ku = \begin{cases} f(x,y) & \text{if } (x,y) \in \Omega^-, \\ 0 & \text{if } (x,y) \in \Omega^+, \end{cases} \tag{7}$$

$$[u]_\Gamma = Q, \qquad [u_n]_\Gamma = 0, \tag{8}$$

$$u(x,y)|_{\partial\Omega} = u_0(x,y), \qquad u(x,y)_{\partial R} = u_1(x,y), \tag{9}$$

where $\partial\Omega = \partial\Omega_1 \cup \partial\Omega_2$. Along the bottom of boundary ∂R outside of the domain Ω, we use a simple smooth extension from $u_0(\mathbf{X}_i, 0)$ to zero at the two bottom boundaries of R, $i = 1, 2$; and set $u_1(x,y) = 0$ along the rest of boundary of R. Note that the jump conditions are defined in the interior of R excluding two points $\mathbf{X}_1$ and $\mathbf{X}_2$ where the Dirichlet boundary conditions are defined in prior. The problem is enlarged by introducing additional augmented boundary variable Q, but it is worth doing so since we can utilize a fast Poisson on a rectangular domain R. The idea is that: given a discrete Q, we can solve the Helmholtz equation using a fast solver; the augmented variable then is updated using the GMRES iteration unless the boundary condition $u(x,y)|_{\partial\Omega} = u_0(x,y)$ is satisfied.

Let $x_i = a + ih_x$, $i = 0, 1, \cdots, m$, $h_x = (b-a)/m$; and $y_j = c + jh_y$, $j = 0, 1, \cdots, n$, $h_y = (d-c)/n$ be a grid. We use a discrete real level set function φ_{ij} to implicitly define the interior boundary $\Gamma = \{(x,y),\ \varphi(x,y) = 0\}$. Often $\varphi(x,y) \in C^2(R)$ is chosen as the signed distance function for which $\|\nabla\varphi\| = 1$. With the level set function, we classify the grid points in R as regular and irregular. At each grid, we define

$$\varphi_{ij}^{min} = \min\left\{ \varphi_{i-1,j}, \varphi_{ij}, \varphi_{i+1,j}, \varphi_{i,j-1}, \varphi_{i,j+1} \right\}, \tag{10}$$

$$\varphi_{ij}^{max} = \max\left\{ \varphi_{i-1,j}, \varphi_{ij}, \varphi_{i+1,j}, \varphi_{i,j-1}, \varphi_{i,j+1} \right\}. \tag{11}$$

If $\varphi_{ij}^{min}\varphi_{ij}^{max} > 0$ which means that the level set function does not change sign, then the grid point (x_i, y_j) is a regular one; Otherwise the grid point is an irregular one.

At an irregular grid point $\mathbf{x}_{ij} = (x_i, y_j)$ where $\varphi_{ij} \geq 0$ (or $\varphi_{ij} \leq 0$), we numerically compute the orthogonal projection $\mathbf{x}_{ij}^*$, corresponding to one of (X_k, Y_k)'s on the interface

$$\mathbf{x}_{ij}^* = \mathbf{x}_{ij} + \lambda\left(\mathbf{p}^T\nabla\varphi(\mathbf{x}_{ij})\right) + \frac{1}{2}\lambda^2\mathbf{p}^T H(\mathbf{x}_{ij})\mathbf{p}, \tag{12}$$

where $\mathbf{p} = \nabla\varphi(\mathbf{x}_{ij})$ is the gradient of $\varphi(x,y)$, and $H(\mathbf{x}_{ij})$ is the Hessian matrix of $\varphi(x,y)$ evaluated at $\mathbf{x}_{ij}$. The scale λ is the solution of the quadratic equation such that $\varphi(\mathbf{x}_{ij}^*) = 0$, see [7] for the detail. Given a discrete quantity along the interface, we can interpolate the quantity at the discrete points to get its value, the first and second order surface derivatives anywhere along the interface. We refer the readers to Section 11.1.5 in [7] for the details.

The set of orthogonal projections (X_k, Y_k), $k = 1, 2, \cdots, N_b$ of all irregular grid points on the interface from a particular side, say $\Omega^- = \Omega$ side, forms a discretization of the interface Γ. We refer the reader to Section 1.6.4 in [7] about how to find

approximate orthogonal projections. Then the discrete augmented variable $\mathbf{Q}_k$ of the continuous one Q is defined at those orthogonal projections.

At a point of the interface (X, Y), the local coordinate system in the normal and tangential directions is defined as (see Figure 2 for an illustration),

$$\begin{cases} \xi = (x - X)\cos\theta + (y - Y)\sin\theta, \\ \eta = -(x - X)\sin\theta + (y - Y)\cos\theta, \end{cases} \tag{13}$$

where θ is the angle between the x-axis and the normal direction, pointing to the Ω^+ sub-domain. Under such new coordinates system, the interface can be parameterized as

$$\xi = \chi(\eta) \qquad \text{with} \qquad \chi(0) = 0, \quad \chi'(0) = 0. \tag{14}$$

The curvature of the interface at (X, Y) is $\chi''(0)$.

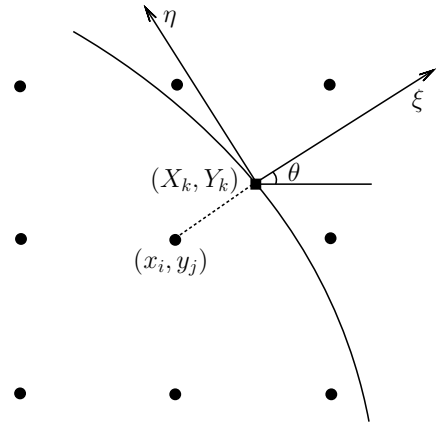


FIGURE 2. A diagram of an irregular grid point (x_i, y_j), its orthogonal projections on the interface (X_k, Y_k), and the local coordinates at (X_k, Y_k) in the normal and tangential directions.

If we know the jump in the solution $[u] = w$ and the normal derivative $[u_n] = Q$, then we can have the following jump relations at a point (X, Y) on the interface which is necessary to derive the accurate finite difference method.

$$\begin{aligned} &[u] = w, \qquad [u_\eta] = w', \qquad [u_\xi] = Q, \\ &[u_{\eta\eta}] = -q\chi'' + w'', \qquad [u_{\xi\eta}] = w'\chi'' + Q', \\ &[u_{\xi\xi}] = Q\chi'' - w'' + [f], \end{aligned} \tag{15}$$

where w', Q' and w'' are the first and second order surface derivatives of w and Q on the interface, which are all evaluated at $(\xi, \eta) = (0, 0)$. Here we skip the derivation which is similar to those derived in equation (3.5) in Section 3.1 in [5, 7] assuming that $[u] = w$ and $[\beta u_n] = v$ are given with $\beta = 1$. Note also that we can express the jump conditions in terms of u^+, u^+_η, and u^+_ξ.

Once we have the jump relations in the local coordinates, we can get back the jump relations in the x- and y- directions according to (9.47) in [7]

$$\begin{aligned}
&[u_x] = [u_\xi]\cos\theta - [u_\eta]\sin\theta,\\
&[u_y] = [u_\xi]\sin\theta + [u_\eta]\cos\theta,\\
&[u_{xx}] = [u_{\xi\xi}]\cos^2\theta - 2[u_{\xi\eta}]\cos\theta\sin\theta + [u_{\eta\eta}]\sin^2\theta,\\
&[u_{yy}] = [u_{\xi\xi}]\sin^2\theta + 2[u_{\xi\eta}]\cos\theta\sin\theta + [u_{\eta\eta}]\cos^2\theta.
\end{aligned} \tag{16}$$

For a given augmented variable Q in the discrete form $\{Q_i\} = \mathbf{Q}$, the finite difference solution is the solution of the following linear system of equations,

$$\begin{aligned}
&\frac{U_{i-1,j} - 2U_{i,j} + U_{i+1,j}}{h_x^2} + \frac{U_{i,j-1} - 2U_{i,j} + U_{i,j+1}}{h_y^2} + kU_{i,j}\\
&\qquad = \begin{cases} f_{ij} & \text{if } \mathbf{x}_{ij} \text{ is regular,}\\ f_{ij} + S_{ij} & \text{if } \mathbf{x}_{ij} \text{ is irregular,}\end{cases}
\end{aligned} \tag{17}$$

where $S_{ij} \in C$ are correction terms at irregular grid points to offset the discontinuities in the jump of the solution and the normal derivative. For example, assume the interface cuts the grid line $y(x) = y_j$ in the interval (x_{i-1}, x_{i+1}), say at (x_{ij}^*, y_j), with $x_{ij}^* = x_i + \alpha_{ij}^x\, h,\ 0 \le |\alpha_{ij}^x| < 1$. Without lost of generality, we assume that $(x_i, y_j) \in \Omega^-$. We discretize the u_{xx} to get

$$u_{xx}^-(x_i, y_j) \approx \frac{U_{i-1,j} - 2U_{i,j} + U_{i+1,j} - C_{ij}^x}{h^2}, \tag{18}$$

where the correction term C_{ij}^x is

$$C_{ij}^x = [u] + [u_x]\left(1 - |\alpha_{ij}^x|\right)h + [u_{xx}]\frac{(1 - |\alpha_{ij}^x|)^2 h^2}{2}. \tag{19}$$

Similarly, we can discretize u_{yy} if the interface cut through (y_{j-1}, y_{j-1}). We refer the reader to [7] about how to determine the correction terms. The only difference here is that we use the complex number space. In the matrix vector form, the discretization can be written as

$$A_{11}\mathbf{U} + A_{12}\mathbf{Q} = \mathbf{F} \tag{20}$$

for some sparse matrices A_{11} and A_{12}. Note that A_{11} is the discrete Helmholtz in the complex space. Note also that to get the vector $A_{11}^{-1}\mathbf{F}$ requires one call to a fast Helmholtz solver in complex space. We will explain our fast Helmholtz solver in the next section.

The augmented variable Q, or $\mathbf{Q}$ in discrete form, should be chosen so that the solution u satisfy the boundary condition. At every approximate orthogonal projections of all irregular grid points on the interface, we use the same least squares interpolation described in Section 4 in [6] to interpolate the boundary condition $u(x, y)|_\Gamma = u_0(x, y)$.

At one orthogonal projection $\mathbf{X}_k = (X_k, Y_k)$ corresponding to an irregular grid point (x_i, y_j), the second order accurate least squares interpolation scheme approximating $u(x, y)|_\Gamma = u_0(x, y)$ can be written as

$$\sum_{|\mathbf{x}_{ij} - \mathbf{X}_k| \le \delta_h} \gamma_{ij} U_{ij} + L_k(\mathbf{W}, \mathbf{Q}) = 0 \tag{21}$$

where δ_h is a parameter of $2h \sim 3h$, L_k stands for a linear relation of its augmenters, the discrete form of $w(\mathbf{X})$ and $q(\mathbf{X})$. The consistency condition requires that

$$\sum_{|\mathbf{x}_{ij}-\mathbf{X}_k|\le\delta_h} \gamma_{ij} = 1. \tag{22}$$

Note that the interpolation coefficients should depend on the index k as well, we omit it for simplicity of notations. In the matrix-vector form, the interpolation can be written as

$$A_{21}\mathbf{U} + A_{22}\mathbf{Q} = \mathbf{F}_2. \tag{23}$$

If we combine (20) and (23), we can get the Schur complement system for $\mathbf{Q}$ which has co-dimension one as that of $\mathbf{U}$,

$$\left(A_{22} - A_{21}A_{11}^{-1}A_{12}\right)\mathbf{Q} = \mathbf{F}_2 - A_{21}A_{11}^{-1},\mathbf{F}. \tag{24}$$

or

$$S\mathbf{Q} = \mathbf{F}_s, \qquad S = \left(A_{22} - A_{21}A_{11}^{-1}A_{12}\right), \quad \mathbf{F}_s = \mathbf{F}_2 - A_{21}A_{11}^{-1}.$$

Note that the matrix S is an N_b by N_b matrix, and $\mathbf{Q} \in C^{N_b}$ which has co-dimension one compared with that of U. In implementation, we do not form those matrices explicitly since we use the GMRES method in the complex number space to solve the Schur complement system. Each GMRES iteration requires one call to the fast Helmholtz solver; and the interpolation scheme to compute the residual of the boundary condition. If we define the residual vector as

$$R(\mathbf{Q}) = \left(A_{22} - A_{21}A_{11}^{-1}A_{12}\right)\mathbf{Q} - \mathbf{F}_s = \mathbf{U}_b(\mathbf{Q}) - u_0(\mathbf{X}), \tag{25}$$

where $\mathbf{U}_b(\mathbf{Q})$ is the vector from the interpolation of the solution from grid points $\mathbf{x}_{ij}$ to the interface $\mathbf{X}_k$, $u_0(\mathbf{X})$ is the boundary condition at those $\mathbf{X}_k$'s, then the right hand side vector of the Schur complement is

$$R(\mathbf{0}) = \left(A_{22} - A_{21}A_{11}^{-1}A_{12}\right)\mathbf{0} - \mathbf{F}_s = -\mathbf{F}_s = u_0(\mathbf{X}) - \mathbf{U}_b(\mathbf{0}).$$

The matrix-vector multiplication of $S\mathbf{Q}$ needed for the GMRES iteration given $\mathbf{Q}$ is obtained by first solving the interface problem given $\mathbf{Q}$ to get $\mathbf{U}(\mathbf{Q})$, then the interpolation scheme to get the solution at $\mathbf{U}_b(\mathbf{Q})$.

$$S\mathbf{Q} = R(\mathbf{Q}) - R(\mathbf{0}) = \mathbf{U}_b(\mathbf{Q}) - U_b(\mathbf{0}).$$

We refer the readers to [7] for the details.

3. ZFFT methods for solving two dimensional Poisson/Helmholtz Equation in Rectangular Domains

The ZFFT is referred to the Fast Fourier transform (FFT) in complex space. The Fast Fourier transform (FFT) method for solving a two dimensional Poisson equation was first introduced by Cooley and Tukey [3] in 1965. It was focused on squared domains with uniformed meshes. In 1984, Swarztrauber further developed the fast Poisson/Helmholtz solvers based on FFT for rectangular domains [14]. The fast Poisson solver based on FFT has the optimal operation count that only requires $O(N \log N)(N = mn)$ operations.

It is well known that the condition number of the resulting linear system of equation for the Poisson equation is $O(mn)$, and even bigger for Helmholtz equation

when $k \geq 0$. Thus for reasonable accuracy, double precision is needed. It is not so easy to find available fast Poisson/Helmholtz solver based on FFT in double precision for complex numbers. Below we explain our fast Poisson/Helmholtz solver based on FFT.

For simplicity, we assume that $h_x = h_y = h$. For general situations, we refer the readers to [15]. The ZFFT method is used to solve the liner system of equations

$$\frac{U_{i-1,j} + U_{i+1,j} + U_{i,j-1} + U_{i,j-1} - 4U_{i,j}}{h^2} + kU_{i,j} = f_{ij}, \tag{26}$$

$i = 1, \cdots, m; j = 1, \cdots, n$ in complex space, here f_{ij} has different meaning compared the $f(x, y)$ in (1). In the matrix-vector form, we can re-write the system above as,

$$T_m U + U T_n + h^2 k U = h^2 F, \tag{27}$$

where

(28)

$$T_m = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -2 \end{bmatrix}_{m\times m} \quad T_n = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -2 \end{bmatrix}_{n\times n}$$

By the discrete Fourier transform, $T_r = V_r^{-1} D_r V_r$, $r = m, n$, we have

$$V_m^{-1} D_m V_m U + U V_n^{-1} D_n V_n + h^2 k U = h^2 F, \tag{29}$$

where $V_r = (s_{i.j}^{(r)})_{r,r}$, $s_{i.j}^{(r)} = \sin(\frac{2\pi ij}{2(r+1)})$, and

$$D_r = \begin{bmatrix} \lambda_1^{(r)} & 0 & \cdots & 0 \\ 0 & \lambda_2^{(r)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_r^{(r)} \end{bmatrix}, \quad \lambda_i^{(r)} = -4\sin^2(\frac{i\pi}{2(r+1)}), i, j = 1, \cdots, r. \tag{30}$$

Note that V has another nice property,

$$V_r^{-1} = \frac{2}{r+1} V_r. \tag{31}$$

Multiplying V_m from left and V_n^{-1} from right to both sides of the (29), we have

$$D_m V_m U V_n^{-1} + V_m U V_n^{-1} D_n + h^2 k V_m U V_n^{-1} = h^2 V_m F V_n^{-1}. \tag{32}$$

We define

$$\overline{U} = V_m U V_n^{-1} = (\overline{u}_{ij})_{m,n}; \quad \text{and} \quad \overline{F} = h^2 V_m F V_n^{-1} = (\overline{f}_{ij})_{m,n}, \tag{33}$$

and substitute (33) into (32) to get

$$D_m \overline{U} + \overline{U} D_n + h^2 k \overline{U} = \overline{F}. \tag{34}$$

Since D_m and D_n in (34) are diagonal, we solve (34) for $\overline{U}$ which yields,

$$\overline{U}_{i,j} = \frac{\overline{F}_{i,j}}{\lambda_i^{(m)} + \lambda_j^{(n)} + h^2 k}, i = 1, \cdots, m; j = 1, \cdots, n. \tag{35}$$

Applying the discrete inverse Fourier transform, we get

$$U = V_m^{-1}\overline{U}V_n. \tag{36}$$

Since m is different from n, we need to construct two different sets of discrete Fourier transform in (33) and (36). When the FFT solver is needed repeatedly, like in our application in solving the Helmholtz equation on irregular domains, we store those coefficients and re-use them rather re-constructing them every time.

3.1. Algorithm Summary. Here is the outline of the fast Fourier transform (FFT) in complex space.

Step 1. Perform the discrete Fourier transform on F to get

$$\overline{F} = h^2 V_m F V_n^{-1} = \frac{2h^2}{n+1} V_m F V_n. \tag{37}$$

Step 2. Compute the intermediate solution $\overline{U}$:

$$\overline{u}_{i,j} = \frac{\overline{f_{i,j}}}{\lambda_i^{(m)} + \lambda_j^{(n)} + h^2 k}, i = 1, \cdots, m; j = 1, \cdots, n, \tag{38}$$

where $\lambda_i^{(m)} = -4\sin^2(\frac{i\pi}{2(m+1)})$ and $\lambda_j^{(n)} = -4\sin^2(\frac{j\pi}{2(n+1)})$.

Step 3. Perform the inverse discrete Fourier transform on $\overline{U}$ to get

$$U = V_m^{-1}\overline{U}V_n = \frac{2}{m+1} V_m \overline{U} V_n. \tag{39}$$

3.2. Efficiency Analysis. We can see that the cost of Step 2 above is about $3 \times m \times n$ operations; and Step 1 and Step 3 seem to require a triple matrix multiplication. If we did use a straightforward matrix multiplication, then the cost would be $m^2 n + mn^2$ flops, which is $O(m^3)$ assuming $m \sim n$. However, the triple matrix multiplication actually represents the *discrete Fourier transform*. Thus, with the FFT, the computation cost of each column of a $m \times n$ matrix is reduced to $(m^2 \log_2 m)$, see [1], and the cost of each row of a $m \times n$ matrix is $\log_2(n \times 3n/2)$. Therefore, the total computing cost for a $m \times n$ matrix is $O(mn \log(mn))$.

4. Numerical Examples

We have performed a number of numerical experiments using the proposed Augmented Immersed Interfaced Method in complex space (AIIMC). The results confirmed that the AIIMC is second order accurate for Helmholtz/Poisson equations in complex space on irregular domains. We present some numerical example with the same analytic solution but with three different geometries (boundary) and various wave number k. More examples can be found in [15]. One of the boundaries is the half circle whose level set function is

$$\varphi(x, y) = \sqrt{x^2 + y^2} - 0.5, \ y \geq 0.$$

The second one is the half oval whose level set function is

$$\varphi(x, y) = \frac{x^2}{a_0^2} + \frac{y^2}{b_0^2} - 1, \ y \geq 0.$$

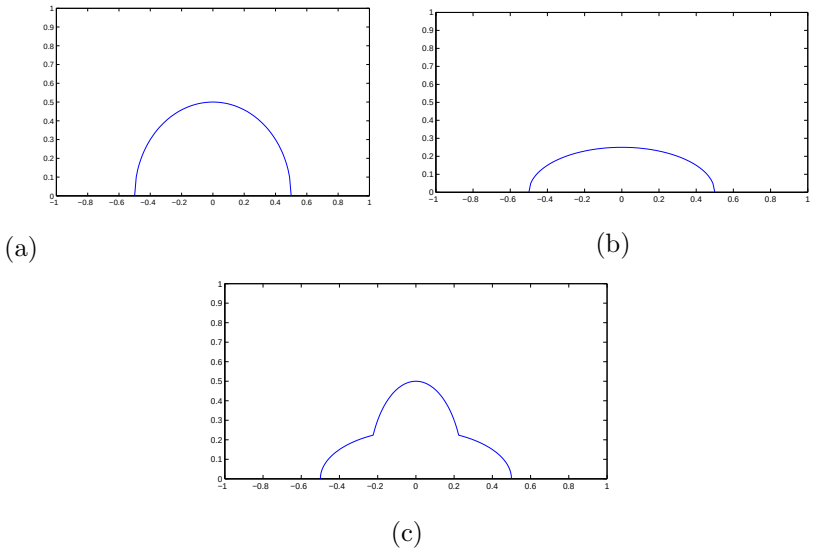
The third one is the half flower petal whose level function is

$$\varphi(x,y) = \min\left\{\varphi_1(x,y),\ \varphi_2(x,y)\right\},$$

where

$$\varphi_1(x,y) = \frac{x^2}{a_0^2} + \frac{y^2}{b_0^2} - 1, \quad \varphi_2(x,y) = \frac{x^2}{b_0^2} + \frac{y^2}{a_0^2} - 1,\ y \geq 0,$$

see Fig. 3 for all the geometries. In computation, we set $a_0 = 0.5$ and $b_0 = 0.25$.



(a) (b)

(c)

FIGURE 3. Irregular domains with a half circle boundary (upper left), a half oval boundary (upper right), and a half flower petal boundary (the bottom plot). For the middle and right figures, we set $a_0 = 0.5$ and $b_0 = 0.25$.

TABLE 1. The grid refinement analysis for the half circle boundary.

	$k = 100i$		$k = 100$		$k = 100 + 100i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	1.9172309E-02		1.3101805E-01		2.8496815E-02	
64	4.9807631E-03	1.94	2.1360192E-02	2.62	7.1232807E-03	2.00
128	1.3884691E-03	1.84	6.3771377E-03	1.74	1.9176186E-03	1.89
256	3.4969866E-04	1.99	1.4476327E-03	2.14	4.7597509E-04	2.01
512	8.8140329E-05	1.99	3.6014252E-04	2.01	1.1939369E-04	2.00
Convergence Order	1.94		2.09		1.97	
	$k = 0$		$k = 10 + 10i$		$k = 1 + 2i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	4.2454093E-02		4.2454093E-02		4.7394051E-02	
64	1.0469232E-02	2.02	1.0469232E-02	2.02	1.1685877E-02	2.02
128	2.6134649E-03	2.00	2.6134649E-03	2.00	2.9153959E-03	2.00
256	7.3696011E-04	1.83	7.3696011E-04	1.83	7.7192953E-04	1.92
512	1.8364497E-04	2.00	1.8364497E-04	2.00	1.9422000E-04	1.99
Convergence Order	1.95		1.95		1.98	

We selected the true solution as $u(x,y) = e^x \cos y$, a genuine non-linear function. The source term is determined from the true solution according to $f(x,y) = u_{xx} +$

TABLE 2. The grid refinement analysis for the half oval boundary.

	$k = 100i$		$k = 100$		$k = 100 + 100i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	1.3909343E-02		1.3494268E-02		2.1417184E-02	
64	3.5219551E-03	1.98	3.4273603E-03	1.98	5.1010118E-03	2.07
128	1.1136624E-03	1.66	1.0692560E-03	1.68	1.5781984E-03	1.69
256	3.9142595E-04	1.51	2.7907968E-04	1.94	5.4499299E-04	1.54
512	9.8978365E-05	1.98	7.4918455E-05	1.90	1.3846264E-04	1.98
Convergence Order	1.74		1.86		1.78	
	$k = 0$		$k = 10 + 10i$		$k = 1 + 2i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	4.2345275E-02		6.3370572E-02		4.9557268E-02	
64	1.0527419E-02	2.01	1.6014427E-02	1.99	1.2323084E-02	2.01
128	2.6216757E-03	2.01	4.0061370E-03	2.00	3.0667723E-03	2.01
256	6.5430825E-04	2.00	9.9966997E-04	2.00	7.6546386E-04	2.00
512	1.6348735E-04	2.00	2.4947955E-04	2.00	1.9119254E-04	2.00
Convergence Order	2.00		2.00		2.00	

TABLE 3. The grid refinement analysis for the flower petal boundary.

	$k = 100i$		$k = 100$		$k = 100 + 100i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	1.7045674E-02		1.1645818E-01		2.6767626E-02	
64	3.5129496E-03	2.28	4.1921111E-02	1.48	4.9592515E-03	2.43
128	1.1078943E-03	1.66	1.5779112E-02	1.41	1.5496942E-03	1.68
256	3.9008112E-04	1.51	3.1673795E-03	2.32	5.4311581E-04	1.51
512	9.8402477E-05	1.99	8.5726590E-04	1.88	1.3750733E-04	1.98
Convergence Order	1.80		1.79		1.84	
	$k = 0$		$k = 10 + 10i$		$k = 1 + 2i$	
Mesh	Error	Rate	Error	Rate	Error	Rate
32	5.2950678E-02		7.0444768E-02		5.9067049E-02	
64	1.2893373E-02	2.04	2.1849600E-02	1.69	1.4873880E-02	1.99
128	4.8007356E-03	1.43	7.1862374E-03	1.60	5.2856641E-03	1.49
256	6.5230164E-04	2.88	1.0328548E-03	2.80	7.3415854E-04	2.85
512	4.5758565E-04	0.52	6.6852508E-04	0.62	5.0220057E-04	0.55
Convergence Order	1.80		1.78		1.81	

TABLE 4. Efficiency analysis of the new augmented IIM for complex space: The number of GMRES iterations without any preconditioning. (a) the half circle boundary; (b) the half oval boundary; (c) the flower petal boundary. Note that, the number of iterations is almost independent of the meshes with a possible $\log N$ factor.

(a) the half circle boundary.

Mesh	$k = 100i$	$k = 100$	$k = 100 + 100i$	$k = 0$	$k = 10 + 10i$	$k = 1 + 2i$
32	9	14	10	8	8	9
64	9	12	9	7	7	8
128	9	12	10	8	8	9
256	9	12	10	9	9	8
512	9	12	10	9	9	9

(b) The half oval boundary.

Mesh	$k = 100i$	$k = 100$	$k = 100 + 100i$	$k = 0$	$k = 10 + 10i$	$k = 1 + 2i$
32	9	10	11	8	10	9
64	9	12	10	7	9	8
128	8	12	10	9	10	9
256	8	11	10	9	10	8
512	8	12	10	9	10	9

(c) The flower petal boundary.

Mesh	$k = 100$i	$k = 100$	$k = 100 + 100i$	$k = 0$	$k = 10 + 10i$	$k = 1 + 2i$
32	10	11	12	11	12	12
64	9	11	10	11	11	11
128	14	16	15	15	16	15
256	10	13	11	13	13	12
512	17	17	15	18	19	18

$u_{yy} + ku$. We check the convergence order of the method using the grid refinement analysis. The errors are defined in the L^∞ norm,

$$E_n = \max_{(x_i,y_j)\in\Omega^-} |U_{ij} - u(x_i, y_j)|, \tag{40}$$

where U_{ij} is the numerical solution, $u(x_i, y_j)$ is the true solution, and n is the mesh size with m/n being a constant. We double the mesh size for the grid refinement analysis by doubling the mesh every time. Thus the convergence rate is

$$Rate = \log_2 \frac{\|E_n\|_\infty}{\|E_{2n}\|_\infty}.$$

In Table 1-3, we show the errors with different wave numbers, mesh sizes; and boundaries. The first column is the mesh size m assuming $m = 2n$. The second, fourth, and sixth columns are the infinity errors measuring at all grid points inside the domain ($\varphi_{ij} \leq 0$. The third, fifth, and seventh columns are the convergence order in the infinity norm. The numerical results clearly indicated second order accuracy for the computed solution. In Table 4, we list number of iterations of the GMRES method which is nearly independent of the mesh size and the wave number except a factor perhaps of $|\log h|$.

5. Conclusions and acknowledgement

This paper has extended the IIM and the augmented IIM for irregular domains to two dimensional Helmholtz/Poisson equations in complex space.

In the process, we have achieved the following: (1). We have developing a new two-dimensional fast Fourier transform in complex space to solve Helmholtz/Poisson equations on rectangular domains. This method is second order accurate with respect to the mesh size with $O(N \log N)$ operations. (2). We have extended and analyzed the Immersed Interface Method for two-dimensional Helmholtz/Poisson equations in complex number space with singularities. (3). We have extended the augment IIM for the Helmholtz/Poisson equation on irregular domains to complex space which involves a least squares interpolation, the Schur complement, and the GMRES iteration.

References

[1] J. W. Demmel, Applied Numerical Linear Algebra, SIAM , 1997, pp.270-274.

[2] Z. Haznadar; Z. Stih, Electromagnetic Fields Waves and Numerical Methods IOS Press 2000.

[3] J. W. Cooley, J. W. Tukey, An algorithm for the machine calculation of complex Fourier series Math. Comput. 19 (1965) 297-301.

[4] B. C. Khoo, Z. Li, P. Lin, Interface Problems and Methods in Biological and Physical Flows, World Scientific, 2009.

[5] R. L. LeVeque and Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM J. Numer. Anal., 31 (1994) 1019-1044.

[6] Z. Li, A fast iterative algorithm for elliptic interface problems. SIAM J. Numer. Anal., 35:230–254, 1998.

[7] Z. Li, K. Ito, The Immersed Interface Method Numerical Solutions of PDEs Involving Interfaces and Irregular Domains, SIAM, 2006.

[8] Z. Li, T. Lin, Y. Lin, and R. C. Rogers, An immersed finite element space and its approximation capability, Numer. Methods Part. Diff. Equ., 20 (2004) 338-367.

[9] Z. Li, H. Zhao, and H. Gao. A numerical study of electro-migration voiding by evolving level set functions on a fixed Cartesian grid. J. Computing Phys., 152 (1999) 281-304,
[10] C. Muller, Foundations of the Mathematical Theory of Electromagnetic Waves, Springer-Verlag, 1969.
[11] C. S. Peskin. Numerical analysis of blood flow in the heart. J. Comput. Phys., 25 (1977) 220-252.
[12] C. S. Peskin. The immersed boundary method. Acta Numer., 11 (2002) 479-517.
[13] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems. SIAM J. Sci. Stat. Comput., 7 (1986) 856-869.
[14] P. N. Swarztrauber, Fast Poisson Solvers. Second International Conference on Computational Modeling of Free and Moving Boundary Problems 93, 1984.
[15] S. Zhang, The Immersed Interface Method for Two Dimensional Poisson/Helmholtz Equations in Complex Number Space, PhD thesis, North Carolina State University, 2013.

Department of Mathematics and Information Technology and Security, Campbell University, Buies Creek, NC 27506

E-mail: zhangs@campbell.edu

Corresponding author. Department of Mathematics, North Carolina State University, Raleigh, NC 27695, & Nanjing Normal University

E-mail: zhilin@math.ncsu.edu

URL: http:///www4.ncsu.edu/∼zhilin/