A HIGH ORDER NYSTRÖM METHOD FOR BOUNDARY INTEGRAL EQUATIONS ON AXISYMMETRIC SURFACES

PATRICK M. YOUNG[†][‡] AND PER-GUNNAR MARTINSSON[†][§]

Abstract. A high order Nyström discretization scheme for rapidly and accurately computing solutions to boundary integral equations (BIEs) on rotationally symmetric surfaces in $\mathbb{R}^3$ is presented. The scheme uses the Fourier transform to reduce the original BIE defined on a surface to a sequence of BIEs defined on a generating curve for the surface. It can handle loads that are not rotationally symmetric. The Nyström discretization is defined on the generating curve of the body and is based on a Gaussian quadrature rule which is modified near the diagonal to retain high-order accuracy for singular kernels. The reduction in dimensionality, along with the use of high-order accurate quadratures, leads to small linear systems that can be inverted directly via, e.g., Gaussian elimination. This makes the scheme particularly fast in environments involving multiple right hand sides. It is demonstrated that for BIEs associated with Laplace's equation, the kernel in the reduced equations can be evaluated very rapidly by exploiting recursion relations for Legendre functions. Numerical examples illustrate the performance of the scheme; in particular, it is demonstrated that for a BIE associated with Laplace's equation on a surface discretized using 320 800 points, the set-up phase of the algorithm takes 57 seconds on a standard desktop, and then solves can be executed in 0.39 seconds. By combining the technique with the FMM, the scheme is easily extended to multiply connected domains where the geometry is only locally axisymmetric.

Key words. boundary integral equations, high order discretization, body of revolution

AMS subject classifications. 65R20

1. Introduction.

This paper presents a high order Nyström discretization technique for boundary integral equations (BIEs) defined on axisymmetric surfaces in $\mathbb{R}^3$. Specifically, we consider second kind Fredholm equations of the form

$$\sigma(\boldsymbol{x}) + \int_\Gamma k(\boldsymbol{x},\boldsymbol{x}')\,\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}') = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Gamma, \tag{1.1}$$

under two assumptions: First, that Γ is a surface in $\mathbb{R}^3$ obtained by rotating a curve γ about an axis. Second, that the kernel k is invariant under rotation about the symmetry axis in the sense that

$$k(\boldsymbol{x},\boldsymbol{x}') = k(\theta-\theta', r, z, r', z'), \tag{1.2}$$

where $(r,\, z,\, \theta)$ and $(r',\, z',\, \theta')$ are cylindrical coordinates for $\boldsymbol{x}$ and $\boldsymbol{x}'$, respectively,

$$\begin{aligned}\boldsymbol{x} &= (r\,\cos\theta,\; r\,\sin\theta,\; z),\\ \boldsymbol{x}' &= (r'\,\cos\theta',\; r'\,\sin\theta',\; z'),\end{aligned}$$

see Figure 1.1. Numerical methods for BIEs under these assumptions have previously been proposed in several different contexts: stress analysis [3], scattering [12, 24, 31, 32, 33], and potential theory [18, 28, 29, 30]. The premise of these works is an observation (see, e.g., [29]) that the rotational symmetry in the geometry can be exploited to reduce the dimensionality of the problem:

Observation 1: *The equation (1.1), which is defined on the two-dimensional surface Γ, can via a Fourier transform in the azimuthal variable be recast as a sequence of*

[†] Dept. of Applied Mathematics, Univ. of Colorado at Boulder, Boulder, CO 80309-0526

[‡] Supported by NSF Grant DMS-0602284

[§] Supported by NSF Grant DMS-0748488

equations defined on the one-dimensional curve γ. To be precise, letting σ_n, f_n, and k_n denote the Fourier coefficients of σ, f, and k, respectively (for a precise definition see (2.6), (2.7), (2.8)), the equation (1.1) is equivalent to the sequence of equations

$$\sigma_n(r,z) + \sqrt{2\pi}\int_\gamma k_n(r,z,r',z')\,\sigma_n(r',z')\,r'\,dl(r',z') = f_n(r,z), \qquad (r,z)\in\gamma, \quad (1.3)$$

for $n \in \mathbb{Z}$.

Whenever f can be represented with a moderate number of Fourier modes, the transformation of (1.1) to (1.3) can dramatically accelerate numerical computations. The principal benefit is that while (1.1) would upon discretization lead to one large linear system, the formulation (1.3) leads to a sequence of uncoupled small linear systems (one for each Fourier mode). Since the system matrices are dense, the gain is substantial. (Observe that the cost of moving between physical space and Fourier space is essentially negligible due to the speed of the FFT.) Moreover, when discretizing BIEs, it is much easier to work with a curve than with a surface. The drawback of the axisymmetric approach is that the kernel function k_n in (1.3) is not easily evaluated.

All observations listed so far are well known and have been exploited in, e.g., [3, 12, 18, 24, 28, 29, 30, 31, 32, 33]. The present work improves upon earlier results in two key regards: (1) Highly efficient methods for evaluating the kernel k_n in (1.3) are presented for the case where k is either the single or the double layer kernel associated with Laplace's equation. (2) While earlier work discretized (1.3) using low-order accurate Galerkin or collocation methods, we have constructed a high order accurate Nyström scheme. (The advantages of such schemes are described in, e.g., [8].) It discretizes γ into $N_{\rm P}$ panels, and then collocates σ_n and f_n at $N_{\rm G}$ Gaussian nodes on each panel. By modifying the weights for quadrature nodes that are "close" (in the same or touching panels) a discretization error of approximately $O((1/N_{\rm P})^{N_{\rm G}})$ is attained.

Numerical experiments indicate that for simple surfaces, a relative accuracy of 10^{-10} is obtained using as few as a couple of hundred points along the generating curve. The rapid convergence of the discretization leads to linear systems of small size that can be solved *directly* via, *e.g.*, Gaussian elimination, making the algorithm particularly effective in environments involving multiple right hand sides and when the linear system is ill-conditioned.

In order to describe the asymptotic cost of the proposed method, we make the simplifying assumption that the number of Fourier modes computed equals the number of points used to discretize the generating curve γ. Splitting the computational cost into a "set-up" cost that needs to be incurred only once for a given geometry and given discretization parameters, and a "solve" cost representing the time required to process each right hand side, we have

$$T_{\rm setup} \sim \underbrace{N^{3/2}\log(N)}_{\text{construction of linear systems}} + \underbrace{N^2}_{\text{inversion of systems}}, \qquad (1.4)$$

and

$$T_{\rm solve} \sim \underbrace{N\log(N)}_{\text{FFT of boundary data}} + \underbrace{N^{3/2}}_{\text{application of inverses}}, \qquad (1.5)$$

where N is the total number of degrees of freedom in the system. (A precise description of the asymptotic costs as functions of the number of Fourier modes, of the number of panels, and of the order of the Gaussian quadrature nodes can be found in Section 4.3.) The constants of proportionality hidden in (1.4) and (1.5) were estimated via numerical experiments reported in Section 6. As a preview of the effectiveness of the presented approach, let us mention that for a problem involving $N = 320\,800$ degrees of freedom, the setup time was 57 seconds and the solve time was 0.39 seconds when the method was implemented on a standard desktop PC. With this many degrees of freedom, solutions with more than ten digits of accuracy were constructed for all geometries considered. These numbers indicate that while there exist methods with linear complexity for solving (1.1), the present approach will in practice be much faster unless N is extremely large, see Remark 1.1.

The Nyström discretization scheme is particularly simple to combine with accelerated methods such as the Fast Multipole Method [15] and $\mathcal{H}$-matrix methods [20] which means that the proposed scheme can be extended to handle domains involving several rotationally symmetric bodies (whose symmetry axes do not need to be aligned). The very high speed is to some degree lost since interbody interactions cannot be accelerated via the FFT, but the benefits of high accuracy remain. For a numerical example involving 27 bodies (see Figure 8.1) which were discretized using a total of 500 000 degrees of freedom, it was found that within about an hour and a half, a solution with ten correct digits could be computed, see Section 7.

The paper is organized as follows: Section 2 describes the reduction of (1.1) to (1.3) and quantifies the error incurred by truncating the Fourier series. Section 3 presents the Nyström discretization of the reduced equations using high order quadrature applicable to kernels with integrable singularities, and the construction of the resulting linear systems. Section 4 summarizes the algorithm for the numerical solution of (1.3) and describes its computational costs. Section 5 presents the application of the algorithm for BIE formulations of Laplace's equation and describes the rapid calculation of k_n in this setting. Section 6 presents numerical examples applied to Laplace's equation. Section 7 extends our approach to multiply connected domains and Section 8 presents numerical results for this geometry. Section 9 gives conclusions and possible extensions and generalizations.

REMARK 1.1. *There is a variety of so called "fast" methods that could be used to solve (1.1) with $O(N(\log N)^q)$ complexity, e.g., the Fast Multipole Method [15], Panel Clustering [19], Adaptive Cross Approximation/$\mathcal{H}$-matrices [4], and $\mathcal{H}^2$-matrices [5]. Due to the extreme speed of the FFT, problems would have to be very large before "fast" methods would outperform the simplistic direct solver that we propose. Moreover, for problems involving axisymmetric domains that are sufficiently complicated that a very large number of discretization points along γ are needed, it would almost surely be advantageous to apply a "fast" method (e.g. [25]) to the reduced equations (1.3). In consequence, the fast kernel evaluations proposed here would still be of use.*

2. Fourier representation of BIE.

2.1. Problem formulation. Let Γ be a surface in $\mathbb{R}^3$ obtained by rotating a smooth contour γ about a fixed axis and consider the boundary integral equation

$$\sigma(\boldsymbol{x}) + \int_\Gamma k(\boldsymbol{x}, \boldsymbol{x}')\, \sigma(\boldsymbol{x}')\, dA(\boldsymbol{x}') = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Gamma. \tag{2.1}$$

In this section, we will demonstrate that if the kernel k is rotationally symmetric in a sense to be made precise, then by taking the Fourier transform in the azimuthal variable, (2.1) can be recast as a sequence of BIEs defined on the curve γ. To this

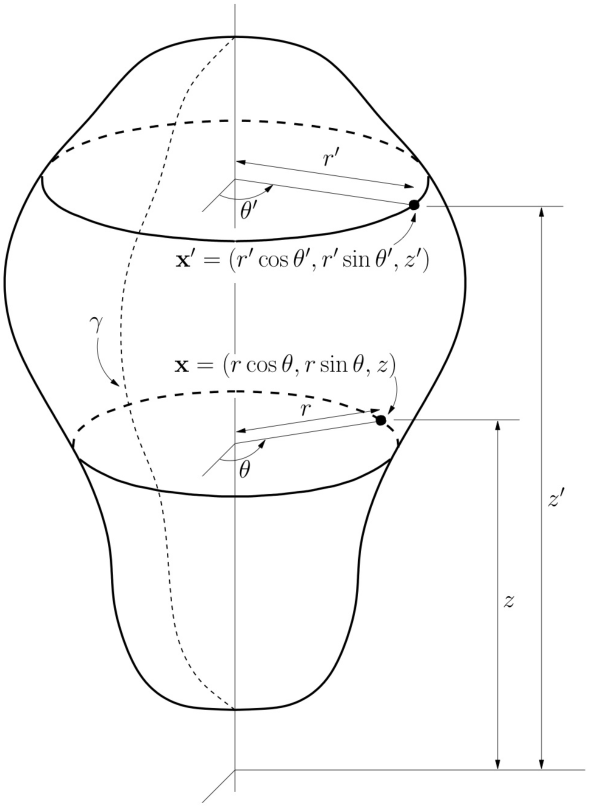

r'
θ'
$\mathbf{x}' = (r'\cos\theta', r'\sin\theta', z')$
γ
$\mathbf{x} = (r\cos\theta, r\sin\theta, z)$
r
θ
z'
z

FIG. 1.1. *The axisymmetric domain* Γ *generated by the curve* γ.

end, we introduce a Cartesian coordinate system in $\mathbb{R}^3$ with the third coordinate axis being the axis of symmetry. Then cylindrical coordinates (r, z, θ) are defined such that

$$\begin{aligned} x_1 &= r\,\cos\theta, \\ x_2 &= r\,\sin\theta, \\ x_3 &= z. \end{aligned}$$

Figure 1.1 illustrates the coordinate system.

The kernel k in (2.1) is now rotationally symmetric if for any two points $\boldsymbol{x}, \boldsymbol{x}' \in \Gamma$,

$$k(\boldsymbol{x}, \boldsymbol{x}') = k(\theta - \theta', r, z, r', z'), \tag{2.2}$$

where (θ', r', z') are the cylindrical coordinates of $\boldsymbol{x}'$.

2.2. Separation of variables. We define for $n \in \mathbb{Z}$ the functions f_n, σ_n, and k_n via

$$f_n(r,z) = \int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, f(\theta,r,z)\, d\theta, \tag{2.3}$$

$$\sigma_n(r,z) = \int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, \sigma(\theta,r,z)\, d\theta, \tag{2.4}$$

$$k_n(r,z,r',z') = \int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, k(\theta,r,z,r',z')\, d\theta. \tag{2.5}$$

The definitions (2.3), (2.4), and (2.5) define f_n, σ_n, and k_n as the coefficients in the Fourier series of the functions f, σ, and k about the azimuthal variable,

$$f(\boldsymbol{x}) = \sum_{n\in\mathbb{Z}} \frac{e^{in\theta}}{\sqrt{2\pi}}\, f_n(r,z), \tag{2.6}$$

$$\sigma(\boldsymbol{x}) = \sum_{n\in\mathbb{Z}} \frac{e^{in\theta}}{\sqrt{2\pi}}\, \sigma_n(r,z), \tag{2.7}$$

$$k(\boldsymbol{x},\boldsymbol{x}') = k(\theta-\theta',r,z,r',z') = \sum_{n\in\mathbb{Z}} \frac{e^{in(\theta-\theta')}}{\sqrt{2\pi}}\, k_n(r,z,r',z'). \tag{2.8}$$

To determine the Fourier representation of (2.1), we multiply the equation by $e^{-in\theta}/\sqrt{2\pi}$ and integrate θ over $\mathbb{T}$ (for our purposes, we can think of $\mathbb{T}$ as simply the interval $[-\pi,\,\pi]$). Equation (2.1) can then be said to be equivalent to the sequence of equations

$$\sigma_n(r,z) + \int_{\gamma\times\mathbb{T}} \left[\int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, k(\boldsymbol{x},\boldsymbol{x}')\, d\theta\right] \sigma(\boldsymbol{x}')\, dA(\boldsymbol{x}') = f_n(r,z), \qquad n\in\mathbb{Z}. \tag{2.9}$$

Invoking (2.8), we evaluate the bracketed factor in (2.9) as

$$\begin{aligned}\int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, k(\boldsymbol{x},\boldsymbol{x}')\, d\theta &= \int_{\mathbb{T}} \frac{e^{-in\theta}}{\sqrt{2\pi}}\, k(\theta-\theta',r,z,r',z')\, d\theta \\ &= e^{-in\theta'} \int_{\mathbb{T}} \frac{e^{-in(\theta-\theta')}}{\sqrt{2\pi}}\, k(\theta-\theta',r,z,r',z')\, d\theta = e^{-in\theta'}\, k_n(r,z,r',z').\end{aligned} \tag{2.10}$$

Inserting (2.10) into (2.9) and executing the integration of θ' over $\mathbb{T}$, we find that (2.1) is equivalent to the sequence of equations

$$\sigma_n(r,z) + \sqrt{2\pi}\int_{\gamma} k_n(r,z,r',z')\, \sigma_n(r',z')\, r'\, dl(r',z') = f_n(r,z), \quad n\in\mathbb{Z}. \tag{2.11}$$

For future reference, we define for $n\in\mathbb{Z}$ the boundary integral operators $\mathcal{K}_n$ via

$$[\mathcal{K}_n\, \sigma_n](r,z) = \sqrt{2\pi}\int_{\gamma} k_n(r,z,r',z')\, \sigma_n(r',z')\, r'\, dl(r',z'). \tag{2.12}$$

Then equation (2.11) can be written

$$\big(I + \mathcal{K}_n\big)\, \sigma_n = f_n, \qquad n\in\mathbb{Z}. \tag{2.13}$$

When each operator $I + \mathcal{K}_n$ is continuously invertible, we can write the solution of (2.1) as

$$\sigma(r, z, \theta) = \sum_{n\in\mathbb{Z}} \frac{e^{in\theta}}{\sqrt{2\pi}} [(I + \mathcal{K}_n)^{-1} f_n](r, z). \tag{2.14}$$

2.3. Truncation of the Fourier series. When evaluating the solution operator (2.14) in practice, we will choose a truncation parameter N_{F}, and evaluate only the lowest $2N_{\mathrm{F}} + 1$ Fourier modes. If N_{F} is chosen so that the given function f is well-represented by its lowest $2N_{\mathrm{F}} + 1$ Fourier modes, then in typical environments the solution obtained by truncating the sum (2.14) will also be accurate. To substantiate this claim, suppose that ε is a given tolerance, and that N_{F} has been chosen so that

$$||f - \sum_{n=-N_{\mathrm{F}}}^{N_{\mathrm{F}}} \frac{e^{in\theta}}{\sqrt{2\pi}} f_n|| \le \varepsilon. \tag{2.15}$$

We define an approximate solution via

$$\sigma_{\mathrm{approx}} = \sum_{n=-N_{\mathrm{F}}}^{N_{\mathrm{F}}} \frac{e^{in\theta}}{\sqrt{2\pi}} (I + \mathcal{K}_n)^{-1} f_n. \tag{2.16}$$

From Parseval's identity, we then find that the error in the solution satisfies

$$\begin{aligned} ||\sigma - \sigma_{\mathrm{approx}}||^2 &= \sum_{|n|>N_{\mathrm{F}}} ||(I + \mathcal{K}_n)^{-1} f_n||^2 \le \sum_{|n|>N_{\mathrm{F}}} ||(I + \mathcal{K}_n)^{-1}||^2\, ||f_n||^2 \\ &\le \left(\max_{|n|>N_{\mathrm{F}}} ||(I + \mathcal{K}_n)^{-1}||^2\right) \sum_{|n|>N_{\mathrm{F}}} ||f_n||^2 \le \left(\max_{|n|>N_{\mathrm{F}}} ||(I + \mathcal{K}_n)^{-1}||^2\right) \varepsilon^2. \end{aligned}$$

It is typically the case that the kernel $k(\boldsymbol{x}, \boldsymbol{x}')$ has sufficient smoothness such that the Fourier modes $k_n(r, z, r', z')$ decay as $n \to \infty$. Then $||\mathcal{K}_n|| \to 0$ as $n \to \infty$ and $||(I + \mathcal{K}_n)^{-1}|| \to 1$. Thus, an accurate approximation of f leads to an approximation in σ that is of the same order of accuracy. Figure 6.3 provides numerical evidence that when k is the double layer kernel associated with the Laplace equation and γ is a simple curve, the maximum and minimum singular values of $||I + \mathcal{K}_n||$ rapidly approach the same positive constant as n increases and implying that $||(I + \mathcal{K}_n)^{-1}||$ approaches a positive constant with rapid convergence.

3. Discretization of BIEs in two dimensions. The technique in Section 2 reduces the BIE (2.1) defined on an axisymmetric surface $\Gamma = \gamma \times \mathbb{T}$ contained in $\mathbb{R}^3$, to a sequence of BIEs defined on the curve γ contained in $\mathbb{R}^2$. These equations take the form

$$\sigma(\boldsymbol{x}) + \sqrt{2\pi} \int_\gamma k_n(\boldsymbol{x}, \boldsymbol{x}')\, \sigma(\boldsymbol{x}')\, r'\, dl(\boldsymbol{x}') = f(\boldsymbol{x}), \qquad \boldsymbol{x} \in \gamma, \tag{3.1}$$

where the kernel k_n is defined as in (2.5). In this section, we describe some standard techniques for discretizing an equation such as (3.1). For simplicity, we limit attention to the case where γ is a smooth closed curve, but extensions to non-smooth curves can be handled by slight variations of the techniques described here, *e.g.*, [6, 7, 21].

3.1. Parameterization of the curve. Let γ be parameterized by a vector-valued smooth function $\boldsymbol{\tau} : [0,T] \to \mathbb{R}^2$. The parameterization recasts (3.1) as an integral equation defined on the interval $[0,T]$:

$$\sigma(\boldsymbol{\tau}(t)) + \sqrt{2\pi} \int_0^T k_n(\boldsymbol{\tau}(t), \boldsymbol{\tau}(s))\, \sigma(\boldsymbol{\tau}(s))\, r'(\boldsymbol{\tau}(s))\, |d\boldsymbol{\tau}/ds|\, ds = f(\boldsymbol{\tau}(t)), \qquad (3.2)$$

where $t \in [0,T]$. To keep our formulas uncluttered, we suppress the parameterization of the curve and the dependence on n and introduce a new kernel

$$K(t,s) = \sqrt{2\pi}\, k_n(\boldsymbol{\tau}(t), \boldsymbol{\tau}(s))\, r'(\boldsymbol{\tau}(s))\, |d\boldsymbol{\tau}/ds|, \qquad (3.3)$$

as well as the functions

$$\varphi(t) = \sigma(\boldsymbol{\tau}(t)) \quad \text{and} \quad \psi(t) = f(\boldsymbol{\tau}(t)).$$

Then techniques for solving

$$\varphi(t) + \int_0^T K(t,s)\, \varphi(s)\, ds = \psi(t), \qquad t \in [0,T], \qquad (3.4)$$

where ψ is given and φ is to be determined, will be equally applicable to (3.2).

3.2. Nyström method. We will discretize (3.4) via Nyström discretization on standard Gaussian quadrature nodes, see [2]. To this end, we divide the interval $\Omega = [0,T]$ into a disjoint partition of $N_{\rm P}$ intervals,

$$\Omega = \bigcup_{p=1}^{N_{\rm P}} \Omega_p,$$

where each Ω_p is a subinterval called a *panel*. On each panel Ω_p, we place the nodes of a standard $N_{\rm G}$-point Gaussian quadrature rule $\{t_i^{(p)}\}_{i=1}^{N_{\rm G}}$. The idea is now to enforce (3.4) at each of the $N_{\rm P} N_{\rm G}$ nodes:

$$\sigma(t_i^{(p)}) + \int_0^T K(t_i^{(p)}, s)\, \varphi(s)\, ds = \psi(t_i^{(p)}), \qquad (i,p) \in \{1,2,\dots,N_{\rm G}\} \times \{1,2,\dots,N_{\rm P}\}.$$

To obtain a numerical method, suppose that we can construct for $p,q \in \{1,\, 2,\, \dots,\, N_{\rm P}\}$ and $i,j \in \{1,\, 2,\, \dots,\, N_{\rm G}\}$ numbers $A_{i,j}^{(p,q)}$ such that

$$\int_0^T K(t_i^{(p)}, s)\, \varphi(s)\, ds \approx \sum_{q=1}^{N_{\rm P}} \sum_{j=1}^{N_{\rm G}} A_{i,j}^{(p,q)}\, \varphi(t_j^{(q)}). \qquad (3.5)$$

Then the Nyström method is given by solving the linear system

$$\varphi_i^{(p)} + \sum_{q=1}^{N_{\rm P}} \sum_{j=1}^{N_{\rm G}} A_{i,j}^{(p,q)}\, \varphi_j^{(q)} = \psi_i^{(p)}, \qquad (i,p) \in \{1,2,\dots,N_{\rm G}\} \times \{1,2,\dots,N_{\rm P}\}, \quad (3.6)$$

where $\psi_i^{(p)} = \psi(t_i^{(p)})$ and $\varphi_i^{(p)}$ approximates $\varphi(t_i^{(p)})$. We write (3.6) compactly as

$$(I + A)\, \varphi = \psi,$$

where A is a matrix formed by $N_{\rm P} \times N_{\rm P}$ blocks, each of size $N_{\rm G} \times N_{\rm G}$. We let $A^{(p,q)}$ denote the block of A representing the interactions between the panels Ω_p and Ω_q.

3.3. Quadrature and interpolation.

We need to determine the numbers $A_{i,j}^{(p,q)}$ such that (3.5) holds. The detailed construction is given in Section 3.4, and utilizes some well-known techniques of quadrature and interpolation, which we review in this section.

3.3.1. Standard Gaussian quadratures.

Given an interval $[0,\, h]$ and a positive integer $N_{\rm G}$, the $N_{\rm G}$-point standard Gaussian quadrature rule consists of a set of $N_{\rm G}$ *nodes* $\{t_j\}_{j=1}^{N_{\rm G}} \subset [0,\, h]$, and $N_{\rm G}$ *weights* $\{w_j\}_{j=1}^{N_{\rm G}}$ such that

$$\int_0^h g(s)\, ds = \sum_{j=1}^{N_{\rm G}} w_j\, g(t_j),$$

whenever g is a polynomial of degree at most $2N_{\rm G} - 1$, see [1].

3.3.2. Quadrature rules for singular functions.

Now suppose that given an interval $[0,\, h]$ and a point $t \in [-h,\, 2h]$, we seek to integrate over $[0,\, h]$ functions g that take the form

$$g(s) = \phi_1(s) \log|s - t| + \phi_2(s), \tag{3.7}$$

where ϕ_1 and ϕ_2 are polynomials of degree at most $2N_{\rm G} - 1$. As shown in, *e.g.*, [10] and Section 5, the kernel's associated with Laplace's equation when formulated as in (2.5) contain such logarithmic singularities. This is also true of the kernel's associated with the Helmholtz equation, see for example, [11, 26]. Standard Gaussian quadrature would be highly inaccurate if applied to integrate (3.7). Rather, we seek a $N_{\rm G}'$-node quadrature that will evaluate

$$\int_0^h g(s)\, ds \tag{3.8}$$

exactly. Techniques for constructing such generalized quadratures are readily available in the literature, see for example [23]. These quadratures will be of degree $2N_{\rm G} - 1$, just as with standard Gaussian quadratures and exhibit comparable accuracy, although in general $N_{\rm G}' > N_{\rm G}$. The generalized quadratures used in this paper were determined using the techniques of [23], and can be found in [35].

We observe that the quadrature nodes constructed by such methods typically do not coincide with the nodes of the standard Gaussian quadrature of equivalent degree. This complicates the construction of the matrix A, as described in Section 3.4.

3.4. Constructing the matrix A.

Using the tools reviewed in Section 3.3, we are now in position to construct numbers $A_{i,j}^{(p,q)}$ such that (3.5) holds. We first note that in forming block $A^{(p,q)}$ of A, we need to find numbers $A_{i,j}^{(p,q)}$ such that

$$\int_{\Omega_q} K(t_i^{(p)}, s)\, \varphi(s)\, ds \approx \sum_{j=1}^{N_{\rm G}} A_{i,j}^{(p,q)}\, \varphi(t_j^{(q)}), \qquad i = 1,\, 2,\, \ldots,\, N_{\rm G}. \tag{3.9}$$

When Ω_p and Ω_q are well separated, the integrand in (3.9) is smooth, and our task is easily solved using standard Gaussian quadrature (as described in Section 3.3.1):

$$\int_{\Omega_q} K(t_i^{(p)}, s)\, \varphi(s)\, ds \approx \sum_{j=1}^{N_{\rm G}} w_j\, K(t_i^{(p)}, t_j^{(q)})\, \varphi(t_j^{(q)}).$$

It directly follows that the ij entry of the block $A^{(p,q)}$ takes the form

$$A^{(p,q)}_{i,j} = w_j\, K(t^{(p)}_i,\, t^{(q)}_j). \tag{3.10}$$

Complications arise when we seek to form a diagonal block $A^{(p,p)}$, or even a block that is adjacent to a diagonal block. The difficulty is that the kernel $K(t,s)$ has a singularity as $s \to t$. As discussed in Section 3.3.2, the singularity is logarithmic for the kernels considered in Section 5. To be precise, for any fixed t, there exist smooth functions u_t and v_t such that

$$K(t,s) = \log|t-s|\, u_t(s) + v_t(s).$$

We see that when $t^{(p)}_i$ is a point in Ω_q the integrand in (3.9) becomes singular. When $t^{(p)}_i$ is a point in a panel neighboring Ω_q, the problem is less severe, but Gaussian quadrature would still be inaccurate. To maintain full accuracy, we use the modified quadrature rules described in Section 3.3.2. For every node $t^{(p)}_i \in \Omega_p$, we construct a quadrature $\{\hat{w}^{(p,q)}_{i,\ell}, \hat{t}^{(p,q)}_{i,\ell}\}_{\ell=1}^{N'_{\rm G}}$ such that

$$\int_{\Omega_q} K(t^{(p)}_i, s)\, \varphi(s)\, ds \approx \sum_{\ell=1}^{N'_{\rm G}} \hat{w}^{(p,q)}_{i,\ell}\, K(t^{(p)}_i, \hat{t}^{(p,q)}_{i,\ell})\, \varphi(\hat{t}^{(p,q)}_{i,\ell}). \tag{3.11}$$

In order to have a quadrature evaluated at the Gaussian nodes $t^{(q)}_j \in \Omega_q$, we next use Lagrange interpolation. Let $\{L^{(q)}_j\}_{j=1}^{N_{\rm G}}$ denote the standard Lagrange interpolating functions of order $N_{\rm G}-1$ defined on Ω_q. Then

$$\varphi(t) \approx \sum_{j=1}^{N_{\rm G}} L^{(q)}_j(t)\, \varphi(t^{(q)}_j), \qquad t \in \Omega_q. \tag{3.12}$$

Inserting (3.12) into (3.11), we find that

$$\int_{\Omega_q} K(t^{(p)}_i, s)\, \varphi(s)\, ds \approx \sum_{\ell=1}^{N'_{\rm G}} \hat{w}^{(p,q)}_{i,\ell}\, K(t^{(p)}_i, \hat{t}^{(p,q)}_{i,\ell}) \sum_{j=1}^{N_{\rm G}} L^{(q)}_j(\hat{t}^{(p,q)}_{i,\ell})\, \varphi(t^{(q)}_j).$$

We now find that the block $A^{(p,q)}$ of A has entries

$$A^{(p,q)}_{i,j} = \sum_{\ell=1}^{N'_{\rm G}} \hat{w}^{(p,q)}_{i,\ell}\, K(t^{(p)}_i, \hat{t}^{(p,q)}_{i,\ell})\, L^{(q)}_j(\hat{t}^{(p,q)}_{i,\ell}), \qquad i,j \in \{1,2,\dots,N_G\}. \tag{3.13}$$

We observe that the formula (3.13) is expensive to evaluate; in addition to the summation, it requires the construction of a quadrature rule for each point $t^{(p)}_i$ and evaluation of Lagrange interpolants. Fortunately, this process must be executed only for matrix elements corresponding to interactions between a panel and itself, or a panel and its nearest neighbors, which constitutes a small band of entries about the diagonal of the matrix.

4. A general algorithm.

4.1. Summary. At this point, we have shown how to convert a BIE defined on an axisymmetric surface in $\mathbb{R}^3$ to a sequence of equations defined on a curve in $\mathbb{R}^2$ (Section 2), and then how to discretize each of these reduced equations (Section 3). Putting everything together, we obtain the following algorithm for solving (2.1):

1. Let the right hand side f, truncation parameter $N_{\rm F}$, and the discretization parameters $N_{\rm P}$ and $N_{\rm G}$ be given.

2. Form for $n = -N_{\rm F},\, -N_{\rm F} + 1,\, \dots,\, N_{\rm F}$ the matrix A_n discretizing the equation (2.13) encapsulating the n'th Fourier mode. The matrix is formed via Nyström discretization as described in Section 3 with the discretization parameters $N_{\rm P}$ and $N_{\rm G}$..

3. Evaluate via the FFT the terms $\{f_n\}_{n=-N_{\rm F}}^{N_{\rm F}}$ in the Fourier representation of f (as defined by (2.3)), and solve for $n = -N_{\rm F},\, -N_{\rm F}+1,\, -N_{\rm F}+2,\, \dots,\, N_{\rm F}$ the equation $(I + A_n)\,\sigma_n = f_n$ for σ_n. Construct $\sigma_{\rm approx}$ using formula (2.16) evaluated via the FFT.

The construction of the matrices A_n in Step 2 can be accelerated using the FFT (as described in Section 4.2), but even with such acceleration, it is typically a sizable percentage of the total cost of the algorithm. However, this step needs to be performed only once for any given geometry, and given discretization parameters $N_{\rm F}$, $N_{\rm P}$, and $N_{\rm G}$. The method therefore becomes particularly efficient when (2.1) needs to be solved for a sequence of right-hand sides. In this case, it may be worth the cost to pre-compute the inverse of each matrix $I + A_n$.

4.2. Techniques for forming the matrices. We need to construct for each Fourier mode n, a matrix A_n consisting of $N_{\rm P}\times N_{\rm P}$ blocks $A_n^{(p,q)}$, each of size $N_{\rm G}\times N_{\rm G}$. Constructing an off-diagonal block $A_n^{(p,q)}$ when Ω_p and Ω_q are not directly adjacent is relatively straightforward. For any pair of nodes $t_i^{(p)} \in \Omega_p$ and $t_j^{(q)} \in \Omega_q$, we need to construct the numbers, *cf.* (3.3) and (3.10),

$$A_{n;i,j}^{(p,q)} = \sqrt{2\pi}\, w_j\, k_n(\boldsymbol{\tau}(t_i^{(p)}),\, \boldsymbol{\tau}(t_j^{(q)}))\, r'(\boldsymbol{\tau}(t_j^{(q)}))\, |d\boldsymbol{\tau}(t_j^{(q)})/ds|, \tag{4.1}$$

for $n = -N_{\rm F},\, -N_{\rm F} + 1,\, \dots,\, N_{\rm F}$, where $\boldsymbol{\tau}$ is a parameterization of γ (see Section 3.1) and the kernel k_n is defined by (2.5). Fortunately, we do not need to explicitly evaluate the integrals in (2.5) since all the $2N_{\rm F} + 1$ numbers can be evaluated by a single application of the FFT to the function

$$\theta \mapsto k(\theta,\, \boldsymbol{\tau}(t_i^{(p)}),\, \boldsymbol{\tau}(t_j^{(q)})). \tag{4.2}$$

When $\boldsymbol{\tau}(t_i^{(p)})$ is not close to $\boldsymbol{\tau}(t_j^{(q)})$, the function in (4.2) is smooth, and the trapezoidal rule implicit in applying the FFT is highly accurate.

Evaluating the blocks on the diagonal, or directly adjacent to the diagonal is somewhat more involved. The matrix entries are now given by the formula, *cf.* (3.3) and (3.13),

$$A_{k;i,j}^{(p,q)} = \sum_{\ell=1}^{N'_{\rm G}} \hat{w}_{i,\ell}^{(p,q)}\, k_n(\boldsymbol{\tau}(t_i^{(p)}), \boldsymbol{\tau}(\hat{t}_{i,\ell}^{(p,q)}))\, r'(\hat{t}_{i,\ell}^{(p,q)})\, |d\boldsymbol{\tau}(\hat{t}_{i,\ell}^{(p,q)})/ds|\, L_j^{(q)}(\hat{t}_{i,\ell}^{(p,q)}), \tag{4.3}$$

where $\boldsymbol{\tau}$ and k_n are as in (4.1). To further complicate things, the points $\boldsymbol{\tau}(t_i^{(p)})$ and

$\boldsymbol{\tau}(\hat{t}_{i,\ell}^{(p,q)})$ are now in close proximity to each other, and so the functions

$$\theta \mapsto k(\theta,\, \boldsymbol{\tau}(t_i^{(p)}),\, \boldsymbol{\tau}(\hat{t}_{i,\ell}^{(p,q)})) \tag{4.4}$$

have a sharp peak around the point $\theta = 0$. They are typically still easy to integrate away from the origin, so the integrals in (2.5) can for a general kernel be evaluated relatively efficiently using quadratures that are adaptively refined near the origin.

Even with the accelerations described in this section, the cost of forming the matrices A_n tends to dominate the computation whenever the kernels k_n must be evaluated via formula (2.5). In particular environments, it is possible to side-step this problem by evaluating the integral in (2.5) analytically. That this can be done for the single and double layer kernels associated with Laplace's equation is demonstrated in Section 5.

4.3. Computational costs. The asymptotic cost of the algorithm described in Section 4.1 has three components: (a) the cost of forming the matrices $\{A_n\}_{n=-N_{\rm F}}^{N_{\rm F}}$, (b) the cost of transforming functions from physical space to Fourier space and back, and (c) the cost of solving the linear systems $(I + A_n)\,\sigma_n = f_n$. In this section, we investigate the asymptotic cost of these steps. We consider a situation where $N_{\rm F}$ Fourier modes need to be resolved, and where $N_{\rm P} N_{\rm G}$ nodes are used to discretize the curve γ. For simplicity of the presentation, we will assume that the parameter N_G is fixed, and set $N_\gamma = N_G N_P$. Further, where indicated we will make the assumption that $N_F \approx N_\gamma$ so that the total number of degrees of freedom used in the discretization is given by N, where $N^{1/2} = N_F = N_\gamma$.

(a) Cost of forming the linear systems: Suppose first that we have an analytic formula for each kernel k_n. (As we do, *e.g.*, when the original BIE (2.1) involves either the single or the double layer kernel associated with Laplace's equation, see Section 5.) Then the cost $T_{\rm mat}$ of forming the matrices satisfies

$$T_{\rm mat} \qquad \sim \qquad \underbrace{N_\gamma^2\, N_{\rm F}}_{\text{cost from kernel evaluations}} \qquad + \qquad \underbrace{N_\gamma^2\, N_{\rm F}}_{\text{cost from composite quadrature}} \qquad = N^{3/2}.$$

When the kernels have to be evaluated numerically via formula (2.5), the cost of forming the matrices is still moderate. In the rare situations where the kernel is smooth, standard Gaussian quadrature can be used everywhere and the FFT acceleration described in Section 4.2 can be used for all entries. In this situation,

$$T_{\rm mat} \sim N^{3/2} \log N$$

and note that the constant of proportionality associated with the use of the FFT is extremely small in this case.

In the more typical situation where each kernel k_n involves an integrable singularity at the diagonal, the FFT acceleration can still be used to rapidly evaluate all entries well-removed from the diagonal. However, entries close to the diagonal must be formed via the composite quadrature rule combined with numerical evaluation of k_n via an adaptive quadrature. The costs associated with this approach are also given by

$$T_{\rm mat} \sim N^{3/2} \log N,$$

but the constant of proportionality has the potential to be very large in this case due to the adaptive evaluation of the kernels.

(b) Cost of Fourier transforms: The boundary data defined on the surface must be converted into the Fourier domain, and once the solution is obtained in Fourier space, it must be brought back to physical space. This is executed via the FFT at a cost $T_{\rm fft}$ satisfying

$$T_{\rm fft} \sim N_\gamma\, N_{\rm F}\, \log(N_{\rm F}) = N\log(N). \tag{4.5}$$

We observe that the constant of proportionality in (4.5) is very small, and the cost of this step is typically negligible compared to the costs of the other steps.

(c) Cost of linear solves: Using standard Gaussian elimination, the cost $T_{\rm solve}$ of solving $N_{\rm F}$ linear systems $(I + A_n)\,\sigma_n = f_n$, each of size $N_\gamma \times N_\gamma$, satisfies

$$T_{\rm solve} \sim N_\gamma^3\, N_{\rm F} = N^2.$$

In situations where the equations need to be solved for multiple right hand sides, it is advantageous to first compute the inverses $(I + A_n)^{-1}$, and then simply apply these to each right hand side (or, alternatively, to form the LU factorizations, and then perform triangular solves). The cost $T_{\rm inv}$ of computing the inverses, and the cost $T_{\rm apply}$ of applying them then satisfy

$$T_{\rm inv} \sim N_\gamma^3\, N_{\rm F} = N^2$$
$$T_{\rm apply} \sim N_\gamma^2\, N_{\rm F} = N^{3/2}.$$

We make some practical observations:

- In practice, cost of forming the matrices by far dominates the other costs unless the kernel is either smooth, or analytic and/or fast methods for evaluating k_n are available.

- The scheme is highly efficient in situations where the same equation needs to be solved for a sequence of different right hand sides. Given an additional right hand side, the added cost $T_{\rm solve}$ is given by

  $$T_{\rm solve} \sim N\log(N) + N^{3/2},$$

  with a very small constant of proportionality. We note that this cost remains small even if an analytic formula for k_n is not available.

- The system matrices $I + A_n$ often have internal structure that allow them to be inverted using "fast methods" such as, *e.g.*, those in [25]. The cost of inversion and application can then be accelerated to near optimal complexity.

5. Simplifications for the double layer kernels associated with Laplace's equation.

5.1. The double layer kernels of Laplace's equation. Let $D \subseteq \mathbb{R}^3$ be a bounded domain whose boundary is given by a smooth surface Γ, let $E = \bar{D}^{\rm c}$ denote the domain exterior to D, and let $\boldsymbol{n}$ and be the outward unit normal to D. Consider the interior and exterior Dirichlet problems of potential theory [17],

$$\Delta u = 0 \;\text{ in }\; D, \quad u = f \;\text{ on }\; \Gamma, \qquad \text{(interior Dirichlet problem)} \tag{5.1}$$
$$\Delta u = 0 \;\text{ in }\; E, \quad u = f \;\text{ on }\; \Gamma, \qquad \text{(exterior Dirichlet problem)} \tag{5.2}$$

with the added condition that u tends towards zero uniformly for the exterior problem. The solutions to (5.1) and (5.2) can be written in the respective forms

$$\begin{aligned} u(\boldsymbol{x}) &= \int_\Gamma \frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}'), \quad \boldsymbol{x}\in D,\\ u(\boldsymbol{x}) &= \int_\Gamma \left(-\frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}+\frac{1}{4\pi|\boldsymbol{x}-\boldsymbol{x}_0|}\right)\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}'), \quad \boldsymbol{x}\in E,\ \ \boldsymbol{x}_0\in D, \end{aligned}$$

where σ is a boundary charge distribution that can be determined using the boundary conditions. The additional term in the kernel of the second equation above is included to enforce the proper decay rate of the solution at infinity, see *e.g.*, [17]. By taking the limit of these equations as $\boldsymbol{x}$ approaches the boundary, we arrive at the boundary integral equations

$$-\frac{1}{2}\sigma(\boldsymbol{x})+\int_\Gamma \frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}')=f(\boldsymbol{x}), \tag{5.3}$$

$$-\frac{1}{2}\sigma(\boldsymbol{x})+\int_\Gamma \left(-\frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}+\frac{1}{4\pi|\boldsymbol{x}-\boldsymbol{x}_0|}\right)\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}')=f(\boldsymbol{x}), \tag{5.4}$$

where $\boldsymbol{x}\in\Gamma$ in (5.3) and (5.4). The details of this derivation can be found, *e.g.*, in Chapter 8.7 of [17].

REMARK 5.1. *There are other integral formulations for the solution to Laplace's equation, e.g. Chapter 7 of [2] describes a variety of approaches. The double layer formulation presented here is a good choice in that it provides an integral operator that leads to well conditioned linear systems. However, the methodology of this paper is equally applicable to single-layer formulations that lead to first kind Fredholm BIEs.*

5.2. Separation of variables. Using the procedure given in Section 2, if $\Gamma = \gamma\times\mathbb{T}$, then (5.1) and (5.2) can be recast as a series of BIEs defined along γ. We express $\boldsymbol{n}$ in cylindrical coordinates as

$$\boldsymbol{n}(\boldsymbol{x}')=(n_{r'}\cos\theta', n_{r'}\sin\theta', n_{z'}).$$

Further,

$$\begin{aligned} |\boldsymbol{x}-\boldsymbol{x}'|^2 &= (r\cos\theta-r'\cos\theta')^2+(r\sin\theta-r'\sin\theta')^2+(z-z')^2\\ &= r^2+(r')^2-2rr'\cos(\theta-\theta')+(z-z')^2 \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}') &= (n_{r'}\cos\theta', n_{r'}\sin\theta', n_{z'})\cdot(r\cos\theta-r'\cos\theta', r\sin\theta-r'\sin\theta', z-z')\\ &= n_{r'}(r\cos(\theta-\theta')-r')+n_{z'}(z-z'). \end{aligned}$$

Then for a point $\boldsymbol{x}'\in\Gamma$, the kernel of the internal Dirichlet problem can be expanded as

$$\frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}=\frac{1}{\sqrt{2\pi}}\sum_{n\in\mathbb{Z}} e^{in(\theta-\theta')}d_n^{(i)}(r,z,r',z'),$$

where

$$d_n^{(i)}(r,z,r',z')=\frac{1}{\sqrt{32\pi^3}}\int_{\mathbb{T}} e^{-in\theta}\left[\frac{n_{r'}(r\cos\theta-r')+n_{z'}(z-z')}{(r^2+(r')^2-2rr'\cos\theta+(z-z')^2)^{3/2}}\right]d\theta.$$

Similarly, the kernel of the external Dirichlet problem can be written as

$$-\frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3}+\frac{1}{4\pi|\boldsymbol{x}-\boldsymbol{x}_0|}=\frac{1}{\sqrt{2\pi}}\sum_{n\in\mathbb{Z}}e^{in(\theta-\theta')}d_n^{(e)}(r,z,r',z'),$$

with

$$d_n^{(e)}(r,z,r',z')=\frac{1}{\sqrt{32\pi^3}}\int_{\mathbb{T}}e^{-in\theta}\Bigg(-\frac{n_{r'}(r\cos\theta-r')+n_{z'}(z-z')}{(r^2+(r')^2-2rr'\cos\theta+(z-z')^2)^{3/2}}+ \\ +\frac{1}{(r^2+r_0^2-2rr_0\cos\theta+(z-z_0)^2)^{1/2}}\Bigg)\,d\theta,$$

where $\boldsymbol{x}_0$ has been written in cylindrical coordinates as $(r_0\cos(\theta_0), r_0\sin(\theta_0), z_0)$. With the expansions of the kernels available, the procedure described in Section 4 can be used to solve (5.3) and (5.4) by solving

$$-\frac{1}{2}\sigma_n(r,z)+\sqrt{2\pi}\int_\gamma d_n^{(i)}(r,r',z,z')\sigma_n(r',z')\,r'\,dl(r',z')=f_n(r,z) \tag{5.5}$$

and

$$-\frac{1}{2}\sigma_n(r,z)+\sqrt{2\pi}\int_\gamma d_n^{(e)}(r,r',z,z')\sigma_n(r',z')\,r'\,dl(r',z')=f_n(r,z), \tag{5.6}$$

respectively for $n=-N_{\rm F},-N_{\rm F}+1,\dots,N_{\rm F}$. Note that the kernels $d_n^{(i)}$ and $d_n^{(e)}$ contain a log-singularity when both $r'=r$ and $z'=z$.

Equivalently, (5.5) and (5.6) can be arrived at by considering Laplace's equation written in cylindrical coordinates,

$$\frac{\partial^2 u}{\partial r^2}+\frac{1}{r}\frac{\partial u}{\partial r}+\frac{1}{r^2}\frac{\partial^2 u}{\partial\theta^2}+\frac{\partial^2 u}{\partial z^2}=0.$$

Taking the Fourier transform of u with respect to θ gives

$$\frac{\partial^2 u_n}{\partial r^2}+\frac{1}{r}\frac{\partial u_n}{\partial r}-\frac{n^2}{r^2}u_n+\frac{\partial^2 u_n}{\partial z^2}=0,\quad n\in\mathbb{Z},$$

where $e_n=e_n(\theta)=e^{in\theta}/\sqrt{2\pi}$ and $u=\sum_{n\in\mathbb{Z}}e_nu_n$. Then (5.5) and (5.6) are now associated with this sequence of PDEs.

5.3. Evaluation of kernels. The values of $d_n^{(i)}$ and $d_n^{(e)}$ for $n=-N_{\rm F},-N_{\rm F}+1,\dots,N_{\rm F}$ need to be computed efficiently and with high accuracy to construct the Nyström discretization of (5.5) and (5.6). Note that the integrands of $d_n^{(i)}$ and $d_n^{(e)}$ are real valued and even functions on the interval $[-\pi,\pi]$. Therefore, $d_n^{(i)}$ can be written as

$$d_n^{(i)}(r,z,r',z')=\frac{1}{\sqrt{32\pi^3}}\int_{\mathbb{T}}\left[\frac{n_{r'}(r\cos t-r')+n_{z'}(z-z')}{(r^2+(r')^2-2rr'\cos t+(z-z')^2)^{3/2}}\right]\cos(nt)\,dt. \tag{5.7}$$

Note that $d_n^{(e)}$ can be written in a similar form.

This integrand is oscillatory and increasingly peaked at the origin as both $r'\to r$ and $z'\to z$. As long as r' and r as well as z' and z are well separated, the integrand does not experience peaks near the origin, and as discussed before, the FFT provides a fast and accurate way for calculating $d_n^{(i)}$ and $d_n^{(e)}$.

In regimes where the integrand is peaked, the FFT no longer provides a means of evaluating $d_n^{(i)}$ and $d_n^{(e)}$ with the desired accuracy. One possible solution to this issue is applying adaptive quadrature to fully resolve the peak. However, this must be done for each value of n required and becomes prohibitively expensive if N_{F} is large.

Fortunately, an analytical solution to (5.7) exists. As noted in [10], the single-layer kernel can be expanded with respect to the azimuthal variable as

$$\begin{aligned} s(\boldsymbol{x},\boldsymbol{x}') = \frac{1}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|} &= \frac{1}{4\pi(r^2+(r')^2-2rr'\cos(\theta-\theta')+(z-z')^2)^{1/2}} \\ &= \frac{1}{\sqrt{2\pi}}\sum_{n\in\mathbb{Z}} e^{in(\theta-\theta')} s_n(r,z,r',z'), \end{aligned}$$

where

$$\begin{aligned} s_n(r,z,r',z') &= \frac{1}{\sqrt{32\pi^3}}\int_{\mathbb{T}} \frac{\cos(nt)}{(r^2+(r')^2-2rr'\cos(t)+(z-z')^2)^{1/2}}\,dt \\ &= \frac{1}{\sqrt{8\pi^3 rr'}}\int_{\mathbb{T}} \frac{\cos(nt)}{\sqrt{8(\chi-\cos(t))}}\,dt \\ &= \frac{1}{\sqrt{8\pi^3 rr'}}\mathcal{Q}_{n-1/2}(\chi), \end{aligned}$$

$\mathcal{Q}_{n-1/2}$ is the half-integer degree Legendre function of the second kind, and

$$\chi = \frac{r^2+(r')^2+(z-z')^2}{2rr'}.$$

To find an analytical form for (5.7), first note that in cylindrical coordinates the double-layer kernel can be written in terms of the single-layer kernel,

$$\begin{aligned} \frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{4\pi|\boldsymbol{x}-\boldsymbol{x}'|^3} =& \frac{n_{r'}(r\cos(\theta-\theta')-r')+n_{z'}(z-z')}{4\pi(r^2+(r')^2-2rr'\cos(\theta-\theta')+(z-z')^2)^{3/2}} \\ =& \frac{1}{4\pi}\left[n_{r'}\frac{\partial}{\partial r'}\left(\frac{1}{(r^2+(r')^2-2rr'\cos(\theta-\theta')+(z-z')^2)^{1/2}}\right)+\right. \\ &\left. + n_{z'}\frac{\partial}{\partial z'}\left(\frac{1}{(r^2+(r')^2-2rr'\cos(\theta-\theta')+(z-z')^2)^{1/2}}\right)\right]. \end{aligned}$$

The coefficients of the Fourier series expansion of the double-layer kernel are then given by $d_n^{(i)}$, which can be written using the previous equation as

$$\begin{aligned} d_n^{(i)}(r,z,r',z') =& n_{r'}\int_{\mathbb{T}}\frac{\partial}{\partial r'}\left(\frac{\cos(nt)}{(32\pi^3(r^2+(r')^2-2rr'\cos(t)+(z-z')^2))^{1/2}}\right)dt+ \\ &+ n_{z'}\int_{\mathbb{T}}\frac{\partial}{\partial z'}\left(\frac{\cos(nt)}{(32\pi^3(r^2+(r')^2-2rr'\cos(t)+(z-z')^2))^{1/2}}\right)dt \\ =& n_{r'}\frac{\partial}{\partial r'}\left(\frac{1}{\sqrt{8\pi^3 rr'}}\mathcal{Q}_{n-1/2}(\chi)\right)+n_{z'}\frac{\partial}{\partial z'}\left(\frac{1}{\sqrt{8\pi^3 rr'}}\mathcal{Q}_{n-1/2}(\chi)\right) \\ =& \frac{1}{\sqrt{8\pi^3 rr'}}\left[n_{r'}\left(\frac{\partial\mathcal{Q}_{n-1/2}(\chi)}{\partial\chi}\frac{\partial\chi}{\partial r'}-\frac{\mathcal{Q}_{n-1/2}(\chi)}{2r'}\right)+n_{z'}\frac{\partial\mathcal{Q}_{n-1/2}(\chi)}{\partial\chi}\frac{\partial\chi}{\partial z'}\right]. \end{aligned}$$

To utilize this form of $d_n^{(i)}$, set $\mu = \sqrt{\frac{2}{\chi+1}}$ and note that

$$
\begin{aligned}
&\frac{\partial \chi}{\partial r'} = \frac{(r')^2 - r^2 - (z-z')^2}{2r(r')^2},\\
&\frac{\partial \chi}{\partial z'} = \frac{z'-z}{rr'},\\
&\mathcal{Q}_{-1/2}(\chi) = \mu K(\mu),\\
&\mathcal{Q}_{1/2}(\chi) = \chi\mu K(\mu) - \sqrt{2(\chi+1)}E(\mu),\\
&\mathcal{Q}_{-n-1/2}(\chi) = \mathcal{Q}_{n-1/2}(\chi),\\
&\mathcal{Q}_{n-1/2}(\chi) = 4\frac{n-1}{2n-1}\chi\mathcal{Q}_{n-3/2}(\chi) - \frac{2n-3}{2n-1}\mathcal{Q}_{n-5/2}(\chi),\\
&\frac{\partial \mathcal{Q}_{n-1/2}(\chi)}{\partial \chi} = \frac{2n-1}{2(\chi^2-1)}\left(\chi\mathcal{Q}_{n-1/2} - \mathcal{Q}_{n-3/2}\right),
\end{aligned}
$$

where K and E are the complete elliptic integrals of the first and second kinds, respectively. The first two relations follow immediately from the definition of χ and the relations for the Legendre functions of the second kind can be found in [1]. With these relations in hand, the calculation of $d_n^{(i)}$ for $n = -N_{\rm F}, -N_{\rm F}+1, \ldots, N_{\rm F}$ can be done accurately and efficiently when r' and r as well as z' and z are in close proximity. The calculation of $d_n^{(e)}$ can be done analogously.

REMARK 5.2. *Note that the forward recursion relation for the Legendre functions $\mathcal{Q}_{n-1/2}(\chi)$ is unstable when $\chi > 1$. In practice, the instability is mild when χ is near 1 and can still be employed to accurately compute values in this regime. Additionally, if stability becomes and issue, Miller's algorithm [13] can be used to calculate the values of the Legendre functions using the backwards recursion relation, which is stable for $\chi > 1$.*

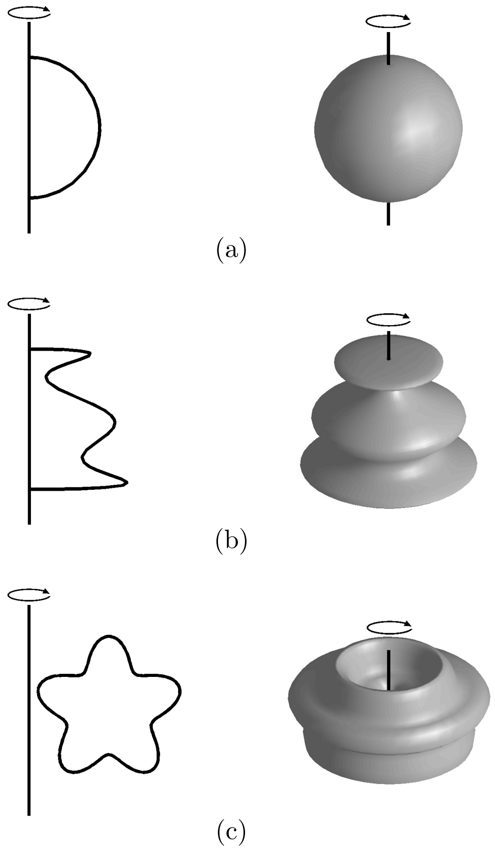
6. Numerical results.

This section describes several numerical experiments performed to assess the efficiency and accuracy of the the numerical scheme outlined in Section 4.1. All experiments were executed for the double layer kernels associated with Laplace's equation, calculated using the technique described in Section 5.3. Note that the kernels in this case give us the property that $A_{-n} = A_n$, and so we need only to invert $N_{\rm F} + 1$ matrices. The geometries investigated are described in Figure 6.1. The generating curves were parameterized by arc length, and split into $N_{\rm P}$ panels of equal length. A 10-point Gaussian quadrature has been used along each panel, with the modified quadratures used to handle the integrable singularities in the kernel. These quadratures are listed in [35]. The algorithm was implemented in FORTRAN, using BLAS, LAPACK, and the FFT library provided by Intel's MKL library. All numerical experiments have been carried out on a Macbook Pro with a 2.5 GHz Intel Core 2 Duo and 2GB of RAM.

6.1. Computational costs.

Using the domain in Figure 6.1(a) and the interior Dirichlet problem, timing results are given in Table 6.1. The reported results include:

$N_{\rm P}$	the number of panels used to discretize the contour
$N_{\rm F}$	the Fourier truncation parameter (we keep $2N_{\rm F}+1$ modes)
$T_{\rm mat}$	time to construct the linear systems (as described in Section 5.3)
$T_{\rm inv}$	time to invert the linear systems
$T_{\rm fft}$	time to Fourier transform the right hand side and the solution
$T_{\rm apply}$	time to apply the inverse to the right hand side



(a)

(b)

(c)

Fig. 6.1. *Domains used in numerical examples. All items are rotated about the vertical axis. (a) A sphere. (b) A wavy block. (c) A starfish torus.*

The most expensive component of the calculation is the construction of the linear systems and their inversion. This is primarily a result of the cost of evaluating the kernel and applying the modified quadrature rules, and the use of dense matrix algebra to invert the matrices. Table 6.2 compares the use of the recursion relation in evaluating the kernel when it is near-singular to using an adaptive Gaussian quadrature. The efficiency of the recursion relation is clearly evident in this case. The timings for the analytic/FFT evaluation of the kernel experience a large speed up as χ approaches 1 due to the algorithm switching over from the FFT to the recursion relation. In general, a gain of 2 to 4 orders of magnitude in speed is obtained in using the methodology of Section 5.3 in evaluating the kernels in comparison to using adaptive quadrature.

Figure 6.2 plots the time to construct the linear systems as the number of degrees of freedom $N = N_\gamma(2N_F + 1)$ increases, for the case when $N_\gamma \approx 2N_F + 1$. The estimated asymptotic costs given in Figure 6.2 match well with the estimates derived in Section 4.3. It is also clear that as N grows, the cost of inversion will eventually dominate. We remark that this cost can be greatly lowered by using fast techniques for the inversion of boundary integral operators, but the algorithm is quite fast for the problem sizes considered here.

We observe that the largest problem reported in Table 6.1 involves 320 800 degrees of freedom. The method requires 57 seconds of pre-computation for this example, and is then capable of computing a solution u from a given data function f in 0.39 seconds.

$N_{\rm P}$	$2N_{\rm F}+1$	$T_{\rm mat}$	$T_{\rm inv}$	$T_{\rm fft}$	$T_{\rm apply}$
5	25	1.63e-02	1.36e-03	6.97e-05	3.80e-05
10	25	3.38e-02	5.65e-03	1.37e-04	1.40e-04
20	25	9.30e-02	3.24e-02	2.85e-04	8.61e-04
40	25	3.08e-01	2.19e-01	6.31e-04	6.05e-03
80	25	1.05e+00	1.62e+00	1.17e-03	2.06e-02
5	51	2.43e-02	2.62e-03	2.00e-04	6.54e-05
10	51	5.94e-02	1.12e-02	3.93e-04	2.57e-04
20	51	1.84e-01	6.55e-02	8.11e-04	3.10e-03
40	51	6.57e-01	4.36e-01	1.69e-03	1.08e-02
80	51	2.38e+00	3.26e+00	3.24e-03	4.67e-02
5	101	5.18e-02	5.02e-03	8.18e-04	1.25e-04
10	101	1.48e-01	2.18e-02	1.64e-03	6.77e-04
20	101	4.99e-01	1.28e-01	3.35e-03	5.83e-03
40	101	1.84e+00	8.58e-01	6.71e-03	2.00e-02
80	101	7.79e+00	6.36e+00	1.33e-02	8.30e-02
5	201	9.96e-02	9.80e-03	1.65e-03	2.49e-04
10	201	3.06e-01	4.42e-02	3.38e-03	3.16e-03
20	201	1.06e+00	2.55e-01	6.77e-03	1.04e-02
40	201	3.98e+00	1.69e+00	1.37e-02	3.93e-02
80	201	1.67e+01	1.26e+01	2.83e-02	1.61e-01
5	401	1.74e-01	1.95e-02	3.16e-03	6.97e-04
10	401	5.49e-01	8.80e-02	6.47e-03	5.94e-03
20	401	1.91e+00	5.07e-01	1.29e-02	1.94e-02
40	401	7.96e+00	3.37e+00	2.68e-02	7.66e-02
80	401	3.17e+01	2.56e+01	5.58e-02	3.35e-01

Table 6.1

Timing results in seconds performed for the domain given in Figure 6.1(a) for the interior Dirichlet problem.

χ	Adaptive Quadrature (s)	Analytic Form (s)	Ratio
1.1	9.89E-02	7.61E-05	1.30E+03
1.01	1.01E-01	3.59E-04	2.81E+02
1.001	1.03E-01	3.60E-04	2.87E+02
1.0001	1.06E-01	9.21E-06	1.16E+04

Table 6.2

Timings comparing the use of adaptive quadrature and the technique of Section 5.3 for evaluating the kernels. For fixed (r, z) and (r', z'), we evaluate the kernels $\{k_n(r, z, r', z')\}_{n=0}^{201}$ corresponding to the single and double layer kernels of Laplace's equation via adaptive quadrature and the analytic scheme as described in Section 5.3 for various values of χ. Note that the integrand is increasingly peaked as $\chi \to 1$. All coefficients are computed with (at least) 10 digits of accuracy.

6.2. Accuracy and conditioning of discretization. The accuracy of the discretization has been tested using the interior and exterior Dirichlet problems on the domains given in Figure 6.1. Exact solutions were generated by placing a few random

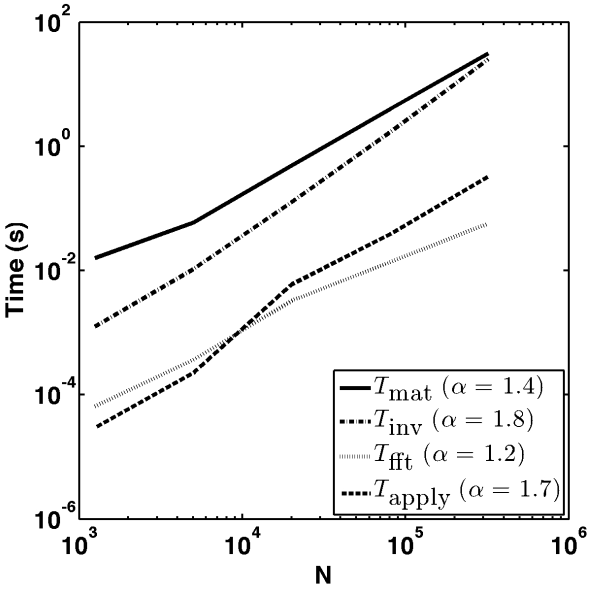


FIG. 6.2. *Timings of the algorithm as the number of degrees of freedom $N = N_\gamma(2N_F + 1)$ increases. The timings reported here are for the case $N_\gamma \approx 2N_F + 1$. We assume that the computational cost takes the form $T \sim O(N^\alpha)$.*

point charges outside of the domain where the solution was calculated. The solution was evaluated at random points defined on a sphere encompassing (or interior to) the boundary. The errors reported in Tables 6.3, 6.4, and 6.5 are relative errors measured in the l^∞-norm, $||u_\epsilon - u||_\infty/||u||_\infty$, where u is the exact potential and u_ϵ is the potential obtained from the numerical solution.

In all cases, 10 digits of accuracy has been obtained from a discretization involving a relatively small number of panels, due to the rapid convergence of the Gaussian quadrature. The convergence is rapid enough to make it difficult to numerically estimate, but the order of the method is roughly $(1/N_{\text{P}})^{N_{\text{G}}}$ given the high order quadrature used. The number of Fourier modes required to obtain 10 digits of accuracy is on the order of 100 modes. Although not investigated here, the discretization technique naturally lends itself to nonuniform refinement of the surface, allowing one to resolve features of the surface that require finer resolution.

The number of correct digits obtained as the number of panels and number of Fourier modes increases eventually stalls. This is a result of a loss of precision in determining the kernels, as well as cancelation errors incurred when evaluating interactions between nearby points. This is especially prominent with the use of Gaussian quadratures, as points cluster near the ends of the panels. If more digits are required, high precision arithmetic can be employed in the setup phase of the algorithm.

Figure 6.3 shows the maximum and minimum singular values of an 80 panel discretization for the $N_{\text{F}} = -200, \dots, 200$ Fourier modes used in the discretization of the interior Dirichlet problem, on the domain shown in Figure 6.1(a). The integral equations of this paper are second kind Fredholm equations, and generally lead to well-conditioned systems. As seen in Figure 6.3, this hold true for the discretization presented in this paper.

N_{P}	$2N_{\mathrm{F}}+1$				
-	25	50	100	200	400
5	1.93869e-04	4.10935e-07	5.37883e-08	5.37880e-08	5.37880e-08
10	1.93869e-04	4.10513e-07	3.27169e-12	6.72270e-13	6.72270e-13
20	1.93869e-04	4.10513e-07	3.30601e-12	1.66132e-13	1.66132e-13
40	1.93869e-04	4.10513e-07	3.23162e-12	8.28568e-14	8.28568e-14
80	1.93869e-04	4.10512e-07	2.92918e-12	2.92091e-13	2.92091e-13

Table 6.3
Error in internal Dirichlet problem solved on domain (a) in Figure 6.1.

N_{P}	$2N_{\mathrm{F}}+1$				
-	25	50	100	200	400
5	9.11452e-04	9.11464e-04	9.11464e-04	9.11464e-04	9.11464e-04
10	4.15377e-05	4.15416e-05	4.15416e-05	4.15416e-05	4.15416e-05
20	6.31923e-07	1.29234e-07	1.29235e-07	1.29235e-07	1.29235e-07
40	7.04741e-07	3.10049e-11	3.08152e-11	3.08305e-11	3.08359e-11
80	7.04779e-07	5.62558e-11	5.05306e-11	5.05257e-11	5.05232e-11

Table 6.4
Error in external Dirichlet problem solved on domain (b) in Figure 6.1.

N_{P}	$2N_{\mathrm{F}}+1$				
-	25	50	100	200	400
5	3.80837e-04	3.83707e-04	3.83707e-04	3.83707e-04	3.83707e-04
10	2.41602e-05	6.81564e-06	6.81556e-06	6.81556e-06	6.81556e-06
20	3.03272e-05	5.98506e-09	2.53980e-11	2.54112e-11	2.54118e-11
40	3.03272e-05	6.01273e-09	6.95662e-12	6.94592e-12	6.94546e-12
80	3.03272e-05	6.01059e-09	5.25217e-12	5.26674e-12	5.26515e-12

Table 6.5
Error in external Dirichlet problem solved on domain (c) in Figure 6.1.

7. Extension to multiply connected domains. We have so far restricted ourselves to the solution of boundary integral equations defined on a single surface in space. In this section, we extend the results of the previous sections to multiply connected domains. Specifically, we utilize high order Gaussian quadrature to discretize each surface, and then solve the system via an iterative solver that is accelerated with the Fast Multipole Method. We will restrict our attention to the exterior Dirichlet problem, but the treatment of other integral equations, boundary conditions, and domains is analogous.

7.1. Formulation. Multiply connected domains introduce an additional complication in obtaining the solution of Laplace's equation via a boundary integral approach. W denote the number of bodies defining the boundary Γ as N_B, so that $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \ldots \cup \Gamma_{N_B}$, and assume that the bodies are well separated. We will also assume that each of these bodies individually satisfies the axisymmetric constraint, but they need not be oriented in the same direction or along the same axis. It is well known that for multiply connected domains, the boundary integral operator defined

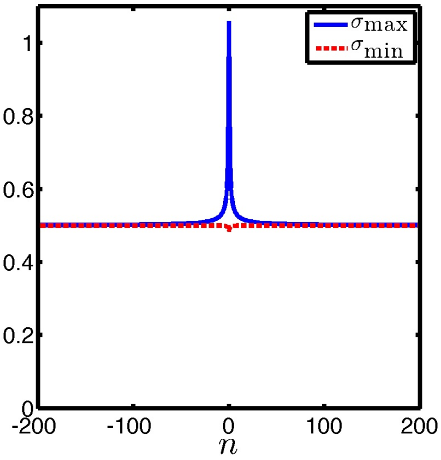


Fig. 6.3. *Maximum and minimum singular values for the matrices resulting from an 80 panel discretization of a sphere using 401 Fourier modes, where n is the the matrix associated with the n^{th} Fourier mode.*

by (5.4) has a nontrivial null space of dimension $N_B - 1$ [27]. A standard approach to remove this null space is provided by variants of Mikhlin's method, *c.f.* [14, 22, 34].

Alternatively, one can represent the solution to Laplace's equation on a multiply connected domain at a point $\boldsymbol{x}$ outside of the bodies by a combination of a single and a double layer [22],

$$u(\boldsymbol{x}) = \int_\Gamma \frac{1}{4\pi}\left(\frac{1}{|\boldsymbol{x}-\boldsymbol{x}'|} + \frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{|\boldsymbol{x}-\boldsymbol{x}'|^3}\right)\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}').$$

The resulting boundary integral equation is given by

$$\frac{1}{2}\sigma(\boldsymbol{x}) + \int_\Gamma \frac{1}{4\pi}\left(\frac{1}{|\boldsymbol{x}-\boldsymbol{x}'|} + \frac{\boldsymbol{n}(\boldsymbol{x}')\cdot(\boldsymbol{x}-\boldsymbol{x}')}{|\boldsymbol{x}-\boldsymbol{x}'|^3}\right)\sigma(\boldsymbol{x}')\,dA(\boldsymbol{x}') = f(\boldsymbol{x}). \tag{7.1}$$

This is the approach we take, as it leads to particularly simple linear systems and is effective for a moderate number of bodies.

7.2. Construction of the linear system. There is no straightforward way to decouple the multiply connected problem into a series of integral equations defined on a generating curve, as the axes of symmetry for each body are not necessarily aligned. Instead, we consider a Nyström discretization of the surfaces defining the boundary Γ. Let $K_{i,j}$ denote the interactions between the surfaces Γ_i and Γ_j. Then (7.1) can be written as

$$\frac{1}{2}\sigma I + \begin{pmatrix} K_{1,1} & K_{1,2} & \cdots & K_{1,N_B} \\ K_{2,1} & K_{2,2} & \cdots & K_{2,N_B} \\ \vdots & \vdots & \ddots & \vdots \\ K_{N_B,1} & K_{N_B,2} & \cdots & K_{N_B,N_B} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_{N_B} \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N_B} \end{pmatrix}, \tag{7.2}$$

where σ_i and f_i are the charge potential and boundary conditions on the i^{th} body. The blocks $K_{i,j}$ when $i \neq j$ represent integral operators with smooth kernels, and thus any standard quadrature can be used to discretize these blocks in (7.2). The natural choice in light of the discretization used previously in this paper is to discretize each body with a standard Gaussian quadrature along the body's generating curve, and to use the trapezoidal rule in the other direction. This gives a highly accurate representation of $K_{i,j}$ when $i \neq j$, and leads to $N = N_B N_\gamma (2N_F + 1)$ unknowns in the linear system, assuming each body is discretized with the same number of points.

Unfortunately, using this discretization on the diagonal blocks $K_{i,i}$, for $i = 1, \dots, N_B$, does not represent the self interactions of the i^{th} body in an accurate manner; the kernel is singular in this regime. To construct a high order discretization of the diagonal blocks in (7.2), we will use the technique described in Section 3 coupled with the Fourier representation of the BIE given in Section 2. Let $\hat{\mathcal{K}}_{i,i}$ be the block-diagonal operator whose diagonal blocks are given by the operators $\mathcal{K}_n$ defined by (2.12), for the surface Γ_i. Then the action of the diagonal block $K_{i,i}$ in (7.2) can be represented by

$$K_{i,i} = \mathcal{F}_i^{-1} \hat{\mathcal{K}}_{i,i} \mathcal{F}_i, \tag{7.3}$$

where the periodic Fourier transform $\mathcal{F}_i : L^2(\mathbb{T}) \mapsto \ell^2(\mathbb{Z})$ maps a function's azimuthal dependence to its azimuthal Fourier coefficients. The expression given by (7.3) is discretized by the appropriate number of Fourier modes and panels to ensure that the nodes coincide with those resulting from the discretization of $K_{i,j}$ when $i \neq j$.

Note that the kernels in (7.3) can be calculated efficiently using the procedure described in Section 5.3.

7.3. Solution to the linear system. In general, the linear system defined by (7.2) is of size $N = N_B N_\gamma (2N_F + 1)$, assuming each body is discretized with the same number of panels and Fourier modes, and is too large to be explicitly formed and inverted. Further, since the system of equations cannot be formulated in its entirety with the Fourier representation used when the domain is simply connected, we do not have access to the inverse of the matrix in (7.2) as given by (2.14).

However, because the off diagonal blocks in (7.2) are discretized by a conventional Nyström method in $\mathbb{R}^3$, this portion of the matrix can be applied to a vector with a conventional fast multipole method in $O(N)$ operations, *c.f.* [9, 16].

The diagonal blocks of the matrix can be applied to a vector using (7.3). This costs $O(N_B N_\gamma^2 N_F \log N_F)$ operations. Asymptotically, this cost would dominate the cost of applying the matrix, but in practice, the high accuracy of the discretization results in a small number of unknowns on each body and as shown in Section 8, the FMM is the primary bottleneck in the application of the matrix.

With access to a fast matrix-vector multiply for applying the operator in (7.2), iterative methods become accessible for obtaining the solution. Further, since we are solving a second kind Fredholm equation containing a compact operator, the resulting discretized system will be well conditioned and an iterative solver like GMRES will converge in a small number of iterations (at least when the number of bodies is of moderate size).

7.4. An algorithm for multiply connected domains. A summary of the algorithm described in Sections 7.1-7.3 is given below:

1. Let a right hand side f and discretization parameters N_F, N_G, and N_P be given for each body $i = 1, \dots, N_B$.

2. For each body $i = 1, \ldots, N_B$, form the block diagonal matrices as given by (7.3), using the techniques described in Section 4 and the accelerations given in Section 5.3 for calculating the kernels.

3. Using an iterative solver appropriate for nonsymmetric systems, solve the system given by (7.2). Apply the off diagonal component of the matrix with the FMM, and apply the diagonal blocks directly as described in Section 7.3.

We make no claim that this algorithm is optimal for multiply connected domains. Certainly, the low-rank of the interactions between bodies could be further exploited by techniques such as skeletonization [25] to further increase the speed. However, we have found the algorithm described in this section is very simple to implement and effective enough for problems of reasonable size.

8. Numerical results for multiply connected domains. The algorithm described in Section 7.4 has been implemented in FORTRAN, using BLAS, LAPACK, and the FFT library provided by Intel's MKL library. The adaptive fast multiple algorithm described in [9] has been utilized in applying the off diagonal parts of a matrix, and GMRES is employed to solve the linear system of equations. All numerical experiments in this section have been carried out on a Macbook Pro with a 2.5 GHz Intel Core 2 Duo and 2GB of RAM.

Table 8.1 gives the results for the exterior Dirichlet problem for the domain given in Figure 8.1. The domain consists of 27 ellipses contained in the box $[0, 6.1] \times [0, 6.1] \times [0, 6.1]$, where each ellipse has a major axis of length 2, and a minor axis of length 1. The minimal distance between any two ellipses is 0.05. We solve the linear system via GMRES until the residual is less than 10^{-9}, applying the matrix as described in Section 7.3. The boundary conditions are generated by placing point charges inside the ellipses, and the solution is evaluated at 10 000 random points in the exterior domain.

Note that this geometry consists of one surface that is repeatedly translated and rotated throughout the domain. This allows us to construct the discretization representing the self interactions of an ellipse just once, and repeatedly use it in applying the block-diagonal of the matrix in (7.2), greatly reducing the memory requirements. The parameters in Table 8.1 are:

N	the total number of unknowns in the system
N_{BODY}	the number of points discretizing each body
T_{diag}	time (in seconds) to apply the diagonal blocks to a vector
$T_{\text{off-diag}}$	time (in seconds) to apply the off diagonal blocks to a vector via the FMM
I_{GMRES}	The number of GMRES iterations required to reduce the residual to 10^{-9}
T_{GMRES}	time (in seconds) to solution via GMRES
Error	the maximal relative error in evaluating the solution

N	N_{BODY}	T_{setup}	T_{diag}	$T_{\text{off-diag}}$	I_{GMRES}	T_{GMRES}	Error
542700	20100	6.72e-01	1.43e-01	2.27e02	22	5.00e03	1.57e-10

TABLE 8.1
Results for the domain given in Figure 8.1 for the exterior Dirichlet problem.

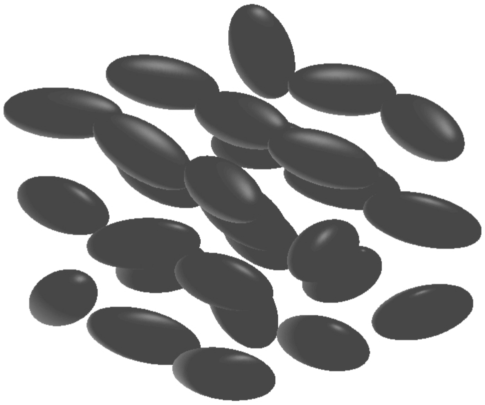


Fig. 8.1. *Randomly oriented ellipses contained in the box* $[0, 6.1] \times [0, 6.1] \times [0, 6.1]$*. Each ellipse has a major axis of length 2, and a minor axis of length 1. The minimal distance between any two ellipses is 0.05.*

Several comments are in order. We observe that the time to apply the block diagonal of the matrix is 3 orders of magnitude faster than the time to apply the off diagonal blocks via the FMM. This confirms the comments made in Section 7.3; for practical situations, the FMM dominates the dominate cost of the algorithm. The primary reason for this is the high accuracy of the discretization. We require only a small number of panels and Fourier modes to discretize each individual body, and this results in diagonal blocks that can be rapidly applied via the FFT. The small size of these matrices results in a rapid execution time. We remark that the cost of the FMM as implemented can be roughly halved. This is because the algorithm is called twice per iteration, once to apply the full matrix, and again to remove contributions from the diagonal blocks that it incorrectly calculated.

We have tested the algorithm on a variety of geometrical domains. The results are similar in nature; in general the number of GMRES iterations required mildly increases with the number of bodies and also increases if the bodies are in close proximity. Note that a finer discretization is required if the bodies are extremely close together.

9. Generalizations and conclusions.

This paper describes a numerical technique for computing solutions to boundary integral equations defined on axisymmetric surfaces in $\mathbb{R}^3$ with no assumption on the loads being axisymmetric. The technique is introduced as a generic method with only very mild conditions imposed on the kernel; specifically, we assume that the kernel has an integrable singularity at the diagonal, and that it is rotationally symmetric (in the sense that (2.2) holds). New contributions of our work include:

1. A highly accurate quadrature scheme for kernels with integrable singularities is introduced. Numerical experiments indicate that solutions with a relative accuracy of 10^{-10} or better can easily be constructed.
2. A rapid technique for numerically constructing the kernel functions k_n in (2.11) is introduced. It works when k is either the single or the double layer potential associated with Laplace's equation. The technique is a hybrid scheme that relies on the FFT when possible, and uses recursion relations for Legendre functions when not. The resulting scheme is fast enough that

a problem involving 320 800 degrees of freedom can be solved in less than a minute on a standard desktop PC. Once one problem has been solved, additional right hand sides can be processed in half a second.

3. A generalization of the technique to multiply connected domains. By combining the high order discretization with the FMM, an algorithm has been developed for obtaining the numerical solution to boundary integral equations defined on multiply connected domains whose individual bodies satisfy the axisymmetric constraint, but need not be oriented in the same direction or along the same axis.

The fast kernel evaluation can be extended to the kernels associated with scattering problems modeled by the Helmholtz equation. In this context, it will be advantageous to apply a fast direct solver such as [25] to solve (1.3).

Acknowledgments. The authors have greatly benefited from valuable suggestions made by Vladimir Rokhlin of Yale University.

REFERENCES

[1] M. Abramowitz and I.A. Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables.* Dover, New York, 1965.

[2] K. Atkinson. *The numerical solution of integral equations of the second kind.* Cambridge University Press, Cambridge, 1997.

[3] A.A. Bakr. *The boundary integral equation method in axisymmetric stress analysis problems.* Springer-Verlag, Berlin, 1985.

[4] M. Bebendorf. Approximation of boundary element matrices. *Numer. Math.*, 86:565–589, 2000.

[5] S. Börm. *Efficient Numerical Methods for Non-local Operators: H-Matrix Compression, Algorithms and Analysis.* European Mathematical Society, 2010.

[6] J. Bremer and V. Rokhlin. Efficient discretization of Laplace boundary integral equations on polygonal domains. *J. Comput. Phys.*, 229:2507–2525, 2010.

[7] J. Bremer, V. Rokhlin, and I. Sammis. Universal quadratures for boundary integral equations on two-dimensional domains with corners. Technical Report TR-1420, Yale University, Department of Computer Science, 2009.

[8] L. F. Canino, J. J. Ottusch, M. A. Stalzer, J. L. Visher, and S. M. Wandzura. Numerical solution of the Helmholtz equation in 2D and 3D using a high-order Nystrom discretization. *J. Comput. Phys.*, 146:627 – 663, 1998.

[9] H. Cheng, L. Greengard, and V. Rokhlin. A fast adaptive multipole algorithm in three dimensions. *J. Comput. Phys.*, 155:468–498, 1999.

[10] H.S. Cohl and J.E. Tohline. A compact cylindrical Green's function expansion for the solution of potential problems. *Astrophys. J.*, 527:86–101, 1999.

[11] J. T. Conway and H.S. Cohl. Exact Fourier expansion in cylindrical coordinates for the three-dimensional Helmholtz Green function. *Z. Angew Math. Phys.*, 61:425–443, 2010.

[12] J.L. Fleming, A.W. Wood, and W.D. Wood Jr. Locally corrected Nyström method for EM scattering by bodies of revolution. *J. Comput. Phys.*, 196:41–52, 2004.

[13] A. Gil, J. Segura, and N.M. Temme. *Numerical methods for special functions.* SIAM, Philadelphia, 2007.

[14] A. Greenbaum and L. Greengard. Laplace's equation and the Dirichlet-Neumann map in multiply connected domains. *J. Comput. Phys.*, 105:267–278, 1993.

[15] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.

[16] L. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. In *Acta numerica, 1997*, volume 6 of *Acta Numer.*, pages 229–269. Cambridge Univ. Press, Cambridge, 1997.

[17] R.B. Guenther and J.W. Lee. *Partial differential equations of mathematical physics and integral equations.* Dover, New York, 1988.

[18] A.K. Gupta. The boundary integral equation method for potential problems involving axisymmetric geometry and arbitrary boundary conditions. Master's thesis, University of Kentucky, 1979.

[19] W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method. *Numer. Math.*, 54:463–491, 1989.

[20] Wolfgang Hackbusch. A sparse matrix arithmetic based on H-matrices; Part I: Introduction to H-matrices. *Computing*, 62:89–108, 1999.

[21] J. Helsing and R. Ojala. Corner singularities for elliptic problems: Integral equations, graded meshes, quadrature, and compressed inverse preconditioning. *J. Comput. Phys.*, 227:8820–8840, 2008.

[22] J. Helsing and E. Wadbro. Laplace's equation and the Dirichlet-Neumann map: a new mode for Mikhlin's method. *J. Comput. Phys.*, 202:391–410, 2005.

[23] P. Kolm and V. Rokhlin. Numerical quadratures for singular and hypersingular integrals. *Comput. Math. Appl.*, 41:327–352, 2001.

[24] A.H. Kuijpers, G. Verbeek, and J.W. Verheij. An improved acoustic Fourier boundary element method formulation using fast Fourier transform integration. *J. Acoust. Soc. Am.*, 102:1394–1401, 1997.

[25] P.G. Martinsson and V. Rokhlin. A fast direct solver for boundary integral equations in two dimensions. *J. Comput. Phys.*, 205:1–23, 2004.

[26] G. Matviyenko. on the azimuthal Fourier components of the Greens function for the Helmholtz equation in three dimensions. *J. Math. Phys.*, 36:5159–5169, 1995.

[27] S.G. Mikhlin. *Integral Equations.* Pergammon, London, 1957.

[28] C. Provatidis. A boundary element method for axisymmetric potential problems with non-axisymmetric boundary conditions using fast Fourier transform. *Engrg. Comput.*, 15:428–449, 1998.

[29] F.J. Rizzo and D.J. Shippy. A boundary integral approach to potential and elasticity problems for axisymmetric bodies with arbitrary boundary conditions. *Mech. Res. Commun.*, 6:99–103, 1979.

[30] D.J. Shippy, F.J. Rizzo, and A.K. Gupta. Boundary-integral solution of potential problems involving axisymmetric bodies and nonsymmetric boundary conditions. In J.E. Stoneking, editor, *Developments in Theoretical and Applied Mechanics*, pages 189–206, 1980.

[31] B. Soenarko. A boundary element formuluation for radiation of acoustic waves from axisymmetric bodies with arbitrary boundary conditions. *J. Acoust. Soc. Am.*, 93:631–639, 1993.

[32] S.V. Tsinopoulos, J.P. Agnantiaris, and D. Polyzos. An advanced boundary element/fast Fourier transform axisymmetric formulation for acoustic radiation and wave scattering problems. *J. Acoust. Soc. Am.*, 105:1517–1526, 1999.

[33] W. Wang, N. Atalla, and J. Nicolas. A boundary integral approach for accoustic radiation of axisymmetric bodies with arbitrary boundary conditions valid for all wave numbers. *J. Acoust. Soc. Am.*, 101:1468–1478, 1997.

[34] L. Ying, G. Biros, and D. Zorin. A high-order 3D boundary integral equation solver for elliptic PDEs in smooth domains. *J. Comput. Phys.*, 219:247–275, 2005.

[35] P.M. Young and P.G. Martinsson. A direct solver for the rapid solution of boundary integral equations on axisymmetric surfaces in three dimensions. Technical report, University of Colorado at Boulder, Department of Applied Mathematics, 2010. http://arxiv.org/abs/1002.2001.