Chapter 3

A Fast, High-Order Method in Two Dimensions: Numerical Implementation

In this chapter, we present several significant improvements to the original numerical implementation of the two-dimensional method introduced in [13], as described in the previous chapter. The numerical solution of the associated approximate integral equation (2.3) consists of two main parts: efficient, high-order numerical quadrature rules and an efficient linear solver.

The numerical evaluation of the integrals in (2.6) requires both angular and radial integration. In Section 2.3, we described a method for computing the required angular integrals (2.9), *exactly* by means of Fourier smoothing of the scatterer. Furthermore, this method requires only $\mathcal{O}(M \log M)$ operations for each radial point. (For smooth scatterers, however, the angular integrals can instead be computed efficiently and with high-order accuracy by direct application of the trapezoidal rule without replacing m with m^{2M} (see Section 3.1.1).)

On the other hand, to compute the radial integrals $(K_\ell v^M)(a)$ (2.2), we present an improved scheme based on Chebyshev polynomials approximation. More precisely, we approximate $I_\ell(r)$ in (2.9) with Chebyshev polynomials on each of several subintervals (see Section 3.1.2). When the integration domain contains the origin, we resolve the logarithmic singularity in the Hankel function by integrating by parts (see Section 3.1.2.1). Then by means of appropriate pre-computed integral moments (see Section 3.1.2.2), we obtain a high-order approximation of the required integrals. Computation of these integrals requires only $\mathcal{O}(N)$ operations.

This high-order discretization of the required integrals in (2.6) results in a linear system with unknowns $v_\ell(a_j)$ for $\ell = -M, \ldots, M$ and $j = 0, \ldots, N_r$, where a_n are the radial points in the discretization. We use the Generalized Minimal Residual (GMRES) iterative method to solve this linear system (see Section 3.2.1). The number of GMRES iterations required to achieve a given residual tolerance increases rapidly with increasing problem size as measured by interior wavelengths (proportional to $\kappa n(R_1 - R_0)$ for a constant refractive index). We describe a preconditioner (see Section 3.2.2) that reduces this required number iterations for a variety of scattering configurations. We precondition the problem with an integral equation corresponding to a piecewise constant, radially layered approximating scatterer. By means of an equivalent partial differential equation, it can be shown that such an approximate integral equation admits an inverse, which can be computed in closed form. Application of this inverse requires only $\mathcal{O}(N)$ operations.

3.1 High-Order Numerical Integration

3.1.1 Angular Integration

We first consider the angular integration (2.9). As discussed in Section 2.3, $I_\ell(r)$ can be computed exactly even for discontinuous scatterers by replacing m by m^{2M} and choosing a sufficient number of trapezoidal rule integration points. Direct application of the trapezoidal rule in the case of a discontinuous inhomogeneity (without replacing m by m^{2M}) would yield only first-order convergence.

For smooth inhomogeneities m, however, we can obtain high-order convergence simply through direct application of the trapezoidal rule. This follows from the fact that, for smooth scatterers, the integrand $m(r,\theta)v^M(r,\theta)$ is a high-order approximation to the smooth and periodic function $m(r,\theta)u(r,\theta)$ and the trapezoidal rule yields high-order accuracy for smooth and periodic integrands. Hence, although $I_\ell(r)$ may always be computed exactly by replacing m by m^{2M}, for sufficiently smooth inhomogeneities, the direct application of the trapezoidal rule is somewhat simpler (since the Fourier coefficients $m_\ell(r)$ are not required) and produces nearly the same accuracy.

In this case, given the values $v_\ell(r)$, we first compute $v^M(r,\theta_j)$ for $\theta_j = 2\pi j/N_\theta$, where

N_θ is the number of trapezoidal rule integration points, i.e.,

$$v^M(r,\theta_j) = \sum_{\ell=-M}^{M} v_\ell(r,\theta_j) e^{2\pi i j\ell/N_\theta}$$

for $j = 0,\ldots,N_\theta - 1$. This sum is computed in $\mathcal{O}(N_\theta \log N_\theta)$ operations for each radial point r by means of an FFT, where N_θ must be chosen such that $N_\theta > 2M$. Using the periodicity in θ of m and v^M, the trapezoidal rule gives

$$I_\ell(r) \approx I_\ell^{N_\theta}(r) = \sum_{j=0}^{N_\theta-1} m(r,\theta_j) v^M(r,\theta_j) e^{-2\pi i j\ell/N_\theta}, \tag{3.1}$$

which we compute for $\ell = -M,\ldots,M$ by means of an FFT. Hence, given N_r radial points, this algorithm computes the integrals $I_\ell(r)$ with high-order accuracy in a total of $\mathcal{O}(N \log N)$ operations where $N = \mathcal{O}(N_r M)$.

It is interesting to note the relationship between this direct trapezoidal rule approach and the discrete convolution approach described in Section 2.3. When m is replaced by m^{2M} and N_θ is chosen such that $N_\theta > 4M$, these two approaches yield identical results. In other words, the trapezoidal rule approach described above is *identical* to the FFT-based method for computing a discrete convolution (when m^{2M} is used and $N_\theta > 4M$) [45, pp. 531–537]. Hence, these algorithms require only a single implementation; the approach that is actually used is determined by the choice of scatterer, either m or m^{2M}, and the related choice of N_θ.

3.1.2 Radial Integration

The radial integration requires the computation of two functions

$$\begin{aligned} F_\ell(a) &= \int_{R_0}^{a} Y_\ell(\kappa a) J_\ell(\kappa r) I_\ell(r) r dr \\ G_\ell(a) &= \int_{a}^{R_1} J_\ell(\kappa a) Y_\ell(\kappa r) I_\ell(r) r dr, \end{aligned}$$

for $R_0 \le a \le R_1$. In terms of these functions, $(K_\ell v^M)(a)$ is given by

$$(K_\ell v^M)(a) = -i\frac{\kappa^2}{4}\left[\int_{R_0}^{a} H_\ell^1(\kappa a)J_\ell(\kappa r)I_\ell(r)rdr + \int_{a}^{R_1} J_\ell(\kappa a)H_\ell^1(\kappa r)I_\ell(r)rdr\right] \quad (3.2)$$

$$= \frac{\kappa^2}{4}\left[F_\ell(a) + G_\ell(a) - i\frac{J_\ell(\kappa a)}{Y_\ell(\kappa R_1)}F_\ell(R_1)\right] \quad (3.3)$$

High-order accuracy in evaluating $F_\ell(a)$ and $G_\ell(a)$ requires that we treat discontinuities and/or singularities in $I_\ell(r)$ appropriately. The singularities in $I_\ell(r)$ are due to the singularities in the Fourier coefficients of the scatterer, $m_\ell(r)$. Thus, unless $m(r,\theta)$ is smooth within the annulus $R_0 \le r \le R_1$, $m_\ell(r)$ will exhibit singularities.

Following the discussion in [13], we assume that $m(r,\theta)$ is piecewise smooth, i.e., we assume that there are a finite number of disjoint sets $\mathcal{D}_i$ with piecewise smooth boundaries such that $m(r,\theta)$ is smooth on each $\mathcal{D}_i$. Singularities in $I_\ell(r)$ can occur at values of r that correspond either to non-smooth points in the boundary of a set $\mathcal{D}_i$ or to a point of tangency between the circle C_r of radius r and a set $\mathcal{D}_i$. Non-smooth points in the boundary of a set $\mathcal{D}_i$ lead to corner-type singularities. Points of tangency lead to singularities of type $(r-r_0)^\alpha$ for $0 \le \alpha \le 1$. For example, most of the examples of Section 5.1 have square-root type singularities, $\sqrt{r-r_0}$. Discontinuities in $I_\ell(r)$ may also arise, for example, when the circle C_r intersects a set $\mathcal{D}_i$ along a finite segment of its boundary.

Discontinuities and corner-type singularities are handled simply by subdividing the integration domain $[R_0, R_1]$ at the singularity points into two or more intervals. Singularities of the type $(r-r_0)^\alpha$ can be resolved by changing variables in the radial dimension. For example, as suggested in [13], we resolve square-root singularities occurring at the endpoints of the interval $[a,b]$ with the following change of variable

$$cos(\phi) = \sqrt{\frac{r^2-a^2}{b^2-a^2}}. \quad (3.4)$$

Of course, in order to resolve such singularities, one must first deduce the type of singularity. However, this is not difficult to accomplish in many situations.

Thus, by breaking the integration domain $[R_0, R_1]$ into multiple intervals and by appropriately changing variables to resolve remaining $(r-r_0)^\alpha$-type singularities, we are left with a smooth $I_\ell(r)$ on each interval of integration. Since the extension to this fully general setting is straightforward but also tedious, we describe only how the computation $F_\ell(a)$

and $G_\ell(a)$ to high-order accuracy for a *single* interval on which no change of variables is required. This special case illustrates all of the major aspects of the fully general method without unduly complicating the exposition.

We divide the integration interval $[R_0, R_1]$ into several subintervals of equal length. We then approximate $I_\ell(r)$ by a high-order polynomial expansion on each subinterval

$$I_\ell(r) \approx \sum_{n=0}^{N_c-1} c_{nk} p_n(r), \tag{3.5}$$

where $c_{nk} \in \mathbb{C}$, $k = 1, \ldots, N_i$ and N_i is the number of subintervals. Also, $p_n(r)$ is a polynomial of order n. Thus, we thereby obtain a polynomial expansion of order $N_c - 1$.

There are several possible choices for the $p_n(r)$. In [13], Lagrange interpolating polynomials are used. However, for Lagrange interpolating polynomials, the computation of the coefficients c_{nk} is numerically unstable for large N_c. This instability can be mitigated somewhat by using a more stable algorithm, at the cost of computational complexity. In [13], the chosen routine requires $\mathcal{O}(N_c^3)$ operations.

Hence, we use Chebyshev polynomials instead. Chebyshev polynomials, T_n, provide excellent approximation (nearly equal to the minimax polynomial [3, pp. 225–236]) while also allowing stable evaluation of the c_{nk} in $\mathcal{O}(N_c^2)$ operations. For r in the k^{th} subinterval, we have

$$I_\ell(r) \approx \sum_{n=0}^{N_c-1} c_{nk\ell} T_n(x_k(r)), \tag{3.6}$$

where the functions $x_k(r)$ are simply linear maps of the k^{th} subinterval to the interval $[-1, 1]$, which is the standard interval of definition for the Chebyshev polynomials. More precisely, if the k^{th} subinterval is given by $[a, b]$, then, for $r \in [a, b]$, we have

$$x_k(r) = \frac{r-a}{b-a} - \frac{b-r}{b-a}.$$

We will also make use of the inverse of this map, which, for $y \in [-1, 1]$, is given by

$$x_k^{-1}(y) = \frac{1}{2}\left[(b-a)y + (b+a)\right].$$

The evaluation of the Chebyshev coefficients for a function defined on the standard

interval $[-1,1]$ requires the function values at

$$y_j = \cos\left(\frac{\pi(j-1/2)}{N_c}\right), \tag{3.7}$$

for $j = 1,\ldots,N_c$. Hence, we require the value of $I_\ell(r)$ at the corresponding points in each of the subintervals, i.e., at the points

$$a_{jk} = x_k^{-1}(y_j) \tag{3.8}$$

for $j = 1,\ldots,N_c$ and $k = 1,\ldots,N_i$. Notice that the a_{jk} do not include the endpoints of each subinterval since the y_j do not include the endpoints of $[-1,1]$. Since we will need these endpoints for parts of our calculation, we include the left and right endpoints of each subinterval as a_{0k} and a_{N_pk}, respectively, where $N_p = N_c + 1$. (Note that by this definition $a_{0k} = a_{N_pk-1}$ for $k > 1$.)

Hence, these a_{jk} are our radial discretization points giving a total of $N_r = \mathcal{O}(N_pN_i)$ radial points. Our other discretization parameter is the number of modes M in the approximate solution v^M. Hence, the total number of unknowns $N = \mathcal{O}(N_rM) = \mathcal{O}(N_pN_iM)$. The number of points per subinterval N_p determines the *order* Chebyshev approximation is held *fixed* while the number of subintervals N_i and the number of modes M is increased to obtain more accuracy. Given the values of $I_\ell(a_{jk})$, one can compute the Chebyshev coefficients $c_{nk\ell}$ in $\mathcal{O}(N_p^2)$ operations per subinterval and per mode giving a total of $\mathcal{O}(N_iN_p^2M) = \mathcal{O}(N)$ operations for all subintervals. (One can reduce the N_p^2 complexity for each subinterval to $\mathcal{O}(N_p \log N_p)$ by use of FFTs. However, since N_p remains fixed, this does not change the overall complexity and since we typically use a relatively small value of N_p, e.g., $N_p = 9, 17$, we have found that this approach provides little benefit.)

Thus, given these Chebyshev approximations on each subinterval, we must only compute the following integral moments

$$P_{njk\ell} = \int_{a_{0k}}^{a_{jk}} Y_\ell(\kappa a_{jk}) J_\ell(\kappa r) T_n(x_k(r)) r dr \tag{3.9}$$

$$Q_{njk\ell} = \int_{a_{jk}}^{a_{N_pk}} J_\ell(\kappa a_{jk}) Y_\ell(\kappa r) T_n(x_k(r)) r dr. \tag{3.10}$$

Since all of the functions in the integrand are known analytically, these moments are only

computed once at the beginning of each run. The storage of these moments requires $\mathcal{O}(N_c N_p N_i M) = \mathcal{O}(N)$ memory. These integrals are problematic because of the rapid decay of J_ℓ for large ℓ near the origin as well as the logarithmic and polar singularities in Y_ℓ at the origin. In Section 3.1.2.1, we present a method for resolving the logarithmic singularity in Y_ℓ, which is necessary when the integration domain includes the origin, i.e., when $R_0 = 0$. We then discuss our method for computing the moments in Section 3.1.2.2.

Another practical obstacle when computing these moments concerns the rapid decay of J_ℓ and the rapid growth of Y_ℓ near the origin for large values of ℓ. Hence, for large values of ℓ, J_ℓ may underflow and Y_ℓ may overflow while their product remains machine-representable. We overcome this difficulty by computing scaled versions of J_ℓ and Y_ℓ, thus allowing accurate computation of $J_\ell(a)Y_\ell(b)$, $Y_\ell(a)/Y_\ell(b)$, etc. We describe our method for computing these scaled Bessel functions in Appendix B.

Once we have computed these moments, we can compute the values $F_{jk\ell} = F_\ell(a_{jk})$ and $G_{jk\ell} = G_\ell(a_{jk})$. We first compute the quantities $A_{jk\ell}$ and $B_{jk\ell}$, which involve integration over a single subinterval.

$$\begin{aligned} A_{jk\ell} &= \int_{a_{0k}}^{a_{jk}} Y_\ell(\kappa a_{jk}) J_\ell(\kappa r) I_\ell(r) r dr \\ &\approx \sum_{n=0}^{N_c-1} c_{nk\ell} P_{njk\ell} \\ B_{jk\ell} &= \int_{a_{jk}}^{a_{N_p k}} J_\ell(\kappa a_{jk}) Y_\ell(\kappa r) I_\ell(r) r dr \\ &\approx \sum_{n=0}^{N_c-1} c_{nk\ell} Q_{njk\ell}. \end{aligned}$$

Hence, given the Chebyshev coefficients, which require $\mathcal{O}(N)$ operations to compute, the computation of $A_{jk\ell}$ and $B_{jk\ell}$ requires $\mathcal{O}(N_c N_p N_i M) = \mathcal{O}(N)$ operations. We then sum and scale these values appropriately to compute $F_{jk\ell}$ and $G_{jk\ell}$.

$$\begin{aligned} F_{jk\ell} &= \int_{a_{01}}^{a_{jk}} Y_\ell(\kappa a_{jk}) J_\ell(\kappa r) I_\ell(r) r dr \\ &\approx A_{jk\ell} + \alpha_{jk\ell} F_{N_p k-1\ell}, \text{ for } k = 2, \ldots, N_i \\ F_{j1\ell} &\approx A_{j1\ell}, \end{aligned}$$

where $\alpha_{jk\ell} = \frac{Y_\ell(\kappa a_{jk})}{Y_\ell(\kappa a_{N_p k-1})}$.

$$\begin{aligned} G_{jk\ell} &= \int_{a_{jk}}^{a_{N_p N_i}} J_\ell(\kappa a_{jk}) Y_\ell(\kappa r) I_\ell(r) r dr \\ &\approx B_{jk\ell} + \beta_{jk\ell} G_{0j+1\ell}, \text{ for } k = 1, \ldots, N_i - 1 \\ G_{jN_i\ell} &\approx B_{jN_i\ell}, \end{aligned}$$

where $\beta_{jk\ell} = \frac{J_\ell(\kappa a_{jk})}{J_\ell(\kappa a_{0j+1})}$. Clearly, given $A_{jk\ell}$ and $B_{jk\ell}$, the computation of $F_{jk\ell}$ and $G_{jk\ell}$ requires $\mathcal{O}(N_p N_i M) = \mathcal{O}(N)$ operations.

Therefore, the total complexity of the radial integration, given the integral moments $P_{njk\ell}$ and $Q_{njk\ell}$, is $\mathcal{O}(N)$. Furthermore, assuming that the integral moments are computed with sufficient accuracy, the accuracy of the $F_{jk\ell}$ and $G_{jk\ell}$ is determined by the accuracy of the Chebyshev approximation on each subinterval. Since $I_\ell(r)$ is smooth (or can be made smooth by changing variables and breaking up the integration domain), the Chebyshev approximation is high-order accurate with the order of convergence dependent on the value of N_p, the number of points per subinterval. Thus, we obtain an efficient and high-order accurate method for computing the radial integrals. (See the numerical examples in Section 5.1.2.)

3.1.2.1 Resolution of Logarithmic Singularity

Besides the singularities in $I_\ell(r)$, which we resolve as discussed previously, the Bessel function $Y_\ell(\kappa r)$ exhibits logarithmic and polar singularities at $r = 0$. In this section, we show that the logarithmic singularity can be removed through integration by parts.

This method results as a slight modification of the approaches discussed above: we evaluate $F_{jk\ell}$ as discussed previously, but we change the computation of $G_{jk\ell}$. Note that $Y_\ell(z)$ can be written as

$$Y_\ell(z) = \frac{2}{\pi} \log\left(\frac{z}{2}\right) J_\ell(z) + Y_\ell^{polar}(z),$$

where $Y_\ell^{polar}(z) = \mathcal{O}(z^{-\ell})$ as $z \to 0$. Assume that $R_0 = 0$, since otherwise the logarithmic

singularity is not present. Integrating the logarithmic term by parts gives

$$\begin{aligned}\int_a^{R_1} \log\left(\frac{\kappa r}{2}\right) J_\ell(\kappa r) I_\ell(r) r dr &= \log\left(\frac{\kappa r}{2}\right) \int_0^r J_\ell(\kappa\rho) I_\ell(\rho)\rho d\rho \Big|_a^{R_1} \\ &\quad - \int_a^{R_1} dr \frac{1}{r} \int_0^r J_\ell(\kappa\rho) I_\ell(\rho)\rho d\rho \\ &= \log\left(\frac{\kappa R_1}{2}\right) S_\ell(R_1) - \log\left(\frac{\kappa a}{2}\right) S_\ell(a) - \int_a^{R_1} r^{-1} S_\ell(r) dr,\end{aligned}$$

where $S_\ell(a) = \int_0^a J_\ell(\kappa r) I_\ell(r) dr = \frac{F_\ell(a)}{Y_\ell(\kappa a)}$.

$S_\ell(r)$ vanishes to first order at $r = 0$ and thus, $r^{-1}S_\ell(r)$ is smooth near the origin. Furthermore, since we have computed $F_{jk\ell}$, i.e., the values of $F_\ell(a)$ at Chebyshev points in each subinterval, the values $S_\ell(a_{jk})$ as well as the Chebyshev coefficients of $r^{-1}S_\ell(r)$ on each subinterval are easily computed. Therefore, in $\mathcal{O}(N)$ operations, we compute the Chebyshev coefficients $d_{nk\ell}$ such that

$$r^{-1} S_\ell(r) \approx \sum_{n=0}^{N_c-1} d_{nk\ell} T_n(x_k(r)).$$

The Chebyshev coefficients $d_{nk\ell}$ of the integrand are easily mapped into the Chebyshev coefficients $D_{nk\ell}$ of its indefinite integral [45, pp. 189, 190] as

$$D_{nk\ell} = \frac{d_{n-1k\ell} - d_{n+1k\ell}}{2n}$$

for $n \geq 1$. The constant of integration $D_{0k\ell}$ is arbitrary. Clearly, this mapping requires only $\mathcal{O}(N)$ operations. Furthermore, since the integral $\int_a^{R_1} r^{-1} S_\ell(r) dr$ is smooth, the Chebyshev approximation on each subinterval with coefficients $D_{nk\ell}$ gives a high-order representation of this integral.

To complete the computation of $G_\ell(a_{jk})$, we define

$$G^{\log}_{jk\ell} = \int_{a_{jk}}^{R_1} J_\ell(\kappa a_{jk}) \frac{2}{\pi} \log\left(\frac{\kappa r}{2}\right) J_\ell(\kappa r) I_\ell(r) r dr$$

and

$$G^{polar}_{jk\ell} = \int_{a_{jk}}^{R_1} J_\ell(\kappa a_{jk}) Y^{polar}_\ell(\kappa r) I_\ell(r) r dr.$$

Therefore, $G_{jk\ell} = G^{\log}_{jk\ell} + G^{polar}_{jk\ell}$. We compute $G^{polar}_{jk\ell}$ exactly as we computed $G_{jk\ell}$ in the

previous section with $Q_{njk\ell}$ replaced by

$$Q^{polar}_{njk\ell} = \int_{a_{jk}}^{a_{N_p k}} J_\ell(\kappa a_{jk}) Y^{polar}_\ell(\kappa r) I_\ell(r) r dr. \tag{3.11}$$

We compute $G^{\log}_{jk\ell}$ by first defining, for $k = 1, \ldots, N_i - 1$,

$$\begin{aligned} R_{jk\ell} &= \int_{a_{jk}}^{R_1} r^{-1} S_\ell(r) dr \\ &= R_{0k+1\ell} + \int_{a_{jk}}^{a_{N_p k}} r^{-1} S_\ell(r) r dr \\ &\approx R_{0k+1\ell} + \sum_{n=0}^{N_c-1} D_{nk\ell} \left[T_n(x_k(a_{N_p k})) - T_n(x_k(a_{jk})) \right], \end{aligned}$$

and, for $k = N_i$, $R_{jN_i\ell} = \sum_{n=0}^{N_c-1} D_{nk\ell} \left[T_n(x_k(a_{N_p N_i})) - T_n(x_k(a_{jk})) \right]$. Given the coefficients $D_{nk\ell}$, computation of $R_{jk\ell}$ for $j = 0, \ldots, N_p$, $k = 1, \ldots, N_i$ and $\ell = -M, \ldots, M$ requires $\mathcal{O}(N_c N_p N_i M) = \mathcal{O}(N)$ operations to compute. Thus, we have

$$G^{\log}_{jk\ell} \approx \frac{2}{\pi} J_\ell(\kappa a_{jk}) \left[\log\left(\frac{\kappa R_1}{2}\right) S_{N_p N_i \ell} - \log\left(\frac{\kappa a_{jk}}{2}\right) S_{jk\ell} - R_{jk\ell} \right],$$

where $S_{jk\ell} = \frac{F_{jk\ell}}{Y_\ell(a_{jk})}$. Clearly, the computation of $G^{\log}_{jk\ell}$ requires only $\mathcal{O}(N)$ operations.

Therefore, given accurate values of the integral moments $P_{njk\ell}$ and $Q^{polar}_{njk\ell}$, this approach yields high-order accurate values of the required radial integrals in $\mathcal{O}(N)$ operations. We wish to emphasize that, as stated above, we need only resolve the logarithmic singularity in $Y_\ell(\kappa r)$ when the integration domain includes the origin, i.e., when $R_0 = 0$. It is straightforward to generalize this approach to the case in which we resolve singularities in $I_\ell(r)$ by breaking up the integration domain and/or changing variables.

3.1.2.2 Evaluation of Integral Moments

To complete the method, we need only find an efficient and accurate method for computing the integral moments $P_{njk\ell}$ and $Q_{njk\ell}$. As noted in the previous section, if the integration domain contains the origin, we can resolve the logarithmic singularity in Y_ℓ independently of the moment computation. In such a case, we compute $Q^{polar}_{njk\ell}$ (see equation (3.11)) instead of $Q_{njk\ell}$. However, even when the integration domain does not contain the origin, the rapid growth of the Y_ℓ and the rapid decay of the J_ℓ makes the development of accurate

quadrature schemes for computing the moments challenging. In particular, the Y_ℓ grow like $r^{-\ell}$ and the J_ℓ decay like r^ℓ as either $\ell \to \infty$ or as $r \to 0$. Since quadrature rules, for the most part, depend on accurate polynomial interpolation, accurate integration of these functions would seem to require a large number of integration points.

In this particular case, however, accurate values for these moments are actually not too difficult to obtain. The key insight is that a small *absolute* error, as opposed to a small *relative* error, is required in the computation of the moments. Since we desire a small relative accuracy in the *field* v^M, we require only a small absolute error in the Fourier coefficients of the field v_ℓ—small values of v_ℓ, regardless of their relative error, contribute very little to the value of the field v^M.

Our goal, therefore, is to show that, given a maximum absolute error $\varepsilon > 0$ and an initial radial discretization, we can choose the number of Gaussian quadrature points N_g such that the absolute error in the moment integrals is less that ε for all ℓ and for all subsequent refinements of the radial discretization. Near the origin or for large ℓ, the asymptotic behavior of J_ℓ and Y_ℓ for $\ell \geq 1$ are given by [2, p. 360]

$$J_\ell(z) \sim \frac{1}{\ell!}\left(\frac{z}{2}\right)^\ell$$

and

$$Y_\ell(z) \sim -\frac{(\ell-1)!}{\pi}\left(\frac{z}{2}\right)^{-\ell}.$$

Hence, for $\ell > 2$, the moment integrals,

$$P_\ell(a,b) = Y_\ell(\kappa b)\int_a^b J_\ell(\kappa r)T_n(x_k(r))r dr \tag{3.12}$$

and

$$Q_\ell(a,b) = J_\ell(\kappa a)\int_a^b Y_\ell(\kappa r)T_n(x_k(r))r dr \tag{3.13}$$

have roughly the same properties as

$$-\frac{1}{\pi\ell b^\ell}\int_a^b r^{\ell+1}dr = -\frac{a^2}{\pi\ell(\ell+2)}\left[\left(\frac{b}{a}\right)^2 - \left(\frac{a}{b}\right)^\ell\right] \tag{3.14}$$

and

$$\frac{a^\ell}{\pi\ell}\int_a^b r^{-\ell+1}dr = -\frac{b^2}{\pi\ell(\ell-2)}\left[\left(\frac{a}{b}\right)^2 - \left(\frac{a}{b}\right)^\ell\right], \tag{3.15}$$

respectively. Thus, by considering the relatively simple quantities (3.14) and (3.15), we gain insight into the behavior of integration rules for evaluating (3.12) and (3.13) . We will concentrate on integration rules for (3.13) since it is the more difficult of the two moment integrals because of the singularity in the integrand at $r = 0$. In addition, we restrict our attention to the case $a > 0$, since for $a = 0$, $Q_\ell(a, b) = 0$ for $\ell \geq 1$. (For $\ell = 0$, the only singularity in the integrand is the logarithmic singularity, which is resolved by integrating by parts as described in Section 3.1.2.1.)

Gauss-Legendre integration of a function f defined on the interval $[a, b]$ is given by [3, pp. 276–279]

$$\begin{aligned} \int_a^b f(t)dt &= \left(\frac{b-a}{2}\right)\int_{-1}^{1} f\left(\frac{b+a}{2}+\frac{b-a}{2}x\right)dx \\ &\approx \left(\frac{b-a}{2}\right)\sum_{j=1}^{N_g} w_j f\left(\frac{b+a}{2}+\frac{b-a}{2}x_j\right), \end{aligned} \tag{3.16}$$

where the points $-1 < x_j < 1$ are zeroes of the degree N_g Legendre polynomial on $[-1, 1]$ and $w_j > 0$ are the corresponding weights. We first consider the decay of the sum (3.16) itself for (3.15). We have

$$\frac{a^\ell}{\ell}\int_a^b r^{-\ell+1}dr \approx \frac{b-a}{2\ell}\sum_{j=1}^{N_g} w_j r_j \left(\frac{a}{r_j}\right)^\ell \leq \frac{b(b-a)}{\ell}, \tag{3.17}$$

where $r_j = \frac{(b+a)}{2} + \frac{(b-a)}{2}x_j$. Here we have also used the fact that $w_j > 0$ and that the $\sum_{j=1}^{N_g} w_j = \int_{-1}^{1} dx = 2$ since Gauss-Legendre quadrature integrates polynomials of degree less than $2N_g$ exactly. The sum (3.17) decays with increasing ℓ *independently of* N_g. Note that (3.15) *itself* also decays with ℓ as $\mathcal{O}(\ell^{-2})$. Thus, we conclude that, given an initial discretization, there is a positive integer L such that both (3.15) and (3.17) are smaller than $\varepsilon/2$ for all $\ell > L$, *independently of* N_g. Therefore, the absolute error for $\ell > L$ is smaller than ε. Crucially important in this result is the fact that the moments decay with ℓ. Thus, for sufficiently large ℓ, although the computed moments may have over 100% relative error, their absolute error remains smaller than ε.

We now consider the effect of discretization refinement. Refinement of the radial discretization is accomplished by increasing the number of subintervals N_i. (On each of these subintervals, there is a fixed number of Chebyshev points N_c.) Hence, for two adjacent

discretization points $a = R_0 + \alpha$ and $b = R_0 + \beta$, increasing N_i by some factor γ decreases both α and β by the factor γ. For example, doubling N_i halves both α and β. Similarly, $b-a$ also decreases by the factor γ when N_i increases by the factor γ. It follows from (3.17) that the absolute error for $\ell > L$ decays as we refine the radial discretization. Therefore, our choice of L is not only independent of N_g, but is also independent of subsequent refinements of the discretization.

Now we must consider the absolute error for $\ell \leq L$. The absolute error in the integral of a function f as computed with Gaussian quadrature is bounded by [3, pp. 276–279]

$$E_{N_g}(f) \leq \frac{\pi}{(2N_g)!}\left(\frac{b-a}{2}\right)\left(\frac{b-a}{4}\right)^{2N_g} \max_{a\leq t\leq b}\left|f^{(2N_g)}(t)\right|,$$

where the bound holds asymptotically as $N_g \to \infty$. One can then show that the absolute error $E^Q_{N_g}$ in computing (3.15) is bounded by

$$\begin{aligned} E^Q_{N_g} &= \frac{a^\ell}{\pi\ell} E_{N_g}(r^{-\ell+1}) \\ &\leq \frac{a(b-a)}{2}\left(\frac{b-a}{4a}\right)^{2N_g} \frac{(\ell-2+2N_g)!}{\ell(\ell-2)!(2N_g)!} \\ &\leq \frac{a(b-a)}{2}\left(\frac{b-a}{4a}\right)^{2N_g} \frac{(L-2+2N_g)!}{L(L-2)!(2N_g)!}, \end{aligned}$$

where we have used the fact that $\ell \leq L$. Hence, if we require that $(b-a)/4a < 1$, then $[(b-a)/4a]^{2N_g}$ exhibits exponential decay while $(L-2+2N_g)!/(2N_g)!$ exhibits polynomial growth as N_g increases. Therefore, for N_g sufficiently large $E^Q_{N_g} < \varepsilon$ for $\ell \leq L$. Clearly, $E^Q_{N_g}$ also decays with subsequent discretization refinements. Hence, our choice of N_g depends only on the value of L and the initial discretization.

We conclude that given an initial discretization satisfying $(b-a)/4a < 1$ for all points a, b, we can choose N_g sufficiently large such that the absolute error is smaller than ε for all ℓ and for all subsequent refinements of the discretization. Similar arguments obtain the same result for the absolute error in (3.14). Note that we have only proven this fact in the asymptotic regime of large ℓ or, equivalently, small a and b. However, outside of this asymptotic regime, the Bessel functions have much milder behavior, simply oscillating with wavenumber κ. These oscillations must be resolved by the radial discretization anyway to obtain even minimal accuracy in solving the integral equation. Hence, computing the inte-

gral moments in the oscillatory regime presents no significant difficulties. We do not present any theoretical estimates of the required values of N_g for various initial discretizations and error tolerances ε. However, the examples of Section 5.1 show that the value $N_g = 8$ suffices to achieve machine precision accuracy in our solutions.

3.2 Solution of the Linear System

3.2.1 Linear Solver

The previous sections describe our efficient, high-order method for computing the integral operator $K^M v^M$. The approximate solution v^M is then obtained by solving the linear system

$$v_\ell(a_{jk}) - (K_\ell v^M)(a_{jk}) = u^i_\ell(a_{jk}), \tag{3.18}$$

where $\ell = -M, \ldots, M$, $j = 0, \ldots, N_p$ and $k = 1, \ldots, N_i$. If we denote the matrix associated with the left-hand side by A, the unknown vector $v_\ell(a_{jk})$ by x and the right-hand side by b, equation (3.18) becomes simply $Ax = b$.

In general, A is dense. Therefore, because of the large amount of memory that would be required, we do not actually construct A in solving the linear system. Instead we use an iterative method that requires only the value of the matrix-vector product $y = Ax$ for any given vector x. Thus, y is computed in $\mathcal{O}(N \log N)$ operations by means of the integration schemes described in this chapter.

We use the Generalized Minimal Residual (GMRES) method. This iterative method builds a Krylov subspace associated with the matrix A and an initial guess x_0. The Krylov subspace at the k^{th} iteration is defined as

$$\mathcal{K}_k = span\{r_0, Ar_0, A^2 r_0, \ldots, A^k r_0\},$$

where $r_0 = b - Ax_0$. Typically, GMRES builds an orthonormal basis for this subspace using a modified Gram-Schmidt procedure. The approximate solution to the linear system at the k^{th} iteration, $x_k = x_0 + y_k$, is the vector that minimizes (in the least-squares sense) the residual $r = b - A(x_0 + y)$ for $y \in \mathcal{K}_k$ [29, pp. 38–41].

We chose GMRES because of its generality (it applies to non-Hermitian matrices) and because of its "optimal" convergence properties (optimal in the sense that it produces the

residual with the smallest 2-norm from the Krylov subspace [29, p. 49]). On the other hand, GMRES stores the basis vectors for the Krylov subspace and, hence, requires $\mathcal{O}(kN)$ memory, where k is the number of iterations and N is the number of unknowns. Furthermore, the construction of the orthonormal basis requires $\mathcal{O}(k^2N)$ operations. To avoid the large memory requirement, one can restart GMRES after a specified number of iterations, which has the side-effect of slowing the convergence.

In this context, it is useful to consider how the number of required iterations depends on the problem size. Unlike many classical methods for finite-difference equations, in this integral equation method, the number of iterations required to obtain a given residual tolerance *does not* depend on the mesh size. On the other hand, the number of iterations does depend on the size of the scatterer (as measured in interior wavelengths). In other words, for a given scatterer, as we increase the frequency of the incident wave (increase κ) and/or increase the value of m, the number of iterations required increases significantly (see Section 3.2.2).

Of course, other methods exist for non-Hermitian matrices that do not have the memory and complexity issues of GMRES, such as the Quasi-Minimal Residual (QMR), Biconjugate-Gradient (BiCG), stabilized Biconjugate-Gradient (BiCGSTAB) and Conjugate-Gradient Squared (CGS) methods [29, pp. 92–94]. In general, these methods sacrifice convergence rates for memory and complexity, but may be preferable in some cases. (See Section 4.5 for more discussion on these solvers.)

3.2.2 Preconditioner

In the previous sections, we described the numerical implementation of the method as well as the linear solver. However, when the scatterer is large ($\kappa R \gg 1$), or when the contrast is large ($|m| \gg 1$), the linear solver may converge so slowly that it becomes infeasible to obtain the desired residual value. We try to remedy this problem by means of a preconditioning matrix P for the linear system. The expectation is that pre- or post-multiplication of A (as well as the right-hand side) with an appropriate P will yield a new matrix with an eigenvalue spectrum that allows the linear solver to converge more rapidly. Of course, if P were the exact inverse of A, we would have $PA = AP = I$ and the linear solver would converge in a single iteration. Hence, roughly speaking, an effective preconditioner P will approximate the inverse of A as closely as possible while still allowing efficient pre- or post-multiplication.

Since our numerical method has a complexity of $\mathcal{O}(N \log N)$ where $N = N_p N_i M$, we want the multiplication by the preconditioner to be at least as efficient.

In this section, we introduce a preconditioner based on an approximate scatterer: we approximate the true inhomogeneity m by a piecewise constant, radially layered inhomogeneity $\widetilde{m}$. In a sense, this inhomogeneity forms a zeroth-order approximation to m in both geometry and value. As we will show, because of the relatively simple character of this scatterer, one can invert the associated integral equation in closed form. This inverse operator, which we use as our preconditioner P, can be computed in $\mathcal{O}(N)$ operations using radial integration methods identical to those described previously in Section 3.1.2.

As described above we define $\widetilde{m}$ as

$$\widetilde{m}(x) = \sum_{j=1}^{q} m_j \chi_{A_j}(x),$$

where m_j are constants. The sets $A_j, j = 1, 2, \ldots, q$ are the annular regions

$$A_j = \{x : a_{j-1} \leq |x| \leq a_j\},$$
$$0 = a_0 < a_1 < \cdots < a_q = R_1,$$

where χ_{A_j} is the characteristic function of the set A_j. Here we have assumed that $R_0 = 0$ to simplify the discussion somewhat; the case for $R_0 > 0$ proceeds similarly. The preconditioner P is given by the inverse of the associated integral equation

$$v(x) + \kappa^2 \sum_{j=1}^{q} m_j \int_{A_j} \Phi(\kappa|x-y|) v(y) dy = w(x), \tag{3.19}$$

where $|x| \leq R_1$ and $\Phi(z) = i/4 H_0^1(z)$. Given the inverse P, we then left-precondition our original equation as follows

$$P(I - K^M) v^M = P u^{i,M}.$$

This linear system is solved using an iterative solver, as described previously. Since the right-hand side of (3.19) is given by $w = (I - K^M) v^M$, $w = w^M$.

To solve (3.19), we derive an equivalent differential equation. Define a new unknown $u = v - w$, where v solves (3.19) and w is the given right-hand side. Although v is only a

solution for $|x| \leq R_1$, we can define u in all of $\mathbb{R}^2$ by

$$u = -\kappa^2 \sum_{j=1}^{q} m_j \int_{A_j} \Phi(\kappa|x-y|)v(y)dy.$$

For $x \in A_j$, we have

$$(\Delta + \kappa^2 n_j^2)u(x) = \kappa^2 m_j w, \quad j = 1, 2, \ldots, q$$

where $m_j = 1 - n_j^2$. For $|x| > R_1$ (and $|x| < R_0$ when $R_0 \neq 0$),

$$(\Delta + \kappa^2)u(x) = 0.$$

Furthermore, it is not difficult to show that u satisfies the Sommerfeld radiation condition as $|x| \to \infty$ [17, pp. 216–217]. This yields a differential equation in all of $\mathbb{R}^2$ that is equivalent to the preconditioning integral equation (3.19).

This equivalent differential equation can be solved in closed form as follows. For $x \in A_j$, we write the solution as a sum of a particular solution and a homogeneous solution, $u(x) = u_p(x) + u_h(x)$. Hence, for $x \in A_j$, a particular solution to the equation is

$$u_p(x) = -\kappa^2 m_j \int_{A_j} \Phi(\kappa n_j|x-y|)w(y)dy.$$

Since $w = w^M$, $u_p = u_p^M$ and in polar coordinates (a, ϕ)

$$(u_p)_\ell(a) = -\kappa^2 m_j \left[H_\ell^1(\kappa n_j a) \int_{a_{j-1}}^{a} J_\ell(\kappa n_j r) w_\ell(r) r dr + J_\ell(\kappa n_j a) \int_{a}^{a_j} H_\ell^1(\kappa n_j r) w_\ell(r) r dr \right], \tag{3.20}$$

for $a_{j-1} \leq a \leq a_j$. Clearly, these integrals can be computed with high-order accuracy and in $\mathcal{O}(N)$ operations using the *same methods* for radial integration that we discussed in Section 3.1.2. Furthermore, since the integration methods require the value of $w_\ell(r)$ at the Chebyshev points, which are exactly the values we have, application of these methods to the computation of $(u_p)_\ell$ requires no interpolation.

The homogeneous solution on the other hand is given by

$$u_h(a,\phi) = \begin{cases} \sum_{\ell=-\infty}^{\infty} \alpha_\ell^{(1)} J_\ell(\kappa n_j a) e^{i\ell\phi}, & \text{if } j = 1, \\ \sum_{\ell=-\infty}^{\infty} \left[\alpha_\ell^{(j)} J_\ell(\kappa n_j a) + \beta_\ell^{(j-1)} Y_\ell(\kappa n_j a)\right] e^{i\ell\phi}, & \text{if } j = 2, 3, \ldots, q, \end{cases} \tag{3.21}$$

for $a_{j-1} \le a \le a_j$. (If $R_0 \neq 0$, the homogeneous solution for $j = 1$ is given by a linear combination of J_ℓ and Y_ℓ instead of J_ℓ alone. Furthermore, if $R_0 > 0$, we must consider the homogeneous solution in the additional region $a < R_0$, which takes the same form as for $j = 1$ in (3.21).) Finally, for $a > R_1$, we have

$$u(a,\phi) = \sum_{\ell=-\infty}^{\infty} \beta_\ell^{(q)} H_\ell^1(\kappa a) e^{i\ell\phi}.$$

Clearly, given the correct values of the coefficients $\alpha_\ell^{(j)}$ and $\beta_\ell^{(j)}$, we can compute $(u_h)_\ell(a)$ for $\ell = -M, \ldots, M$ and all radial discretization points in $\mathcal{O}(N)$ operations. We thus obtain a closed-form solution to the integral equation (3.19).

To compute the $2q$ coefficients $\alpha_\ell^{(j)}$ and $\beta_\ell^{(j)}$, we require that $u \in C^1(\mathbb{R}^2)$ [28, pp. 53, 56]. Hence, by enforcing this condition at each $a_j, j = 1, 2, \ldots, q$, we obtain $2q$ equations, for $2q$ unknowns. For $j = 1$,

$$\begin{aligned} \alpha_\ell^{(1)} J_\ell(\kappa n_1 a_1) + (u_p)_\ell(a_1^-) &= \alpha_\ell^{(2)} J_\ell(\kappa n_2 a_1) + \beta_\ell^{(1)} Y_\ell(\kappa n_2 a_1) + (u_p)_\ell(a_1^+) \\ \alpha_\ell^{(1)} n_1 J_\ell'(\kappa n_1 a_1) + \frac{1}{\kappa}(u_p)_\ell'(a_1^-) &= \alpha_\ell^{(2)} n_2 J_\ell'(\kappa n_2 a_1) + \beta_\ell^{(1)} n_2 Y_\ell'(\kappa n_2 a_1) + \frac{1}{\kappa}(u_p)_\ell'(a_1^+) \end{aligned}$$

For $j = 2, 3, \ldots, q-1$, we have

$$\begin{aligned} \alpha_\ell^{(j)} J_\ell(\kappa n_j a_j) + \beta_\ell^{(j-1)} Y_\ell(\kappa n_j a_j) + (u_p)_\ell(a_j^-) &= \alpha_\ell^{(j+1)} J_\ell(\kappa n_{j+1} a_j) \\ &\quad + \beta_\ell^{(j)} Y_\ell(\kappa n_{j+1} a_j) + (u_p)_\ell(a_j^+) \\ \alpha_\ell^{(j)} n_j J_\ell'(\kappa n_j a_j) + \beta_\ell^{(j-1)} n_j Y_\ell'(\kappa n_j a_j) + \frac{1}{\kappa}(u_p)_\ell'(a_j^-) &= \alpha_\ell^{(j+1)} n_{j+1} J_\ell'(\kappa n_{j+1} a_j) \\ &\quad + \beta_\ell^{(j)} n_{j+1} Y_\ell'(\kappa n_{j+1} a_j) + \frac{1}{\kappa}(u_p)_\ell'(a_j^+) \end{aligned}$$

Finally, for $j = q$, we have

$$\begin{aligned} \alpha_\ell^{(q)} J_\ell(\kappa n_q R_1) + \beta_\ell^{(q-1)} Y_\ell(\kappa n_q R_1) + (u_p)_\ell(R_1^-) &= \beta_\ell^{(q)} H_\ell^1(\kappa R_1) + (u_p)_\ell(R_1^+) \\ \alpha_\ell^{(q)} n_q J_\ell'(\kappa n_q R_1) + \beta_\ell^{(q-1)} n_q Y_\ell'(\kappa n_q R_1) + \frac{1}{\kappa}(u_p)_\ell'(R_1^-) &= \beta_\ell^{(q)} (H_\ell^1)'(\kappa R_1) \end{aligned}$$

Here $(u_p)_\ell(a_j^+) = \lim_{a\to a_j^+}(u_p)_\ell(a)$ and $(u_p)_\ell(a_j^-) = \lim_{a\to a_j^-}(u_p)_\ell(a)$ with corresponding definitions for the derivatives.

For each ℓ, the matrix associated with this linear system is constant and banded with five diagonals. Hence, we compute the LU-decomposition of all of these matrices in $\mathcal{O}(qM)$ operations only once at the beginning of each run. In each iteration, after computing the $(u_p)_\ell(a_j^+)$ and $(u_p)_\ell(a_j^-)$ and their derivatives, we use the LU-decomposition to solve for the values of $\alpha_\ell^{(j)}$ and $\beta_\ell^{(j)}$ for all $j = 1, 2, \ldots, q$. This again requires a total of $\mathcal{O}(qM)$ operations.

Finally, we compute $(u_p)_\ell(a_j^+)$ and $(u_p)_\ell(a_j^-)$ and their derivatives. For $a_{j-1} < a < a_j$,

$$(u_p)_\ell(a) = -\frac{i\pi}{2}\kappa^2 m_j\left[H_\ell^1(\kappa n_j a)\int_{a_{j-1}}^{a} J_\ell(\kappa n_j r)w_\ell(r)rdr + J_\ell(\kappa n_j a)\int_a^{a_j} H_\ell^1(\kappa n_j r)w_\ell(r)rdr\right].$$

$$(u_p)_\ell(a_{j-1}^+) = -\frac{i\pi}{2}\kappa^2 m_j J_\ell(\kappa n_j a_{j-1})\int_{a_{j-1}}^{a_j} H_\ell^1(\kappa n_j r)w_\ell(r)rdr$$

and

$$(u_p)_\ell(a_j^-) = -\frac{i\pi}{2}\kappa^2 m_j H_\ell^1(\kappa n_j a_j)\int_{a_{j-1}}^{a_j} J_\ell(\kappa n_j r)w_\ell(r)rdr$$

Similarly,

$$(u_p)'_\ell(a_{j-1}^+) = -\frac{i\pi}{2}\kappa^3 n_j m_j J'_\ell(\kappa n_j a_{j-1})\int_{a_{j-1}}^{a_j} H_\ell^1(\kappa n_j r)w_\ell(r)rdr$$

and

$$(u_p)'_\ell(a_j^-) = -\frac{i\pi}{2}\kappa^3 n_j m_j (H_\ell^1)'(\kappa n_j a_j)\int_{a_{j-1}}^{a_j} J_\ell(\kappa n_j r)w_\ell(r)rdr$$

These integrals are easily obtained from the values of $(u_p)_\ell(a)$ in $\mathcal{O}(qM)$ operations.

Thus, application of this preconditioner requires only $\mathcal{O}(N)$ operations per iteration. We expect to observe the greatest benefits in using this preconditioner when the original inhomogeneity can be approximated reasonable well by a piecewise constant radially layered scatterer. Of course, the size of the annuli A_j as well as the approximating values m_j can be tuned to improve the convergence rate. We demonstrate the performance of the

preconditioner through the computational examples in Section 5.1.3.