Available at
www.ElsevierMathematics.com
POWERED BY SCIENCE DIRECT®

An International Journal
computers & mathematics
with applications

Computers and Mathematics with Applications 47 (2004) 1–11
www.elsevier.com/locate/camwa

A High-Order, Fast Algorithm for Scattering Calculation in Two Dimensions

J. C. AGUILAR
Departamento de Matemáticas
Instituto Tecnológico Autónomo de México
México, D.F. 01000
aguilar@itam.mx

YU CHEN
Courant Institute of Mathematical Sciences
New York University, New York, NY 10012, U.S.A.
yuchen@cims.nyu.edu

(Received April 2001; revised and accepted August 2003)

Abstract—We present a high-order, fast, iterative solver for the direct scattering calculation for the Helmholtz equation in two dimensions. Our algorithm solves the scattering problem formulated as the Lippmann-Schwinger integral equation for compactly supported, smoothly vanishing scatterers. There are two main components to this algorithm. First, the integral equation is discretized with quadratures based on high-order corrected trapezoidal rules for the logarithmic singularity present in the kernel of the integral equation. Second, on the uniform mesh required for the trapezoidal rule we rewrite the discretized integral operator as a composition of two linear operators: a discrete convolution followed by a diagonal multiplication; therefore, the application of these operators to an arbitrary vector, required by an iterative method for the solution of the discretized linear system, will cost $N^2 \log(N)$ for a N-by-N mesh, with the help of FFT. We will demonstrate the performance of the algorithm for scatterers of complex structures and at large wave numbers. For numerical implementations, GMRES iterations will be used, and corrected trapezoidal rules up to order 20 will be tested.

Keywords—Integral equation, Large wave numbers, Quadrature rules, Logarithmic singularity, Correction coefficients.

1. INTRODUCTION

As is well known, the scattering of acoustic waves by inhomogeneous media can be modeled by the Helmholtz equation, and reformulated as Lippmann-Schwinger integral equation. The subject of this paper is on the numerical solution of the integral equation in two dimensions.

For many applications of acoustic waves in underwater acoustics, ultrasound imaging, and in the related inverse scattering problems, accurate and rapid numerical solutions of the scattering problem are required for large wave numbers. Two issues of numerical difficulty immediately follow these large problems. First, for a scatterer on the order of 100 wavelengths in diameter,

0898-1221/04/$ - see front matter
doi: 10.1016/S0898-1221(03)00446-2

loss of precision due to phase pollution [1] becomes serious, and thus, the use of high-order methods is essential. Second, even with a suitable, high-order scheme, the number of points per wavelength for discretization of each linear dimension is on the order of ten for a problem with any considerable presence of multiple scattering. Thus, for a two-dimensional problem of 100-by-100 wavelengths, the dense linear system for Lippmann-Schwinger equation is million-by-million in size.

Most existing fast solvers for the scattering problem are iterative [2–5], with an optimal number of operations $O(N^3 \log(N))$ for an $N \times N$ computational mesh. In three dimensions, these methods will require at least $O(N^4 \log(N))$. More recently, fast direct solvers [6–9] were developed for the Lippmann-Schwinger equation but found inefficient for use in three dimensions. There are a number of reliable schemes for the solution of the integral equation at low wave number, based on either compression of the kernel [10] or fast multipole approach [11,12]. But they are not well suited for high wave number calculations.

We present a high-order, fast, iterative solver for the scattering calculation in two dimensions that can readily be extended to three dimensions. Our approach is similar to that of [5]: FFT is used to evaluate the discrete convolution that comes from discretization of the integral operator. Our method differs from the existing ones in that we discretize the integral equation with a Nyström method, and directly tackle the logarithmic singularity of the Green's function with high-order, corrected trapezoidal quadratures.

We will examine the performance of the fast solver with several scatterers of complex structures, at various wave numbers, and for quadrature rules of order 2, 4, 14, 20.

REMARK 1.1. We will only test smoothly vanishing scatterers for which no special treatment is required for the correction of the trapezoidal quadratures on the boundary of the scatterer. The issue of correction on the boundary where the scatterer vanishes abruptly is extremely interesting but not satisfactorily resolved.

REMARK 1.2. Extension of the method to three dimensions is possible. In fact, we have constructed several high-order, corrected trapezoidal quadratures for the Lippmann-Schwinger equation in three dimensions. These results will be reported later.

The paper is organized as follows. Section 2 introduces the scattering problem and its integral formulation in 2-D. In Sections 3 and 4, we construct a high-order solver for the scattering problem. In Section 5, we study the behavior of the numerical solution of the scattering problem in 1-D for large wave numbers; we will demonstrate that large wave numbers introduce additional difficulties for lower-order solvers. In Section 6, we present numerical experiments for the 2-D solver constructed in Section 4; we will use relative large wave numbers in our numerical tests.

2. THE LIPPMANN-SCHWINGER EQUATIONS

We consider the scattering problem governed by the Helmholtz equation

$$\Delta\phi(\boldsymbol{x}) + k^2(1 + q(\boldsymbol{x}))\phi(\boldsymbol{x}) = 0. \tag{1}$$

In (1), we assume that the wave number k is a complex number such that $\Im(k) \geq 0$, and $q : \mathbb{R}^2 \to \mathbb{R}$ a smooth function, with $q(\boldsymbol{x}) > -1$ for all $\boldsymbol{x} \in \mathbb{R}^2$. We will be referring to the function q as the scatterer. We further assume that q is compactly supported in a domain D. We will be considering solutions of (1) in the form

$$\phi(\boldsymbol{x}) = \phi_0(\boldsymbol{x}) + \psi(\boldsymbol{x}), \tag{2}$$

where $\phi_0 : D \to \mathbb{C}$ is a solution of the equation

$$\Delta\phi_0(\boldsymbol{x}) + k^2\phi_0(\boldsymbol{x}) = 0 \tag{3}$$

and $\psi : \mathbb{R}^2 \to \mathbb{C}$ is, consequently, the solution of the equation

$$\Delta\psi(\boldsymbol{x}) + k^2(1 + q(\boldsymbol{x}))\psi(\boldsymbol{x}) = -k^2 q(\boldsymbol{x})\phi_0(\boldsymbol{x}), \tag{4}$$

subject to the outgoing (Sommerfeld) radiation condition

$$\lim_{r\to\infty} \sqrt{r}\left(\frac{\partial\psi}{\partial r} - ik\psi\right) = 0, \tag{5}$$

We will be referring to ϕ as the total wave, to ϕ_0 as the incident wave, and to ψ as the scattered wave. Furthermore, we will be referring to the determination of the scattered wave from a given incident wave as the (forward) scattering problem. Suppose that $\boldsymbol{x}, \boldsymbol{\xi}$ are two points in $\mathbb{R}^2$. Then the free space Green's function, subject to the radiation condition, is

$$G(\boldsymbol{x}, \boldsymbol{\xi}) = -\frac{i}{4} H_0(k\|\boldsymbol{x} - \boldsymbol{\xi}\|), \tag{6}$$

where H_0 is the zero-order Hankel function of the first kind. As is well known, the scattering problem (4),(5) is well posed and can be reformulated as the Lippmann-Schwinger integral equation

$$\psi(\boldsymbol{x}) + k^2 \int_D G(\boldsymbol{x}, \boldsymbol{\xi}) q(\boldsymbol{\xi}) \psi(\boldsymbol{\xi})\, d\boldsymbol{\xi} = -k^2 \int_D G(\boldsymbol{x}, \boldsymbol{\xi}) q(\boldsymbol{\xi}) \phi_0(\boldsymbol{\xi})\, d\boldsymbol{\xi}, \tag{7}$$

for ψ. This equation has a conjugate form

$$\sigma(\boldsymbol{x}) + k^2 q(\boldsymbol{x}) \int_D G(\boldsymbol{x}, \boldsymbol{\xi}) \sigma(\boldsymbol{\xi})\, d\boldsymbol{\xi} = -k^2 q(\boldsymbol{x}) \phi_0(\boldsymbol{x}), \tag{8}$$

for the charge density $\sigma : D \to \mathbb{C}$ defined by

$$\psi(\boldsymbol{x}) = \int_D G(\boldsymbol{x}, \boldsymbol{\xi}) \sigma(\boldsymbol{\xi})\, d\boldsymbol{\xi}. \tag{9}$$

3. HIGH-ORDER, CORRECTED TRAPEZOIDAL RULES

The performance of an iterative method solving the integral equations (7) and (8) hinges on the accuracy and efficiency in the evaluation of the integrals

$$u(\boldsymbol{x}) = \int_D G(\boldsymbol{x}, \boldsymbol{\xi}) v(\boldsymbol{\xi})\, d\boldsymbol{\xi}, \tag{10}$$

for $\boldsymbol{x} \in D$ and for any smooth function v that vanishes smoothly outside D. We present in this section a method for evaluating the integral

$$J(v)(\boldsymbol{x}) = \int_D \ln(\boldsymbol{x} - \boldsymbol{\xi}) v(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{11}$$

with high-order quadratures. These quadratures will be adapted in Section 4 to obtain high-order discretizations of (10) and of the integral equation (8).

Let $r = \|\boldsymbol{x} - \boldsymbol{\xi}\|$ where $\boldsymbol{x}, \boldsymbol{\xi}$ are two points in $\mathbb{R}^2$. As is well known [13],

$$g(r) := G(\boldsymbol{x}, \boldsymbol{\xi}) = -\frac{i}{4} H_0(kr) = g_1(r)\ln(r) + g_2(r), \tag{12}$$

where

$$g_1(r) = \frac{1}{2\pi} + \omega_1(r), \tag{13}$$

$$g_2(r) = \frac{1}{2\pi}\left[\ln\left(\frac{k}{2}\right) + \gamma\right] - \frac{i}{4} + \omega_2(r), \tag{14}$$

$$\gamma = 0.5772156649015328606\ldots \text{ is Euler's constant}, \tag{15}$$

and $\omega_1, \omega_2 \sim r^2$ being smooth functions of r.

3.1. Corrected Trapezoidal Rules for Logarithmic Singularity

Given $h > 0$, we define the trapezoidal quadrature for the entire plane $\mathbb{R}^2$ by the formula

$$T_h(f) = h^2 \sum_{i,j\in\mathbb{Z}} f(ih, jh). \tag{16}$$

Obviously, if f compactly supported in a domain D, $T_h(f)$ involves a finite sum over the lattice $\mathbb{Z}$. Denoting by

$$T_{0,h}(f) = h^2 \sum_{i^2+j^2\neq 0} f(ih, jh) \tag{17}$$

the punctured trapezoidal quadrature, we now introduce the corrected trapezoidal rules for the singular integral (11) with a smoothly vanishing v.

A fourth-order, corrected trapezoidal rule for (11) is of the form

$$J(v) = T_{0,h}(v \ln(r)) + h^2 (\ln(h) + c_1) v(0,0) + O\left(h^4\right). \tag{18}$$

That is, $J(v)$ is obtained to fourth order with a punctured trapezoidal rule plus a single correction term involving the constant

$$c_1 = -1.3105329259115095\ldots. \tag{19}$$

Similarly, a sixth-order correction can be written as

$$\begin{aligned} J(v) &= T_{0,h}(v \ln(r)) + h^2 (\ln(h) + c_1) v(0,0) \\ &+h^2 c_2 (v(h,0) + v(-h,0) + v(0,h) + v(0,-h)) + O\left(h^6\right), \end{aligned} \tag{20}$$

with correction coefficients

$$c_1 = -1.2133459579012365\ldots, \qquad c_2 = -0.024296742002568231\ldots. \tag{21}$$

See [14] for details in the determination of these correction coefficients.

3.2. Higher-Order Corrected Trapezoidal Rules

Higher-order rules can also be constructed; see Table 1 for the correction coefficients for orders up to 20, and see [14] for details of their accurate calculation.

To use these correction coefficients for a high-order rule, we require the following notation, see [14] for details. Let $c_1, c_2, \ldots, c_M$ be the coefficients shown in Table 1, where for example $M = 37$ for the 20^{th}-order rule. For each integer $j \in [1, M]$, let l_j, m_j be the integers to satisfy the two conditions

$$0 \le m_j \le l_j, \qquad j = 1 + m_j + \frac{l_j(l_j+1)}{2}. \tag{22}$$

Let $\boldsymbol{P}_j = (l_j, m_j)$; the pair of equations uniquely determines $\boldsymbol{P}_1 = (0,0)$, $\boldsymbol{P}_2 = (1,0)$, $\boldsymbol{P}_3 = (1,1)$, $\ldots$, $\boldsymbol{P}_{37} = (8,0)$, see Figure 1.

For each j let

$$d_j = \sqrt{l_j^2 + m_j^2} = \|\boldsymbol{P}_j\| \tag{23}$$

and

$$W_j = \{(\pm l_j, m_j), (\pm m_j, l_j), (\pm l_j, -m_j), (\pm m_j, -l_j)\}. \tag{24}$$

for example, $W_5 = \{\boldsymbol{P}_5, (2,-1), (1,2), (-1,2), (-2,1), (-2,-1), (-1,-2), (1,-2)\}$. If we define the M coefficients $\tilde{c}_1 = c_1 + \ln(h)$ and $\tilde{c}_j = c_j$ for $j \in [2, M]$, then according to [14] the corrected trapezoidal rule to approximate (11) can be written as

Table 1. Correction coefficients for c for a logarithmic singularity, and for $M = 1$, 2, 4, 6, 7, 11, 16, and 37.

$M = 1$, Fourth Order	$M = 16$, 14^{th} Order	$M = 37$, 20^{th} Order
−1.3105329259115095 d0	−1.1646982357508747 d0	−1.1564478673399723 d0
	−3.5890328129867669 d − 2	−3.8126710449913075 d − 2
	−9.5074099436320872 d − 3	−1.1910098717735232 d − 2
$M = 2$, Sixth Order	8.4541772191636749 d − 3	1.0946813560280918 d − 2
	1.0979359740499282 d − 3	1.8459370374209805 d − 3
−1.2133459579012365 d0	−1.1783003516981361 d − 5	−2.6130060578859742 d − 5
−2.4296742002568231 d − 2	−1.6023206924446483 d − 3	−2.9040233795126303 d − 3
	−1.6849437585541639 d − 4	−4.9701596518365230 d − 4
	3.3320425168508138 d − 6	1.2749931410650803 d − 5
$M = 4$, Eighth Order	−9.8490563660380440 d − 7	−6.5820750987412075 d − 6
	2.2604824606510965 d − 4	7.3301436931546191 d − 4
−1.1882171416684368 d0	1.2470171982677393 d − 5	1.0733577004383499 d − 4
−3.0413000735379221 d − 2	−1.7168213185329377 d − 7	−1.9458112583293086 d − 6
−3.3900200171833950 d − 3	6.6801225895094825 d − 8	1.3143856331004638 d − 6
3.2240746917944449 d − 3	−4.3347365473805450 d − 9	−2.5402660668233166 d − 7
	−1.6344859129100059 d − 5	−1.5475173218203846 d − 4
$M = 7$, Tenth Order		−1.7683396386845410 d − 5
		2.6480195544534095 d − 7
−1.1765131626655374 d0		−2.1443542291607470 d − 7
−3.3070930145520950 d − 2		4.1096136734188740 d − 8
−6.1598611771676465 d − 3		−6.6696462836304480 d − 9
5.5343086429652787 d − 3		2.4751027705921126 d − 5
3.4587810881957096 d − 4		1.9206904724678774 d − 6
−1.7601808923023545 d − 7		−2.4935555816866533 d − 8
−5.0039036749807269 d − 4		2.3124001225072397 d − 8
		−4.3978037096882189 d − 9
$M = 11$, 12^{th} Order		7.1457202698992097 d − 10
		−7.6547561188576653 d − 11
−1.1694962171857752 d0		−2.6158207181242810 d − 6
−3.4698254694377585 d − 2		−1.0183222363887877 d − 7
−8.1243444153848045 d − 3		1.1918930487773071 d − 9
7.1885293443181541 d − 3		−1.2213737542214254 d − 9
7.4595382605746944 d − 4		2.3104783998729453 d − 10
−5.5672375863432573 d − 6		−3.7578874163491022 d − 11
−1.0668259664240182 d − 3		4.0255398498327133 d − 12
−6.6934093317098417 d − 5		−2.1171431666402956 d − 13
1.0591321235750506 d − 6		1.3551691363041958 d − 7
−1.9350916131464208 d − 7		
8.7321567454452694 d − 5		

$$
\begin{aligned}
J(v) &= T_{0,h}(v \ln(r)) + h^2 \ln(h)\, v(0,0) + h^2 \left(\sum_{j=1}^{M} c_j \sum_{(l,m)\in W_j} v(lh, mh) \right) + O\left(h^{2l_M+4}\right) \\
&= T_{0,h}(v \ln(r)) + h^2 \left(\sum_{j=1}^{M} \tilde{c}_j \sum_{(l,m)\in W_j} v(lh, mh) \right) + O\left(h^{2l_M+4}\right).
\end{aligned}
\tag{25}
$$

Since $\boldsymbol{P}_1 = (l_1, m_1) = (0,0)$ and $\boldsymbol{P}_{37} = (l_{37}, m_{37}) = (8,0)$, the order of convergence $2l_M + 4$ in (25)

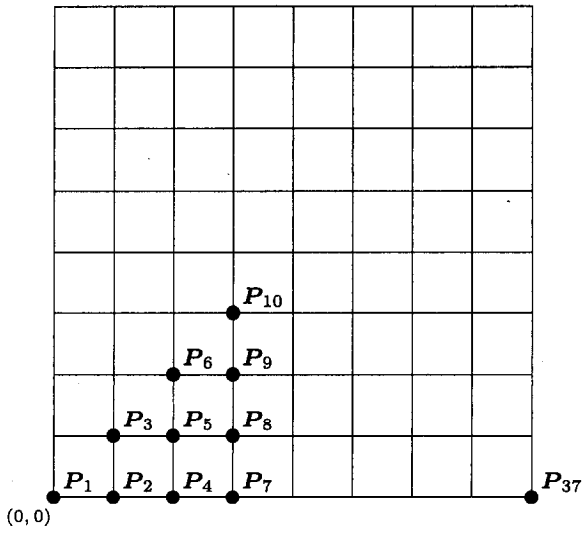


Figure 1. For each positive integer j there corresponds a unique point $\boldsymbol{P}_j = (l_j, m_j) \in \mathbb{Z} \times \mathbb{Z}$ such that $0 \leq m_j \leq l_j$ and $j = 1 + m_j + l_j(l_j + 1)/2$. In the grid, it is shown $\boldsymbol{P}_1 = (0,0)$, $\boldsymbol{P}_2 = (1,0)$, $\boldsymbol{P}_3 = (1,1)$, etc.

is 4 if $M = 1$ (in this case, (25) becomes the diagonal correction (18)) and is 20 if $M = 37$. Notice that when $M = 0$ in (25) there is no correction added to the punctured trapezoidal quadrature, and in this case, the order of convergence in (25) is 2.

4. HIGH-ORDER DISCRETIZATION OF THE INTEGRAL EQUATION

The corrected trapezoidal rules discussed in the preceding section can be used to discretize the integral equation (8). Let us first consider the case of the fourth-order formula (18). It follows from (18) and (12) that (10) can be approximated by the fourth-order rule

$$\begin{aligned} u &= I\left(v \cdot g_1(r) \cdot \ln(r)\right) + I\left(v \cdot g_2(r)\right) \\ &= T_{0,h}\left(v \cdot g_1(r) \cdot \ln(r)\right) + T_h\left(v \cdot g_2(r)\right) + h^2\left(\ln(h) + c_1\right) v(0,0)\, g_1(0) + O\left(h^4\right) \\ &= T_{0,h}\left(v \cdot g(r)\right) + \frac{h^2\left(\ln(h) + c_1\right) v(0,0)}{2\pi} + h^2 v(0,0)\left[\frac{\ln(k/2) + \gamma}{2\pi} - \frac{i}{4}\right] + O\left(h^4\right) \\ &= T_{0,h}\left(v \cdot g(r)\right) + h^2 \beta_1 v(0,0) + O\left(h^4\right), \end{aligned} \tag{26}$$

with

$$\beta_1 = -\frac{i}{4} + \frac{\ln(hk/2) + \gamma + c_1}{2\pi}. \tag{27}$$

In general, for a rule of order $2l_M + 4$, (25) applied to (10) gives us

$$u = T_{0,h}\left(v \cdot g(r)\right) + h^2 \left(\sum_{j=1}^{M} \beta_j \sum_{(l,m) \in W_j} v(lh, mh)\right) + O\left(h^{2l_M + 4}\right), \tag{28}$$

with

$$\beta_1 = g_2(0) + \left(\ln(h) + c_1\right) g_1(0) = -\frac{i}{4} + \frac{1}{2\pi}\left(\ln\left(\frac{hk}{2}\right) + \gamma + c_1\right), \tag{29}$$

$$\beta_j = g_1(d_j h)\, c_j, \qquad \text{for } j = 2, \ldots, M. \tag{30}$$

To discretize the integral equation (8) using (28), we assume that $D = [0, 2\pi] \times [0, 2\pi]$, which is discretized with an uniform mesh (x_m, y_n), $m, n = 0, \ldots, N$, where $y_j = x_j = jh$ with $h = 2\pi/N$. The function v is sampled as

$$v_{m,n} = v(x_m, y_n), \qquad m, n \in [0, N-1]. \tag{31}$$

To obtain an array $\boldsymbol{v} \in \mathbb{C}^{N\times N}$. Similarly, $G(\boldsymbol{x},\boldsymbol{\xi})$ is evaluated on the mesh, except when $\boldsymbol{x}=\boldsymbol{\xi}$, and the correction coefficients are added to appropriate sampled values of G: if the double sequence $\{g_{m,n}\}_{m,n\in\mathbb{Z}}$ is given by

$$g_{m,n} = \begin{cases} \beta_1, & \text{if } (m,n)=(0,0)\,, \\ g\left(d_j h\right)+\beta_j, & \text{if } (m,n)\in W_j, \text{ for some } j\in[2,M]\,, \\ g\left(d_j h\right), & \text{if } (m,n)\in W_j, \text{ for some } j>M, \end{cases} \tag{32}$$

with W_j and d_j as defined in (24) and g as in (12), then it follows immediately from (28) that the value of u in integral (10) on the mesh (x_m, y_n), satisfies

$$u\left(x_m,y_n\right) = h^2 \sum_{r,s=0}^{N-1} g_{m-r,n-s}\cdot v_{r,s} + O\left(h^{2l_M+4}\right). \tag{33}$$

The convolution in (33) can be evaluated efficiently with FFT by periodizing $\{g_{m,n}\}$ and zero-padding $\boldsymbol{v}$ (see [15], for example).

By restricting $\boldsymbol{x}$ to the mesh in D, it follows that the conjugate Lippmann-Schwinger equation (8) can be discretized as

$$\sigma_{m,n} + k^2h^2 q_{m,n}\cdot \sum_{r,s=0}^{N-1} g_{m-r,n-s}\sigma_{r,s} = -k^2 q_{m,n}\cdot\phi_{m,n} + O\left(h^{2l_M+4}\right). \tag{34}$$

Equivalently,

$$\boldsymbol{\sigma}^h + \boldsymbol{q}^h\cdot\left(\boldsymbol{g}^h * \boldsymbol{\sigma}^h\right) = -k^2\boldsymbol{q}^h\cdot\boldsymbol{\phi}^h, \tag{35}$$

where $*$ denotes discrete convolution and $\boldsymbol{g}^h = k^2h^2\{g_{m,n}\}$, defined in (32), is a scaled double sequence of corrected sampled values of the Green's function G. The entries (m,n) of the arrays $\boldsymbol{q}^h$ and $\boldsymbol{\phi}^h$ are the values of the scatterer function q and of the incident wave ϕ_0, respectively, at the grid point (x_m, y_n). The unknown array $\boldsymbol{\sigma}^h$ approximates the charge density σ at the grid points of the square D, $\boldsymbol{\sigma}^h(m,n) = \sigma(x_m,y_n) + O(h^{2l_M+4})$. The array $\boldsymbol{q}^h\cdot\boldsymbol{\phi}^h$ is defined as $\boldsymbol{q}^h\cdot\boldsymbol{\phi}^h(m,m) = \boldsymbol{q}^h(m,n)\boldsymbol{\phi}^h(m,n)$.

The system of equations (35) can be written in the form

$$\boldsymbol{A}_h\boldsymbol{\sigma}^h = \mathbf{b}^h. \tag{36}$$

A matrix-vector multiplication $\boldsymbol{A}_h\boldsymbol{w}$ can be evaluated in $O(P\log(P))$ operations using FFT [15], where $P=N^2$ is the number of unknowns of system (36).

5. HIGH-ORDER METHODS FOR LARGE WAVE NUMBER

One may wonder why we need to design and employ methods with order as high as 20 or 40, and whether these high-order methods will at all make a difference. In this section, we will demonstrate the necessity of high-order methods for solution of the Lippmann-Schwinger equation with large wave number. For many applications, such as medical imaging, radar cross section calculation, the size of the scattering problem measured by the number of wavelengths in each linear dimension could easy be several hundred.

To investigate the error of the numerical solution of the Lippmann-Schwinger equation for large wave numbers, we consider a sample, 1-D scattering problem whose Lippmann-Schwinger equation is

$$\sigma\left(x\right) + k^2 q\left(x\right)\int_0^{2\pi} G\left(x,\xi\right)\sigma\left(\xi\right)\,d\xi = -k^2 q\left(x\right)\phi_0\left(x\right), \tag{37}$$

where G is defined by

$$G(x,\xi) = \frac{e^{ik|x-\xi|}}{2ik}. \tag{38}$$

Equation (37) can be discretized analogously with what is described in Section 4; high-order discretizations can be devised from the quadratures described in [16,17]. Our numerical calculations demonstrate that for large values of the wave number k, and for discretizations of (37) that converge at the rate h^p, the relative error of convergence in the L_2 norm is approximately $C_p k^{p+1} h^p$, with a proportionality coefficient C_p which depends on the scatterer q and on the order p of convergence. This is consistent with the analysis of [1]. Assuming that $I = [0, 2\pi]$, the number of sampling points in I to discretize (37) can be written as kP_w, where k is the wave number and P_w is the number of points per wavelength. Table 2 shows the estimated number of points per wavelength P_w required to obtain a relative error of 10^{-4} for various orders of convergence and for high wave numbers. The results in Table 2 were obtained using a linear combination of 11 Gaussian bumps with fast decay outside $[0, 2\pi]$ as the scatterer function, for which the values of the proportionality coefficient C_p are given below.

Table 2. An estimated number of points per wavelength P_w needed to achieve a relative error (in the L_2 norm) of 10^{-4}, for several orders of convergence and for high wave numbers in the 1-D scattering problem (37).

Wave Number k	P_w, Second Order	P_w, Fourth Order	P_w, 20^{th} Order
50	400	52.8	10.8
100	566	62.8	11.1
200	800	74.7	11.5
300	980	82.7	11.8
400	1132	88.8	12
500	1266	93.9	12.1

Table 3. Estimated value of the coefficients C_p in the expression $C_p k^{p+1} h^p$ for the relative error of convergence for high wave numbers in the 1-D scattering problem (37). The coefficients were calculated with respect to a scatterer formed by 11 Gaussian bumps.

Order p	2	4	6	8	16	20
C_p	8.1d − 3	1.0d − 2	1.2d − 2	1.6d − 2	6.0d − 2	1.0d − 1

Note that to maintain an accuracy of 10^{-4}, the second-order solver doubled the number of points per wavelength as $k = 50$ increased to $k = 200$, whereas there is essentially no increase in the number of points per wavelength for the 20^{th}-order solver. For a 2-D scattering problem (8) with $k = 50$ and using the same number of points per wavelength as in Table 2, the 20^{th}-order solver would be 2160 times faster than the second-order one, and 30 times faster than the fourth-order one.

6. NUMERICAL EXPERIMENTS

In this section, we present numerical tests in 2-D for the solvers constructed in Section 4. The tests use relative high values of the wave number. As discussed in Section 4, there is an additional difficulty for large wave numbers: for a fixed scatterer and fixed order of convergence of the solver, and for a fixed tolerance in the relative error of approximation, the number of points per wavelength increases as the wave number increases. This increment is far more significant for lower-order solvers than for higher-order ones (see Table 2). In the next numerical examples,

we use as scatterers a linear combination of $p \times p$ Gaussian functions:

$$q(x,y) = 0.4 \sum_{i,j=2}^{p+1} f_{i,j}(x,y), \tag{39}$$

such that for $i,j \in \{1,2,\ldots,p+2\}$, $f_{i,j}$ is an exponential function of the form

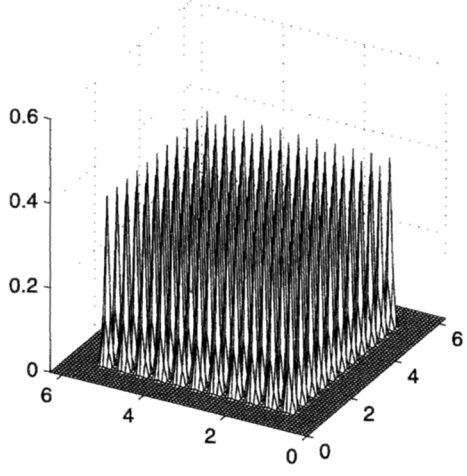
$$f_{i,j}(x,y) = e^{-w\left((x-c_i)^2+(y-c_j)^2\right)},$$

whose center $(c_i,c_j) \in [0,2\pi]\times[0,2\pi]$ has coordinates $(c_i,c_j) = (i\Delta_c, j\Delta_c)$, where $\Delta_c = 2\pi/(p+3)$. The constant w defined as $w = -30/(\Delta_c)^2$, makes $|f_{i,j}(x,y)| \leq e^{-30} \approx 10^{-13}$ if $\|(x,y)-(c_i,c_j)\| \geq \Delta_c$. The functions $f_{i,j}$ are not included in (39) when i or $j \in \{1,p+2\}$; this is to ensure that q and its derivatives (up to order 20) are close to zero near the boundary of $D = [0,2\pi] \times [0,2\pi]$.

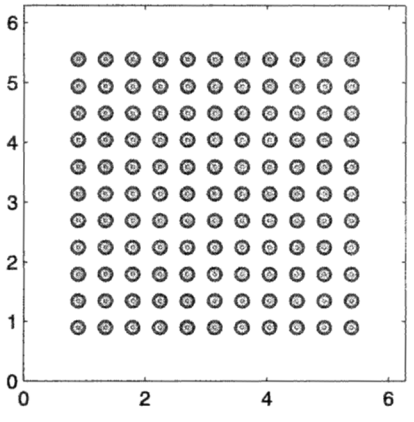
We tested the solver (35) using as scatterer functions 11×11 and 4×4 Gaussian functions (see Figure 2) and for wave numbers $k \in \{20,30,40,50\}$. We set the incident wave as

$$\phi_0(x_1,x_2) = e^{ik(x_1,x_2)\cdot(d_1,d_2)}, \qquad \text{with } (d_1,d_2) = \left(\cos\left(\frac{\pi}{4}\right), \sin\left(\frac{\pi}{4}\right)\right),$$

a plane wave with incident angle $\pi/4$. Numerical results are presented in Tables 4–7. We used several values of the number of grid points to discretize the computational domain $D = [0,2\pi]\times[0,2\pi]$, and computed the relative error of approximation to the charge density function σ in (8) using the solver (35) for orders 2, 4, 14, and 20. The number of grid points in D can be written as $(kP_w)^2$, where k is the wave number and P_w is the number of points per wavelength. Since the the linear system of equations (35) is solved by means of an iterative method, there will be two type of relative errors: one with respect to the approximation of the charge density function σ in (8), and one from the solution of the linear system of equations (35).



(a) Scatterer function defined in equation (39) with 11 × 11 Gaussian functions.



(b) Level curves of the scatterer function.

Figure 2.

Table 4. The table shows the L_2-relative errors of the numerical approximation to the charge density σ, for orders 2, 4, 14, and 20 of the solver, and for a wave number $k = 50$. It was used 11 × 11 Gaussian functions to generate the scatterer function q, and 10 points per wavelength in the computational domain $[0,2\pi] \times [0,2\pi]$.

k	P_w	Order	L_2-Relative Error	Iterations of GMRES(15)	CPU (seconds)
50	10	20	5.7×10^{-8}	44	58
50	10	14	1.0×10^{-6}	37	48
50	10	4	1.3×10^{-3}	21	28
50	10	2	1.3×10^{-1}	9	12

Table 5. Here the orders 2, 4, 14, and 20 of the solver are tested for a wave number $k = 50$. It was used 11×11 Gaussian functions to generate the scatterer function q, and eight points per wavelength.

k	P_w	Order	L_2-Relative Error	Iterations of GMRES(15)	CPU (seconds)
50	8	20	2.2×10^{-6}	36	32
50	8	14	1.2×10^{-5}	32	29
50	8	4	3.2×10^{-3}	18	16
50	8	2	2.0×10^{-1}	7	6

Table 6. A 20^{th}-order numerical approximation. In all cases, 11×11 Gaussian functions were used to generate the scatterer function q, and ten points per wavelength.

k	P_w	Order	L_2-Relative Error	Iterations of GMRES(15)	CPU (seconds)
50	10	20	5.7×10^{-8}	44	58
40	10	20	1.3×10^{-7}	32	29
30	10	20	4.3×10^{-7}	24	12.6
20	10	20	4.0×10^{-6}	21	4.4

Table 7. Numerical results, Example 2. The scatterer was generated using 4×4 Gaussian functions.

k	P_w	Order	L_2-Relative Error	Iterations of GMRES(15)	CPU (seconds)
50	10	20	8.9×10^{-9}	27	35.1
40	10	20	1.0×10^{-8}	25	15.2
30	10	20	1.6×10^{-8}	22	10.4
20	10	20	1.2×10^{-7}	17	3.6

We will use the GMRES method (see [18], for example) to iteratively approximate the solution of (35). The number of iterations of GMRES required to achieve a fixed tolerance is evidently dependent on the scatterer q but is shown independent of the number of sampling points in D. For a fixed tolerance of GMRES, the number of iterations seems independent of the number M of coefficients for logarithmic correction, that is, independent of the order of the solver. In all numerical results, GMRES(15), GMRES with restart after 15 iterations was used. The L_2-relative errors we present are those in the approximation to the charge density; CPU timings were taken on a PC Pentium 4 that runs at 2.0 Ghz.

EXAMPLE 1. We show, in Tables 4–6, the L_2-relative error of approximation to the charge density σ using a scatterer function q as defined in (39) with 11×11 Gaussian functions (see Figure 2). The wave numbers and number of points per wavelength are indicated on each table.

EXAMPLE 2. Table 7 shows the numerical calculations using the scatterer (39) with 4×4 Gaussian functions, and the 20^{th}-order solver. Results are shown for wave numbers $k = 50$, 40, 30, and 20. The number of points per wavelength was ten in all cases.

REFERENCES

1. F. Ihlenburg and I. Babuska, Finite element solution of the Helmholtz equation with high wave number Part II: The h-p version of the FEM, *SIAM Journal on Numerical Analysis* **34** (1), 315–358, (1997).
2. J.M. Song, C.C. Lu, W.C. Chew, *et al.*, Fast Illinois solver code (FISC), *IEEE ANTENNAS PROPAG* **40** (3), 27–34, (June 1998).
3. W. Symes, Iterative procedures for wave propagation in the frequency domain, Technical Report, Rice University, (1996).
4. Y. Chen, W. Chew and S. Zeroug, Fast multipole method as an efficient solver for 2-D elastic surface integral equations, *Comput. Mech.* **20** (6), 495–506, (1997).
5. W. Chew and J. Lin, Solution of the three-dimensional electromagnetic inverse problem by the local shape function and the conjugate gradient fast Fourier transform methods, *J. Opt. Soc. Am.* **14** (11), 3037–3045, (1997).

6. W. Chew and C. Lu, The use of the Huygens' equivalence principle for solving the volume integral equations of scattering, *IEEE ANTENNAS PROPAG* **AP-41** (7), 897–904, (July 1993).
7. P. Jones, J. Ma and V. Rokhlin, A fast direct algorithm for the solution of the Laplace equation on regions with fractal boundaries, *Journal of Computational Physics* **113** (1), (July 1994).
8. C. Lu and W. Chew, The use of the Huygens' equivalence principle for solving 3D volume integral equations of scattering, *IEEE Antennas Propag.* **43** (5), 500–507, (May 1995).
9. Y. Chen, A fast direct solver for the Lippmann-Schwinger equation, *Advances in Computational Mathematics* **16**, 175–190, (2002).
10. S. Kappur and D. Long, IES3: Efficient electrostatic and electromagnetic simulation, *IEEE Computational Science & Engineering* **5** (4), (1998).
11. J. Tausch and J. White, *Multipole Accelerated Capacitance Calculation for Structures with Multiple Dielectrics with High Permittivity Ratios, 33^{rd} Design Automation Conference*, (1996).
12. L. Greengard, J. Huang, S. Wandzura and V. Rokhlin, *IEEE Computational Science and Engineering* **5** (3), (July–September 1998).
13. M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*, Dover, New York, (1965).
14. J.C. Aguilar and Y. Chen, High-order corrected quadrature rules for functions with a logarithmic singularity in 2-D, *Computers Math. Applic.* **44** (8/9), 1031–1039, (2002).
15. W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, *Numerical Recipes in C*, Cambridge University Press, (1997).
16. S. Kapur and V. Rokhlin, High-order corrected trapezoidal quadrature rules for singular functions, *SIAM Journal of Numerical Analysis* **34** (4), 1331–1356, (1997).
17. B.K. Alpert, High-order quadratures for integral operators with singular kernels, *Journal of Computational and Applied Mathematics* **60**, 367–378, (1995).
18. D. Bau and L.N. Trefethen, *Numerical Linear Algebra*, SIAM, Philadelphia, PA, (1997).