



LUND UNIVERSITY

A comparison of splittings and integral equation solvers for a nonseparable elliptic equation

Englund, Jonas; Helsing, Johan

Published in:
BIT Numerical Mathematics

DOI:
[10.1007/s10543-004-5242-4](https://doi.org/10.1007/s10543-004-5242-4)

2004

[Link to publication](#)

Citation for published version (APA):
Englund, J., & Helsing, J. (2004). A comparison of splittings and integral equation solvers for a nonseparable elliptic equation. *BIT Numerical Mathematics*, 44(4), 675-697. <https://doi.org/10.1007/s10543-004-5242-4>

Total number of authors:
2

General rights

Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

A comparison of splittings and integral equation solvers for a nonseparable elliptic equation *

Jonas Englund and Johan Helsing[†]

Numerical Analysis, Centre for Mathematical Sciences,
Lund University, Box 118, S-221 00 LUND, Sweden

May 6, 2004

Abstract

Iterative numerical schemes for solving the electrostatic partial differential equation with variable conductivity, using fast and high-order accurate direct methods for preconditioning, are compared. Two integral method schemes of this type, based on previously suggested splittings of the equation, are proposed, analyzed, and implemented. The schemes are tested for large problems on a square. Particular emphasis is paid to convergence as a function of geometric complexity in the conductivity. Differences in performance of the schemes are predicted and observed in a striking manner.

Key words: Nonseparable elliptic PDE, variable coefficients, Fredholm integral equation, fast multipole method

1 Introduction

For the Poisson, Helmholtz, and modified Helmholtz equations on a rectangle or a circle there exist fast direct solvers which may be considerably more rapid and stable than general purpose iterative solvers based on the finite element and finite difference methods. Two such fast direct solver classes are fast Fourier transform (FFT) or cyclic reduction accelerated finite difference methods and fast multipole accelerated integral methods. While FFT accelerated finite difference solvers is the classic choice, Ethridge and Greengard [12] and Huang, Cheng, and Leiterman [23] recently showed that these can be outperformed by integral method solvers. The work per grid point for the solver classes is comparable, at least for moderate precision arithmetic. Since integral method solvers more easily allow for adaptive refinement and high-order accuracy, than do FFT solvers, the former win.

The usefulness of fast solvers goes beyond the simple equations and geometries mentioned above. Fast and adaptive direct solvers have shown to be competitive with general purpose methods, such as finite element- and finite difference methods in combination with

*This work was supported by the Swedish Research Science Council under contract 621-2001-2799.

[†]Email: helsing@maths.lth.se, Phone: +46-(0)46-2223372 Fax: +46-(0)46-2220595.

multigrid, also for more general separable variable-coefficient elliptic problems [22]. As for complex geometries, there exist fast direct solvers for the Poisson equation [26, 36]. Schemes for more difficult equations often apply a fast solver in an iterative fashion. Examples range from the challenging task of solving the incompressible Navier-Stokes equations inside a circular cylinder [13] to formally simpler setups with linear nonseparable elliptic equations [2, 6, 8, 10, 18, 28, 29, 30, 35, 38]. In this paper we consider a problem of this latter type; the electrostatic equation with Dirichlet boundary conditions

$$\nabla \cdot \sigma \nabla u = s, \quad \text{in } \Omega, \quad (1)$$

$$u = f, \quad \text{on } \Gamma, \quad (2)$$

where u is an unknown potential, σ is a variable conductivity, s is a source term (negative current source density), Ω is a simply connected domain with boundary Γ , and f is boundary data. Our purpose is to investigate what integral methods can do for (1,2) with emphasis on the performance of various fast and high-order accurate schemes as a function of σ . In the numerical examples we restrict our attention to the special case of Ω being a square.

The paper is organized as follows: Section 2 gives a review of previous work on iterative schemes and fast solvers for (1,2). Interestingly, integral methods have been discussed to some extent but, seemingly, never been tested. Section 3 discusses possible integral equation formulations and two of them are singled out for further study. Section 4 derives efficient quadratures for the Laplace equation on polygonal domains. Section 5 discusses discretization of area integrals and the evaluation of layer potentials close to their sources. Numerical examples, including comparison with previous results for simple setups as well as more challenging problems, are presented in Section 6. Our main conclusions are that

- integral methods for (1,2) can be stably implemented on rectangular domains with high-order accuracy also for reformulations that involve the convolution with the gradient of the logarithmic kernel;
- different reformulations of (1,2) can lead to algorithms with very different convergence properties for rapidly varying σ . This seems not to have been noticed by earlier investigators.

2 Review of previous work

Constructing a fast direct solver for the entire problem (1,2), that is, a rapidly computable inverse or Greens' function, is difficult when σ is nonseparable. Fast schemes for this problem typically rely on a splitting of the operator into a supposedly dominant part which is used as a left- or right preconditioner and can be inverted with a fast solver, and a remaining part which can be viewed as a perturbation. After discretization the resulting system of linear equations is solved iteratively. The composition of the fast solver with the dominant operator may be evaluated analytically.

2.1 Splittings

Possible splittings of (1), where the dominant operator appears on the left hand side, include

$$\Delta u = -\frac{\nabla\sigma \cdot \nabla u}{\sigma} + \frac{s}{\sigma}, \quad (3)$$

$$\Delta(ru) = u\Delta r + \frac{s}{r}, \quad (4)$$

$$\Delta(\sigma u) = \nabla \cdot (u\nabla\sigma) + s. \quad (5)$$

Here the quantity r is given by

$$r = \sigma^{1/2}.$$

Introducing the vector \bar{K}

$$\bar{K} = \left\langle \frac{\nabla\sigma}{\sigma} \right\rangle,$$

where angular brackets denote some kind of averaging, the equation (3) can be shifted and rewritten

$$(\Delta + \bar{K} \cdot \nabla) u = \left(\bar{K} - \frac{\nabla\sigma}{\sigma} \right) \cdot \nabla u + \frac{s}{\sigma}. \quad (6)$$

The equation (4) can also be shifted and rewritten as

$$(\Delta - K)w = (h - K)w + \frac{s}{r}. \quad (7)$$

Here

$$\begin{cases} w = ru, \\ h = \Delta r/r, \\ K = \langle h \rangle. \end{cases}$$

If the gradient of the solution to (1,2) is split up into the curl-free part of $\sigma\nabla u$ and the divergence-free part of ∇u as

$$\nabla u = \sigma^{-1}\nabla\phi + \nabla_{\perp}\psi,$$

where ϕ and ψ are scalar potentials and where the operator ∇_{\perp} , in a cartesian coordinate system, should be interpreted as $(\partial/\partial y, -\partial/\partial x)$, then one can get the system

$$\begin{cases} \Delta\phi = s + \nabla_{\perp}\sigma \cdot \nabla\psi, \\ \Delta\psi = -\nabla_{\perp}\sigma^{-1} \cdot \nabla\phi, \end{cases} \quad (8)$$

with boundary conditions given in terms of directional derivatives of ϕ , ψ , and f , see [2].

The splittings (3,4,5,8) all have the Laplacian as dominant operator. This operator is easy to deal with. It is hard to say which of the split equations has the smallest remaining operator but, as we shall see in Section 3, rewriting them as Fredholm second kind integral equations discloses more of their properties. The splitting (6) may have a constant- or separable coefficient operator as dominant operator. The splitting (7) has the Helmholtz, or modified Helmholtz, operator as dominant operator. An advantage with (6,7) is that \bar{K} and K can be chosen as to optimize the rate of convergence of the iterative solver. An advantage with (8), in the context fast Poisson solvers, is that a splitting along these lines easily can handle more general self-adjoint extensions of (1).

The performance of an algorithm relying on any of the splittings (3–8) for solving (1,2) will depend on many things. Five aspects have been particularly studied in the literature: The performance as a function of the fast solver used, the iterative scheme used, the choices of \bar{K} and K , variations in the source term s , and variations in the conductivity σ . We have a particular interest in the last point of this list. To simplify the discussion we characterize σ with two labels – a property called *conductivity ratio* and denoted κ which is the maximum value to the minimum value of σ over the domain, and a property called *geometric complexity* loosely defined by how rapidly σ varies.

2.2 Choices and conclusions of earlier investigators

Some authors choose the splitting (3). Zhao and Yedlin [38] use it for problems with low geometric complexity and $\kappa \leq 3$. The Laplacian is inverted with a direct Chebychev pseudo-spectral solver and fixed-point iteration is used for the preconditioned linear system. Comparison with a method which combines multigrid concepts with the iterative solutions of spectral equations shows that the fixed-point iteration is most efficient. Pares-Sierra and Vallis [29] use (3) in combination with finite differences and a fast Poisson solver. They try various iterative schemes for problems with large κ in a polygonal domain. Gauss-Seidel type iterations are reported to be most efficient, though the question of accuracy in the solution is not addressed. In [32] Rokhlin derives an integral equation for a PDE which has the form (3) as a special case.

Nielsen and Janssen [28] consider the splitting (4) in three dimensions for a conductivity that is chiefly constant but contains sharp transition zones and with $\kappa = 80$ and $f = 0$. The authors compare several iterative solvers. Their algorithms are pseudo-spectral and use a fast sine transform to invert the Laplacian. The authors conclude that most iterative solves are comparable with a slight advantage for the stabilized biconjugate gradient algorithm BiCGstab(l). We will return to [28] in Subsection 6.3.

Simkin and Trowbridge [34], Mayo [25], and Mikhailov [27] all discuss integral equations which, in a sense described in Section 3, are related to the splitting (5). Neither Mayo [25] nor Mikhailov [27] do actually solve any equations. Simkin and Trowbridge [34] only use integral equations to solve problems with piece-wise constant σ . The focus of Simkin’s and Trowbridge’s work is on comparing various formulations for electromagnetic problems, where different choices of potentials lead to partial differential equations of the form (1). The focus of Mayo’s work is on how to avoid computing area integrals explicitly. The focus of Mikhailov’s paper is on how to avoid a full system matrix when discretizing the integral equations. The tool for this is a domain splitting technique involving local parametrices for the operator in (1). A related, simplified method, has been tested numerically in [39].

Elman and Shultz [10] and Strain [35] study rather general, not necessarily self-adjoint, variable-coefficient second-order linear elliptic equations and choose a splitting which reduces to (6) for the problem (1). Strain chooses a constant \bar{K} while Elman and Shultz choose a spatially varying \bar{K} . Elman and Shultz [10] study nonperiodic problems. They derive bounds on convergence rates for iterative methods that are independent of mesh-size for a given problem. Strain [35] studies periodic problems and uses fast direct spectral solvers in combination with multigrid ideas. Strain’s interest lies in the performance of his algorithm as a function of the variations in coefficients and in the source term. He con-

cludes, from experiments, that the performance depends strongly on the variations in the source term and weakly on the variation in the coefficients. See Subsection 6.2 for a few more details.

Several authors choose the splitting (7). Concus and Golub [6] use it to construct a procedure based on second order accurate finite differences, a fast direct Helmholtz solver, and Chebychev iteration. They achieve rapid convergence in terms of iterations for reasonably smooth σ and with $\kappa < 10$ on a square. Pickering and Harley [30] extend this work. They choose a K that varies in one space direction and use a second order accurate fast direct Helmholtz solver. Improved convergence rates were found in some examples. Guillard and Désidéri [18] go further and choose a spatially varying K that corresponds to a separable approximation of σ . With a minimum residual iterative solver preconditioned by a spectral Helmholtz solver they set out to solve a set of periodic problems on the square. The performance, in terms of iterations, is similar to that depicted by the stars in our Figure 3, left image, below. Drawbacks with the scheme include a fair amount of precomputation and a performance that seem to degrade with resolution. Dimitropoulos and Beris [8] also try spectral methods for (7). They use the BiCGstab(l) iterative solver with two Concus- and Golub-type iterations as preconditioner. A fast direct spectral Helmholtz solver can be employed due to the particular choice of boundary conditions. The authors conclude that their method is much more efficient than the original iterative scheme of Concus and Golub [6] and much cheaper than that of Guillard and Désidéri [18]. The precise value of K need not always be optimal for fast convergence. See, further, Subsection 6.1. Some experiments with the splitting (7) are also reported by Pares-Sierra and Vallis [29]. These authors saw rapid convergence when K was nearly constant and no convergence in other cases.

Bernhardt and Brackbill [2] test a scheme relying on the splitting (8), finite differences, and fast direct Poisson solver using the transport equations for a plasma cloud in the presence of a neutral wind. Solving (1), where σ corresponds to an electron concentration, is a part of this problem. The preconditioned linear system is solved with fixed-point iteration. The authors report rapid convergence for their test case, but warn that a better iterative procedure may be necessary to insure convergence for other problems. June-Yub Lee (private communication) reaches a similar conclusion for (8) from tests on an equation that will be presented in Subsection 6.2.

Divo and Kassab [9] avoid splittings completely. They derive a boundary integral equation for (1,2) based on a generalized Green's function which is difficult to construct for general σ . Numerical examples for simple σ are given.

To sum up, the splitting (7) is a frequent choice by investigators working with FFT and cyclic reduction methods. The splitting (5) has been discussed by investigators interested in integral methods. The question of which splitting is most efficient for a given σ still seems open. Many authors, interested in high-order accuracy, work on the unit square and prefer problems that have periodic solutions, at least in one space dimension. The reason for this may be that fast direct high-order accurate solvers for the Poisson and Helmholtz equations are easier to construct in these settings than for more general boundary conditions [1, 4].

3 Integral equations

A popular starting point when deriving integral equations for elliptic problems on a bounded domain Ω with boundary Γ and outward normal ν is the integral representation formula for any twice continuously differentiable function $U(\mathbf{x})$ which can be derived from Green's first two identities, see chapter 8-3 in [17],

$$U(\mathbf{x}) = \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) U(\mathbf{y}) d\Gamma_{\mathbf{y}} - \int_{\Gamma} G(\mathbf{x}, \mathbf{y}) \frac{\partial U}{\partial \nu}(\mathbf{y}) d\Gamma_{\mathbf{y}} + \int_{\Omega} G(\mathbf{x}, \mathbf{y}) \Delta U(\mathbf{y}) d\Omega_{\mathbf{y}}. \quad (9)$$

Here a shorthand notation for the directional derivative with respect to ν is used and G is the free-space Green's function for the Laplacian which in two dimensions is

$$G(\mathbf{x}, \mathbf{y}) = \frac{1}{2\pi} \log |\mathbf{x} - \mathbf{y}|. \quad (10)$$

Integral equations for u of (1,2) could now be derived as follows: first let U in (9) be u , ru , or σu and replace ΔU in (9) with the right hand side of the splittings (3), (4), or (5). Then insert the known boundary values of u on Γ from (2) in the first integral on the right hand side of (9) and also on the left hand side of (9) for \mathbf{x} on Γ . The equation derived from (5) can be simplified by use of Green's first identity and the relation $\nabla_{\mathbf{x}} G(\mathbf{x}, \mathbf{y}) = -\nabla_{\mathbf{y}} G(\mathbf{x}, \mathbf{y})$. See, for example, [27, 34] for explicit expressions in 2D and 3D, and [25] for an example involving an unbounded region Ω .

Let us briefly examine the integral equations derived in the manner just described. If (3) is chosen the equation will be of integro-differential type. If (4) is chosen and if the normal derivative of u on Γ is considered unknown, then the integral equation will be of Fredholm's second kind on Ω and of Fredholm's first kind on Γ . The kernels involve $G(\mathbf{x}, \mathbf{y})$. Alternatively, the normal derivative of u on Γ can be computed numerically and the result is an integro-differential equation on Ω only. The same possibilities hold if (5) is chosen [27], the difference being that the kernel of the area integral now involves $\nabla_{\mathbf{x}} G(\mathbf{x}, \mathbf{y})$.

The representation formula (9) is not the only possible starting point for transforming an elliptic second-order partial differential equation into an integral equation. Another approach, common for constant-coefficient equations, is based on a representation of the solution in terms of abstract layer potentials. See [32] for examples in a variable-coefficient setting.

3.1 Our equations

To avoid possible numerical difficulties associated with first kind- and integro-differential equations we shall not rely solely on (9) for the construction of integral equations for (1,2). Rather, we use modified representation formulas which lead directly to Fredholm second kind equations. We use the purely abstract layer representation

$$u(\mathbf{x}) = \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) \mu(\mathbf{y}) d\Gamma_{\mathbf{y}} + \int_{\Omega} G(\mathbf{x}, \mathbf{y}) \rho(\mathbf{y}) d\Omega_{\mathbf{y}}, \quad (11)$$

for the splitting (3), where μ is an unknown double layer and ρ an unknown area density. We use the mixed abstract layer- and primitive variable representations

$$ru(\mathbf{x}) = \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) \mu(\mathbf{y}) d\Gamma_{\mathbf{y}} + \int_{\Omega} G(\mathbf{x}, \mathbf{y}) \left(u \Delta r + \frac{s}{r} \right) (\mathbf{y}) d\Omega_{\mathbf{y}}, \quad (12)$$

for the splitting (4), and

$$\sigma u(\mathbf{x}) = \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) \mu(\mathbf{y}) d\Gamma_{\mathbf{y}} + \int_{\Omega} \nabla_{\mathbf{x}} G(\mathbf{x}, \mathbf{y}) \cdot u \nabla \sigma(\mathbf{y}) d\Omega_{\mathbf{y}} + \int_{\Omega} G(\mathbf{x}, \mathbf{y}) s(\mathbf{y}) d\Omega_{\mathbf{y}}, \quad (13)$$

for the splitting (5).

One can easily check that (12,13) satisfy (4,5). The representation (11) can be inserted into (3) as to eliminate u . Enforcement of the boundary condition (2) leads to three systems of coupled Fredholm second kind integral equations for the unknowns μ and ρ or μ and u . Further, the unknown density μ can be eliminated and the three systems reduce to three single equations for ρ or for u . These read

$$\left(I + \frac{1}{\sigma} (\nabla \sigma \cdot \nabla) ((I - M_1 D) M_0) \right) \rho(\mathbf{x}) = \frac{1}{\sigma} \left(s - (\nabla \sigma \cdot \nabla) M_1 D f \right)(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (14)$$

$$\left(I - \frac{1}{r} (I - M_1 D) M_0 \Delta r \right) u(\mathbf{x}) = \frac{1}{r} \left((I - M_1 D) M_0 \frac{s}{r} + M_1 D r f \right)(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (15)$$

and

$$\left(I - \frac{1}{\sigma} (I - M_1 D) \nabla M_0 \cdot \nabla \sigma \right) u(\mathbf{x}) = \frac{1}{\sigma} \left((I - M_1 D) M_0 s + M_1 D \sigma f \right)(\mathbf{x}), \quad \mathbf{x} \in \Omega. \quad (16)$$

Here I is the identity operator. The notation M_0 is used for the integral operator

$$M_0 A(\mathbf{x}) = \int_{\Omega} G(\mathbf{x}, \mathbf{y}) A(\mathbf{y}) d\Omega_{\mathbf{y}}, \quad \mathbf{x} \in \Omega \cup \Gamma,$$

the notation ∇M_0 is used for the integral operator

$$\nabla M_0 \cdot \mathbf{A}(\mathbf{x}) = \int_{\Omega} \nabla_{\mathbf{x}} G(\mathbf{x}, \mathbf{y}) \cdot \mathbf{A}(\mathbf{y}) d\Omega_{\mathbf{y}}, \quad \mathbf{x} \in \Omega \cup \Gamma,$$

and notation M_1 is used for the integral operator

$$M_1 a(\mathbf{x}) = 2 \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) a(\mathbf{y}) d\Gamma_{\mathbf{y}}, \quad \mathbf{x} \in \Omega \cup \Gamma, \quad (17)$$

where A and \mathbf{A} are scalar- and vector valued functions on Ω and a is a scalar valued function on Γ . The inverse D is defined on Γ as

$$D a(\mathbf{x}) = (I + M_1)^{-1} a(\mathbf{x}), \quad \mathbf{x} \in \Gamma. \quad (18)$$

This inverse, which only involves the boundary, can be computed rapidly by an iterative numerical scheme, see Section 4. Alternatively, on the square, it can be computed directly, see [12].

We remark that, in addition to (14,15,16), yet two second kind integral equations for (1,2) can be obtained by inserting the abstract representation (11) into (4) or (5) and enforcing (2). The first of these equations is very similar to (15). The second equation is more complicated and contains operators appearing in (14) and (15).

3.2 Comparison of equations

When comparing (14,15,16) we see that (15) is the simplest equation in the sense that it only contains the two integral operators M_0 and M_1 . The equation (16) contains the three integral operators M_0 , ∇M_0 , and M_1 . The operator ∇M_0 is quite similar to M_1 . Its domain of integration differs. The equation (14) is the most difficult equation to implement. It contains the four operators M_0 , ∇M_0 , M_1 , and ∇M_1 . We speculate that the equations (14) and (16) have similar numerical properties. Their leading operators, $\nabla\sigma \cdot \nabla M_0$ and $-\nabla M_0 \cdot \nabla\sigma$, are each others adjoints. Therefore, in the following sections, we will chiefly consider the equations (15,16).

To highlight the connection between our equations (14,15,16) and Poisson solvers, and for later reference, we formally introduce $P(s, f)$ as an operator which solves the Poisson equation with source term s and Dirichlet boundary conditions f . One can now express (14,15,16) of the more translucent forms

$$\rho(\mathbf{x}) + \frac{1}{\sigma}(\nabla\sigma \cdot \nabla)P(\rho, 0)(\mathbf{x}) = \frac{1}{\sigma}(s - (\nabla\sigma \cdot \nabla)P(0, f))(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (19)$$

$$u(\mathbf{x}) - \frac{1}{r}P(u\Delta r, 0)(\mathbf{x}) = \frac{1}{r}P(s/r, rf)(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (20)$$

$$u(\mathbf{x}) - \frac{1}{\sigma}P(\nabla \cdot (u\nabla\sigma), 0)(\mathbf{x}) = \frac{1}{\sigma}P(s, \sigma f)(\mathbf{x}), \quad \mathbf{x} \in \Omega. \quad (21)$$

Alternatively, equations (20,21) can be derived directly from (4,5) and (2).

The efficiency of a Fredholm equation as a computational tool depends, among other things, on the spectrum of the compact integral operator. This spectrum will, for our equations, depend on σ . It is interesting to study the forms which (15,16) assume in the limit of a bimaterial where the source term s is zero and σ is a step function σ_{st} which takes on the value σ_2 in certain simply connected regions, called inclusions, and the value σ_1 in the rest of the plane. Such a study could give insight into the behavior of (15,16) for geometrically complex σ . It is known that Fredholm boundary integral equations involving the double layer kernel or its adjoint perform well for bimaterial problems of the type (1,2) in the sense that the number of iterations needed to solve the discretized equations is virtually independent of the number of inclusions (but sensitive to how closely spaced they are and to the conductivity ratio κ). Solution for bimaterials with a huge number of inclusions can be obtained [15, 19]. An example of an efficient boundary integral equation for an infinite bimaterial is

$$\left(I - \frac{(\sigma_2 - \sigma_1)}{(\sigma_2 + \sigma_1)} M_1 \right) u(\mathbf{x}) = \frac{2\sigma_1}{(\sigma_2 + \sigma_1)} (\mathbf{e} \cdot \mathbf{x}), \quad \mathbf{x} \in \Gamma_0. \quad (22)$$

where the uniform far field boundary condition $\nabla u = \mathbf{e}$ is applied at infinity, where Γ_0 denotes the inclusion boundaries, and where Γ is replaced by Γ_0 in the definition (17) of M_1 . The derivation of (22) uses the representation (9) and Green's second identity, see [15] and references therein. For reasonably smooth boundaries Γ_0 , the potential u is infinitely differentiable in the bimaterial except for at Γ_0 where it is merely continuous.

One can show, using distribution theory, that (16) in the bimaterial limit assumes a form very similar to (22). Consider the leading part of (16) for a setup with $s = 0$, with σ

being a smooth approximation of σ_{st} , and with a boundary condition $f = (\mathbf{e} \cdot \mathbf{x})$ applied far away from the inclusions

$$\left(I - \frac{1}{\sigma} \nabla M_0 \cdot \nabla \sigma \right) u(\mathbf{x}) = \frac{\sigma_1}{\sigma} (\mathbf{e} \cdot \mathbf{x}). \quad (23)$$

The integral operator in (23) has, in the bimaterial limit $\sigma \rightarrow \sigma_{\text{st}}$, only support on the inclusion interfaces. If we first multiply (23) with $2\sigma/(\sigma_2 + \sigma_1)$ from the left and then take the bimaterial limit, we end up with an equation which for \mathbf{x} on Γ_0 is identical to (22). This indicates that (16) should work well for a material with geometrically complex σ .

Similarly, if we, for the same setup, consider the leading part of (15)

$$\left(I - \frac{1}{r} M_0 \Delta r \right) u(\mathbf{x}) = \frac{r_1}{r} (\mathbf{e} \cdot \mathbf{x}), \quad (24)$$

multiply it from the left with $2r/(r_2 + r_1)$, take the bimaterial limit, and allow for u to develop a discontinuity in its normal derivative over Γ_0 at the same rate as that with which σ becomes discontinuous, we end up with an equation for \mathbf{x} on Γ_0

$$\left(I - \frac{(r_2 - r_1)}{(r_2 + r_1)} M_1 - 2 \frac{(r_2 - r_1)}{(r_2 + r_1)} M_0 \left\langle \frac{\partial}{\partial \nu} \right\rangle \right) u(\mathbf{x}) = \frac{2r_1}{(r_2 + r_1)} (\mathbf{e} \cdot \mathbf{x}), \quad \mathbf{x} \in \Gamma_0. \quad (25)$$

Here angular brackets denote the mean of the value on the positive and the negative side of Γ_0 , M_1 is as in (22), and M_0 is now a boundary integral with support on Γ_0 . The limiting form (25) is more complicated than (22). It contains, in addition to terms similar to those in (22), a composition of a compact operator with a differential operator which for the bimaterial problem can be interpreted as a set of Dirichlet-Neumann maps for Laplace's equation. It has been noted that such maps are ill-conditioned on multiply connected domains in the sense that iterative methods for Fredholm second kind equations based on single- and double layer potentials tend to scale "appreciably worse than linear" as the number of closed boundaries increase [36]. One can therefore speculate that (15) should perform worse than (16) for a material with geometrically complex σ .

4 Double layers on polygonal domains

We shall now investigate the solution to the Laplace equation on a polygon Ω as obtained with the double layer ansatz

$$u(\mathbf{x}) = \int_{\Gamma} \frac{\partial G}{\partial \nu_{\mathbf{y}}}(\mathbf{x}, \mathbf{y}) \mu(\mathbf{y}) d\Gamma_{\mathbf{y}}.$$

This problem has relevance for the efficient implementation of the operator D of (18). We assume that the Dirichlet boundary data f is the limiting value of a smooth function in Ω . The corresponding boundary integral equation is simply

$$(I + M_1) \mu(\mathbf{x}) = 2f(\mathbf{x}), \quad \mathbf{x} \in \Gamma. \quad (26)$$

This equation is of Fredholm's second for smooth boundaries Γ .

In order to construct an economic quadrature for the integral in (26) we need to investigate the asymptotic behavior of μ close to the corners. Cormack and Rosen [7] suggest an approach involving gauge conditions as a general tool for this type of problem. We shall proceed more conventional way. An equation with a similar structure as (26) is (22) for a single inclusion. In the limit of vanishing σ_2 the two equations have the same left hand sides, with u corresponding to μ . Assume now that the inclusion geometry involves corners. The asymptotic shape of u in a bimaterial corner of opening angle 2β can be found by applying variable separation to the electrostatic PDE on a bimaterial wedge using ansatzes in polar coordinates of the type $u(r, \phi) = r^\gamma \cos(\gamma\phi)$ and $u(r, \phi) = r^\gamma \sin(\gamma\phi)$ with $\gamma \geq 0$ to guarantee continuity, see [20]. The analysis leads to systems of trigonometric equations. For a corner of a square, where $2\beta = \pi/2$, and in the limit of a vanishing σ_2 one can show that possible values for the exponents γ are all positive integer multiples of $2/3$. We conclude that if composite quadrature is used to discretize (26), an efficient N -point quadrature rule

$$\int_{-1}^1 f(s) ds = \sum_{n=1}^N f(s_n) w_n$$

for a quadrature segment having a corner as its starting point should be exact for the first few basis functions $(s+1)^{2n/3}$, $n = 0, 1, 2, \dots$

The preceding analysis is only asymptotic. We therefore add monomial functions s^n , $n = 0, 1, 2, \dots$ as to better capture the actual behavior of μ . We settle for a 16-point generalized Gaussian quadrature rule and take s^n , $n = 0, 1, 2, \dots, 27$ and $(s+1)^{2n/3}$, $n = 1, 2, 4, 5$ as the 32 basis functions for which the quadrature should be exact. The nodes s_i and the weights w_i are computed with a homotopy method suggested by Yarvin and Rokhlin [37]. We remark that other corner singularities than those discussed here can occur in the context of solving the Poisson or electrostatic equation on a polygonal domain. For example, it may happen, even when σ , f , and s in (1,2) are smooth, that the fields which $M_1 D$ acts on in (15,16) contain terms involving logarithms. Including more types of singular basis functions in the generalized quadrature could resolve the operator D in these cases too. We refrain from doing this. We shall see in Section 6 that the present quadrature resolves our numerical examples sufficiently well.

5 Discretizing and solving integral equations

This section discusses numerical details in discretization and solution process of the equations (15,16). We shall use a Nyström scheme with uniform refinement and composite quadrature. The computational domain Ω is a square and it will be divided into N^2 equisized smaller squares Ω_i . The outer boundary Γ will be divided into $4N$ equisized straight segments Γ_i . The segments Γ_i are placed in such a way that they coincide with boundaries of squares Ω_i which have sides on Γ . On each square Ω_i a tensor product grid with m^2 discretization points are placed corresponding to a total of $m^2 N^2$ interior discretization points. On each segment Γ_i a number n of discretization points are placed, corresponding to a total of $4nN$ boundary points. The discretization points will be referred to as *source points* when they are associated with the variable of integration $\mathbf{y} = (y_1, y_2)$ and as *target points* when they are associated with the variable of evaluation $\mathbf{x} = (x_1, x_2)$.

5.1 General strategy

The following quadratures will be our main tools: We use a Cartesian product rule for source points on squares Ω_i . Aiming at high-order accuracy we choose $m = 12$ and use 12-point Gauss-Legendre quadrature in each direction. We take $n = 16$ and use 16-point Gauss-Legendre quadrature for source points on segments Γ_i which do not contain a corner of the computational domain Ω . We use the 16-point generalized Gaussian quadrature as described in Section 4 on the eight segments Γ_i which do contain a corner of Ω .

An example of a discretized area integral, from (15), is

$$\int_{\Omega_i} G(\mathbf{x}, \mathbf{y}) u \Delta r(\mathbf{y}) d\Omega_{i\mathbf{y}} = \sum_{j_1=1}^{12} \sum_{j_2=1}^{12} w_{j_1} w_{j_2} G(\mathbf{x}, \mathbf{y}_{j_1 j_2}) u \Delta r(\mathbf{y}_{j_1 j_2}), \quad (27)$$

where w_{j_1} and w_{j_2} are scaled one-dimensional Gaussian weights, $\mathbf{y}_{j_1 j_2} = (y_{j_1}, y_{j_2})$, and y_{j_1} and y_{j_2} are one-dimensional Gaussian nodes that are scaled and translated into the square Ω_i .

5.2 Singular kernels and precomputation

Two major problems follow from the strategy of Subsection 5.1. First, the logarithmic kernel and its derivative in the area integrals in (15,16) are not smooth. Therefore a Cartesian product rule on a square Ω_i becomes very inexact when target points are close to or on that square. Second, neither the kernel of the double-layer potential is everywhere smooth. The kernel of the layer potential has sharp peaks for interior target points close to its sources. A similar behavior holds for source points and target points on neighboring segments located on opposite sides of a corner. An interpolatory rule takes care of these situations. An example for an area integral is

$$\int_{\Omega_i} G(\mathbf{x}, \mathbf{y}) u \Delta r(\mathbf{y}) d\Omega_{i\mathbf{y}} = \sum_{j_1=1}^{12} \sum_{j_2=1}^{12} W_{j_1 j_2}(\mathbf{x}) u \Delta r(\mathbf{y}_{j_1 j_2}), \quad (28)$$

where $W_{j_1 j_2}(\mathbf{x})$ are 144 weights that must be precomputed for each target point \mathbf{x} that is close to or in a square Ω_i prior to running the main program. Fortunately, since we are using a uniform mesh, only one set of weights $W_{j_1 j_2}(\mathbf{x})$ need to be computed. These weights are independent of the refinement level up to some mesh dependent constants. One can show

$$W_{j_1 j_2}(\mathbf{x}) = W_{0j_1 j_2}(\mathbf{x})/d^2 - w_{j_1} w_{j_2} \log d/2\pi,$$

where $W_{0j_1 j_2}(\mathbf{x})$ are weights on a square with unit side-length and where d is the side-length of the square Ω_i .

The interpolating functions are taken as follows: On a square Ω_i we interpolate quantities such as $u \Delta r(\mathbf{y})$ at the 144 source points with 144 basis functions of the type

$$y_1^{m_1} y_2^{m_2}, \quad 0 \leq m_1, m_2 \leq 11.$$

On a segment Γ_i that does not contain a corner of Γ and is parameterized with a local parameter s we interpolate the double layer at the 16 source points with the 16 monomial basis functions

$$s^n, \quad 0 \leq n \leq 15.$$

On a segment Γ_i that does contain a corner of Γ we interpolate the double layer at the 16 source points with the 16 basis functions

$$\begin{cases} s^n, & n = 0, 1, \dots, 11, \\ (s+1)^{2n/3}, & n = 1, 2, 4, 5. \end{cases}$$

Having thus described how we perform quadrature for source- and target points that are close to each other we need to define what we mean by 'closeness'. We adopt the following: a target point \mathbf{x} is close to a square Ω_i if it is located in Ω_i , in one of the up to eight possible nearest neighbor squares of Ω_i , or on a boundary of Ω_i . Further, a target point \mathbf{x} is close to a segment Γ_i if it is located in a square Ω_i which has a side that coincides with or neighbors to Γ_i , or if it is located on another segment which neighbors to Γ_i on the opposite side of a corner. With this definition of closeness we can evaluate all integrals appearing in (15,16) with double precision accuracy for sufficiently resolved layer densities.

We shall not detail how all sets of precomputed weights were computed. We just note that many of them can be evaluated analytically, though the primitive functions for integrals containing basis functions of higher degrees are rather complicated. We have chosen an adaptive numeric approach. The number of precomputed real numbers needed for the weights in (15) is on the order of $13 \cdot 144^2$ and about three times as much for (16). This number can be reduced if symmetry is taken more into account.

5.3 Fast multipole method and other speedups

We shall solve the discretized systems (15,16) with the GMRES iterative solver [33]. Each iterative step requires several large matrix vector multiplications and corresponds to solving the Poisson equation with homogeneous boundary conditions once, see (20,21). We use the fast multipole method [5, 16, 31] to achieve linear complexity. In the present code, and for all kernels, we use a non-adaptive implementation based on the original scheme of [16] but which includes some ideas of [21]. See [11] for details. We set the requested tolerance very high, often to $5 \cdot 10^{-16}$, as to test the stability of our code. In a typical example of a matrix-vector multiplication we first evaluate $M_0 u \Delta r(\mathbf{x})$ using the fast multipole method and the formula (27) for all source and target points irrespective of whether they are close to each other or not. Then we add the sum of all integrals (28) minus a correction, which is the part of (27) which should not have been included. This correction is translationally invariant over the squares Ω_i and can, in fact, be included into the weights $W_{j_1 j_2}$.

We remark, again, that Ethridge and Greengard [12] have constructed an extremely fast direct Poisson solver on the square based on the integral operator M_0 . This solver includes features such as adaptive refinement on a quad-tree, diagonal forms of translation operators, the method of images, the use of fewer basis functions in each square Ω_i in combination with singular value splitting and fast cosine transforms for computing local interactions, and multipole expansions of entire area integrals (rather than of kernels) for computing far-field interactions. We expect that huge speedups would result if this solver was used for our equations (15,16). The solver can be applied directly to (15) by using the equivalent form (20). It can also be used for (21). But, although mathematically equivalent to (16), this equation may not be as good for numerics as (16). In (21), derivatives of the unknown quantity u need to be computed. In (16), this numerical differentiation is bypassed at the cost of introducing the operator ∇M_0 .

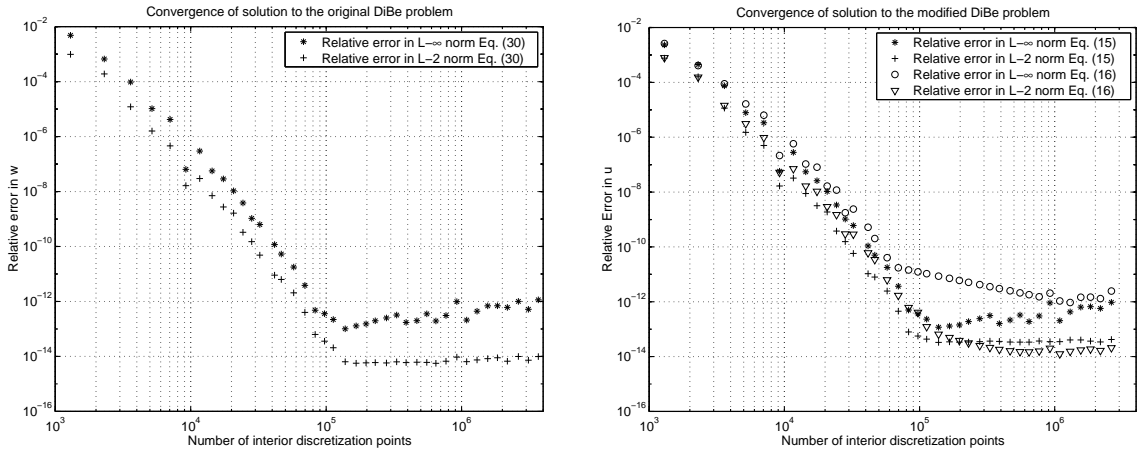


Figure 1: Convergence of solution to the Dimitropoulos and Beris problem. Left: original problem with h as in (29) and solution w given by (31). Equation (30) is used for numerics. Right: modified problem with h as in (33). Equations (15) and (16) are compared.

6 Numerical examples

This section compares the performance of algorithms based on (15) and on (16) for a variety of previously defined and new problems. Our computational domain Ω is always the square $[-1/2, 1/2] \times [-1/2, 1/2]$. Previous problems defined on other squares have been shifted and scaled. Comparing performance, in terms of speed and achievable accuracy, with algorithms of other investigators is indeed difficult. For example, the cost of an “iteration” depends strongly on how many fast solver calls are involved and how rapidly they are executed. Some authors, using FFT based solvers, measure speed in terms of how many FFT calls are needed for a given tolerance. When such solvers are used, one fast Poisson call typically corresponds to two or three FFT calls???

The GMRES iterative solver is used twice in our algorithms. It is used to solve the main equations. It is also used inside each iterative step to compute the inverse D of (18). This latter computation only involves points on Γ . The time it consumes for large systems is vanishingly small compared to the total computational time. Unless otherwise stated, the numbers of GMRES iterations reported below refers to the main equation.

6.1 The Dimitropoulos and Beris problem

We start out by testing our algorithms on two problems which have analytical solutions. Dimitropoulos and Beris [8], see Subsection 2.2, suggest a problem where the function h of (7), called g_2 in [8], is taken as

$$h(\mathbf{x}) = -\frac{8\pi \sin [4\pi(x_1 + x_2)]}{1.5 + \sin [4\pi(x_1 + x_2)]}. \quad (29)$$

This function h fluctuates around zero in the unit cell. A constant shift K in (7) is therefore small compared to the amplitude of h . An algorithm relying on (7), such as the one in [8],

should perhaps then perform similarly to an algorithm relying on (4) such as our algorithm for (15).

Unfortunately, the function h of (29) does not correspond to any function σ . Therefore we can not apply any of the equations (15,16) as they stand. It is, however, simple to change (15) as to solve for the unknown $w = ru$

$$(I - (I - M_1 D)M_0 h) w(\mathbf{x}) = (I - M_1 D)M_0 s^*(\mathbf{x}) + M_1 D f^*(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (30)$$

where f^* is the limiting value of w on Γ and s^* is a new source term. When it comes to choosing s^* we follow [8] and take the function s^* which corresponds to a solution w given by (P_5 in their paper)

$$w(\mathbf{x}) = \frac{(4x_2^2 - 1)(2x_2 - 1)}{4x_2^2 \cos[8\pi x_1] + 2.2}. \quad (31)$$

This solution is constructed so that it is periodic in the x_1 -direction and zero on the boundaries in the x_2 direction.

The convergence of the relative L_∞ error and the relative L_2 error in w is shown in Figure 1, left image. The convergence rate is almost the same as the one obtained by Dimitropoulos and Beris with their spectral algorithm. Compare with their Figure 9. Our achievable accuracy may be better. It is hard to tell since Dimitropoulos and Beris do not display data beyond 16,512 discretization points. The number of GMRES iterations needed to achieve a relative residual of $5 \cdot 10^{-16}$ in the main equation was 14 for all mesh sizes. This behavior is typical for Fredholm equations of the second kind – the distribution of eigenvalues of the system matrix does not change with refined discretization. Dimitropoulos and Beris report a number of at most 9 iterations for their scheme, but it is unclear how many fast solver calls actually are made. A number of 3 – 18 calls per iteration is hinted at. Thus it seems as if our scheme requires a smaller number of calls for convergence.

We now proceed to modify the Dimitropoulos and Beris problem by introducing the function

$$\sigma(\mathbf{x}) = (1.5 + \sin[4\pi(x_1 + x_2)])^2, \quad (32)$$

which corresponds to

$$h(\mathbf{x}) = -\frac{32\pi^2 \sin[4\pi(x_1 + x_2)]}{1.5 + \sin[4\pi(x_1 + x_2)]}, \quad (33)$$

and differs from h in (29) by being a factor of 4π larger. The source term s is modified so that the solution is the same as in the previous problem. With these modifications we apply our equations (15,16) and solve for $u = w/r$. The problem now has a maximum conductivity ratio of $\kappa = 25$. Convergence plots for these equations are displayed in Figure 1, right image. We see that the solution converges stably for both equations. For (16) there are a few points close to the domain corners where pointwise convergence is slow once the L_∞ error reaches a value of about 10^{-11} . This is so since the field which $M_1 D$ acts on in the corners contains minor logarithmic parts which are not properly taken into account by the quadrature, see Section 4. The number of GMRES iterations needed in the main equation for a relative residual less than $5 \cdot 10^{-16}$ was virtually independent of the resolution. A number of 40 iterations were needed for (15) and a number of 38 iterations were needed for (16).

In conclusion, this example shows that integral methods can compete with spectral FFT based methods in terms of convergence, that is, in the number of discretization points and

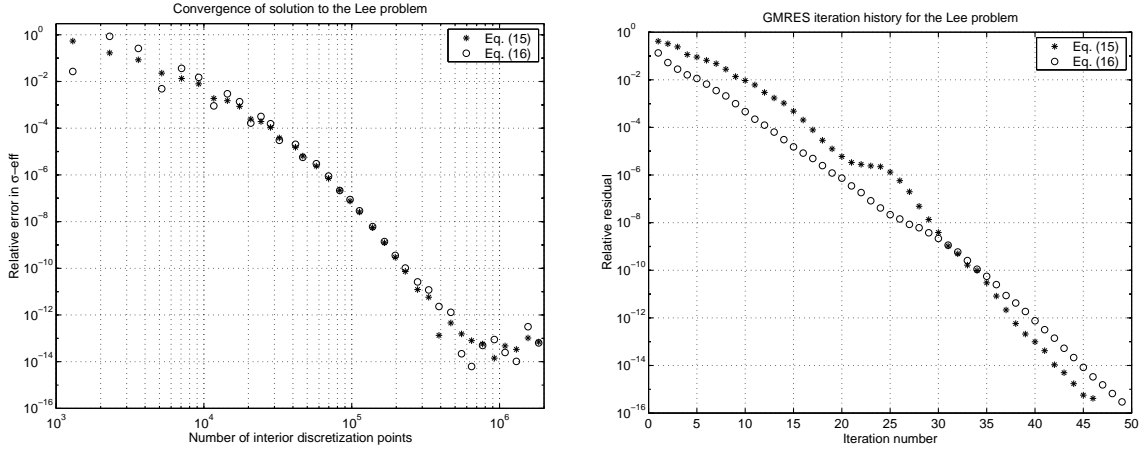


Figure 2: Left: Convergence of the quantity σ_{eff} of (37), computed from the solution to the Lee problem with σ and s as in (34) and (35) and boundary conditions given by (36). A conductivity ratio of $\kappa = 256$ and a wavelength of $\lambda = 2/3$ are chosen. The reference value of the solution is taken as $\sigma_{\text{eff}} = 0.71301409870461$. Right: A GMRES iteration history.

in the number of fast solver calls needed to achieve a given accuracy in the solution. This is interesting since the problem was tailor-made to fit a fast direct spectral solver and since the earlier investigators relied on a splitting (7) which allows for more fine-tuning than the splittings (4) and (5), which form the basis of our algorithms.

6.2 A problem with large and rapid variations in the conductivity

June-Yub Lee suggested a problem (private communication) with conductivity in the unit square given by

$$\sigma(\mathbf{x}, \kappa, \lambda) = \left(1 + \frac{1}{\kappa}\right) + \left(1 - \frac{1}{\kappa}\right) \cos \left[\frac{2\pi}{\lambda}(x_1 - x_2) \right], \quad (34)$$

where λ is a wavelength which describes the geometric complexity and κ is the ratio of the largest to the smallest value of σ . The source term s is given by

$$s(\mathbf{x}) = x_1 x_2 (1 - 4x_1^2)^4 (1 - 4x_2^2)^4, \quad (35)$$

and the boundary conditions are

$$u(\mathbf{x}) = x_2, \quad \text{on } \Gamma. \quad (36)$$

In absence of an analytical solution we study the convergence of the quantity σ_{eff} given by

$$\sigma_{\text{eff}} = \langle \nabla u \cdot \sigma \nabla u \rangle, \quad (37)$$

where angular brackets denote area average. The gradients of the solution are computed numerically.

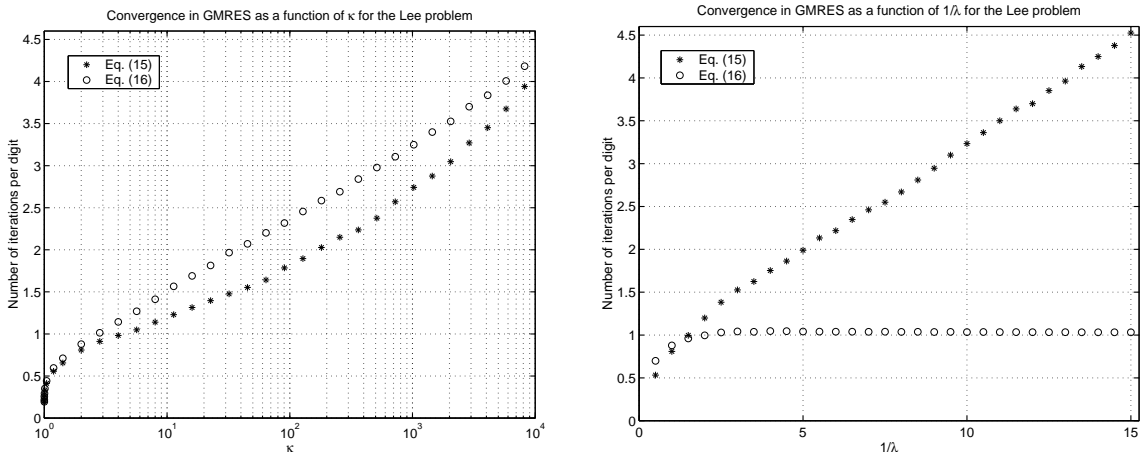


Figure 3: Number of GMRES iterations per digit as defined in (38) with σ and s as in (34) and (35) and boundary conditions given by (36). Left: The wavelength is held fixed at $\lambda = 1$, while the conductivity ratio κ varies. Right: The conductivity ratio is held fixed at $\kappa = 2$, while the wavelength λ , which describes the geometric complexity, varies.

First we compare (15,16) in terms of achievable accuracy. It turns out that the results produced by the two equations are very similar. Figure 2 shows a convergence study of σ_{eff} for $\kappa = 256$ and $\lambda = 2/3$ with up to 1,838,736 interior discretization points on Ω . The number of GMRES iterations needed for the relative residual to be less than $5 \cdot 10^{-16}$ was 46 for (15) and 49 for (16). A full GMRES iteration history is shown in Figure 2.

Then we compare the convergence properties of (15,16) with respect to how many iterations in the GMRES iterative solver that are needed to reduce the relative residual to a given value, as a function of κ when $\lambda = 1$. The problem gets harder as κ grows. The number of GMRES iterations for a given κ is, thanks to the Fredholm property, approximately independent of the number of discretization points. For large κ and severe underresolution, the number of iterations may vary a little, but as soon as the solution is resolved to approximately three digits, the number stays constant. Figure 3, left image, shows that the two equations have essentially the same properties. The quantity displayed, *iterations per digit*, is computed as

$$\text{iterations per digit} = -\frac{\text{iter}}{\log_{10}(\text{res})}, \quad (38)$$

where *res* is the first relative residual appearing that is less than a given number, here taken as 10^{-13} , and *iter* is the number of iteration required to reach this residual.

We now compare the number of GMRES iterations that are needed to reduce the relative residual to a given value as a function of $1/\lambda$ when $\kappa = 2$. A decreasing wavelength corresponds to a more complex geometry, but not necessarily to a harder problem. Figure 3, right image, shows a dramatic difference in performance between the two equations. The number of iterations per digit for (15) grows approximately linearly with $1/\lambda$, while for (16) it is approximately constant. This suggest that algorithms relying on the splitting (4) (and perhaps also on its shifted form (7), which is a frequent choice by previous investigators, see Subsection 2.2) can be very bad. A similar study of how the computing time scales with

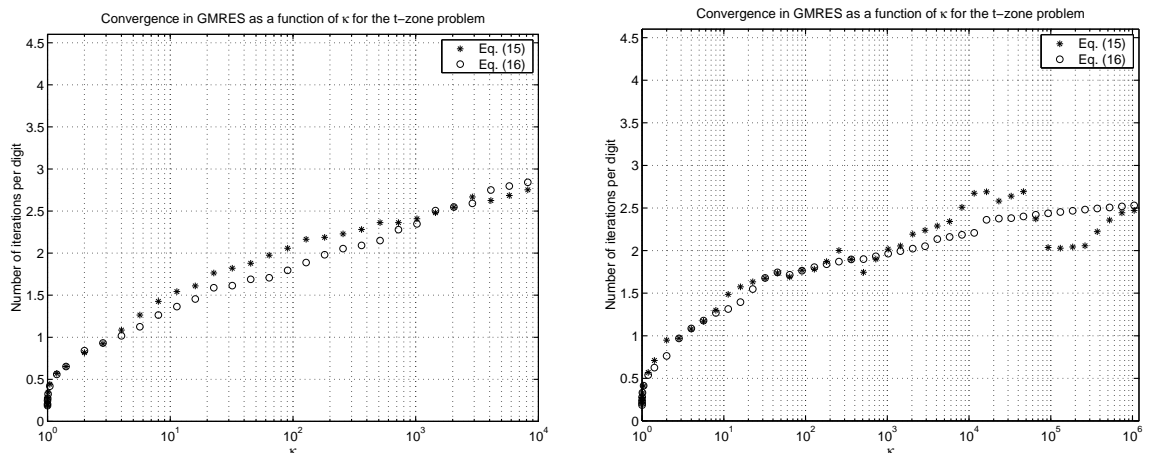


Figure 4: Number of GMRES iterations per digit as defined in (38) with σ as in (39), $s = 0$, and boundary conditions given by (36). Left: The iterations are terminated when the relative residual is less than 10^{-13} . Right: The termination criterion is changed to $5 \cdot 10^{-7}$.

the variation in the coefficients of a second-order elliptic PDE is included in the paper by Strain [35], mentioned in Subsection 2.2. Due to the more general nature of Strain’s PDE it is difficult to make a comparison with our findings. Anyhow, Strain, in his Figure 1, observes a computing time which grows approximately linearly with the coefficient wavenumbers, everything else held constant.

In conclusion, this example shows that the algorithm for (16) relying on the splitting (5) drastically outperformed the algorithm for (15) relying on the splitting (4) for a problem with rapid spatial variation in σ . See Subsection 3.2 for a possible explanation of this difference.

6.3 A problem with a sharp transition zone

Nielsen and Janssen [28], see also Subsection 2.2, treat a three dimensional problem where the conductivity σ is essentially piecewise constant in different regions of the material. Between these regions σ has sharp transition zones.

Here we construct a similar problem in 2D. The purpose is to study the convergence properties of (15,16) as a function of the conductivity ratio κ . The problem gets harder as κ increases, at least initially. Its difficulty also depends on the requested accuracy. If the requested accuracy is low, say in L_2 norm, the transitions are sharp, and the regions where σ is large are reasonably separated, then σ could be viewed as a step function. The problem should be simple to solve, at least with equation (16) which according to Subsection 3.2 resembles a double-layer boundary integral equation in this situation. As the requested accuracy is increased the need to resolve the potential u well in the transition zones increases and the problem gets progressively harder.

We choose a conductivity

$$\sigma(\mathbf{x}) = (\kappa - 1)\xi(\mathbf{x}) + 1, \quad (39)$$

where ξ is a smooth approximation of a step function with a transition from one to zero at $|\mathbf{x}| = 0.25$ and given by

$$\xi(\mathbf{x}) = g_0(g^7(g_1(\mathbf{x}))), \quad (40)$$

where $g^n = g \circ g^{n-1}$, and where g_0 , g , and g_1 are functions defined by

$$g_0(x) = 0.5x, \quad (41)$$

$$g(x) = 1.5x^2 - 0.5x^3, \quad (42)$$

$$g_1(\mathbf{x}) = 2^{1-16(x_1^2+x_2^2)}. \quad (43)$$

The function ξ thus constructed differs from one or zero with a quantity larger than machine epsilon in double precision arithmetic only in the interval $0.18 \leq |\mathbf{x}| \leq 0.32$. (Nevertheless, σ diverges everywhere as $\kappa \rightarrow \infty$.) The source term s is set to zero and the boundary conditions are the same as in (36).

Figure 4 shows the number of iterations per digits, defined in (38), as a function of κ for the two equations (15) and (16). In the left image we iterate until the relative residual is less than 10^{-13} while in the right image we terminate at $5 \cdot 10^{-7}$. Up to 3,686,400 interior discretization points were used in these calculations. One sees that the problem gets harder as κ is increased and also progressively harder as higher accuracy is requested. The algorithms for (15) and (16) behave similarly. As a benchmark for σ_{eff} we present numerical results for $\kappa = 8192$. Here we get convergence to $\sigma_{\text{eff}} = 1.69113014$ with both equations.

While this example in many respects differs from the example of Nielsen and Janssen [28] we still summarize some of their numerical findings. The authors consider several three dimensional problems with conductivities that always involved 16 spherical regions with transition zones and a function ξ with two compositions of the function g of (42) and a $\kappa = 80$. The number of FFT calls needed for a relative residual less than 10^{-12} was around 125. The authors state, without numeric evidence, that the computational cost of their algorithm scales as $O(N \log_2 N)$ where N increases linearly with the number of spherical regions.

In conclusion, we have demonstrated that our integral method can solve a simple electrostatic problem with very high conductivity ratio κ in a moderate number of iterations. For the most extreme problem, with $\kappa \approx 10^6$ in Figure 4, we needed 16 GMRES iterations to get an estimated relative error in σ_{eff} of at least 10^{-6} . Naturally, adaptive discretization would lead to enormous savings in this particular example.

7 Conclusions and extensions

We have solved the nonseparable partial differential equation (1,2) by an approach based on a splitting. Roughly speaking, the split equation is solved iteratively with the dominant operator acting as preconditioner. Its inverse is applied with an integral method. Drawbacks with integral methods for (1,2) at present, compared with FFT-based methods, are that the dominant operator has to be the Laplacian or the Helmholtz operator, and that the most efficient fast multipole codes are not publicly available. Their advantages include that they

are less restricted to rectangles and circles and that they more easily allow for adaptive refinement and high-order accuracy,

Our paper contains the following new material: we have actually implemented integral methods for the problem at hand; our numerical examples are larger and more difficult than those of previous investigators; we have compared the performance of algorithms based on different splittings for geometrically complex σ . In particular, we have compared an algorithm implemented from (15) and relying on the splitting (4), which is a special case of a classic choice [6], with an algorithm implemented from (16) and relying on the splitting (5), which has been discussed in the integral equation community. A difference between the two schemes is revealed by studying the limit in which σ is a step function. In this limit (16) essentially assumes the form of the identity plus a double layer operator, while (15) contains an extra operator which adds ill-conditioning. Numerical experiments with geometrically complex σ supports the conclusion that (16) is superior to (15). We speculate that this difference reflects properties of the underlying splittings themselves, and that it should be visible also in algorithms using FFT-based techniques.

Our computational domain is a square. The extension to general boundaries is extremely important if integral methods are to become a serious competitor to alternative techniques such as the finite element method in combination with multigrid. The problem boils down to finding a fast and high-order accurate Poisson solver on irregular regions. There are several options. McKenney, Greengard, and Mayo [26] have constructed a fast Poisson solver on irregular regions that, for the free-space problem, uses a combination of potential theory and classic fast solvers. The adaptive Poisson solver of [22] may also be used. Another approach, without FFTs, is suggested in [14]. Here the solution is represented as a sum of polynomials that satisfies the Poisson equation locally and harmonic parts which are generated by the boundary values of the polynomials. This omits the need of evaluating convolutions with logarithmic kernels over areas, which is associated with pure integral methods. Instead one has to find the local polynomial, which is a non-standard task. For problems with source terms that admit smooth extrapolation, one could solve an extended Poisson equation with any standard technique on a larger, more regular, domain. See Mammoli [24] and Biros, Ying, and Zorin [3] for attempts in this direction.

References

- [1] Averbuch A, Israeli M, Vozovoi L. ‘A fast Poisson solver of arbitrary order accuracy in rectangular domains’, *SIAM Journal on Scientific Computing* 1998;**19**(3):933–952.
- [2] Bernhardt PA, Brackbill JU. ‘Solution of Elliptic Equations Using Fast Poisson Solvers’, *Journal of Computational Physics* 1984;**53**(3):382–394.
- [3] Biros G, Ying L, Zorin D. ‘A fast solver for Stokes equations with distributed forces in complex geometries’, *Journal of Computational Physics* 2003;**193**(1):317–348.
- [4] Canuto C, Hussaini MY, Quarteroni A, and Zang TA, *Spectral methods in fluid dynamics*, Springer-Verlag, New York, 1988.
- [5] Carrier J, Greengard L, Rokhlin V. ‘A fast adaptive multipole algorithm for particle simulations’, *SIAM Journal on Scientific and Statistical Computing* 1988;**9**(4):669–686.
- [6] Concus P, Golub G. ‘Use of fast direct methods for the efficient numerical solution of nonseparable elliptic equations’, *SIAM Journal on Numerical Analysis* 1973;**10**(6):1103–1120.
- [7] Cormack DE, Rosen D. ‘Gauge conditions and the analysis of singular fields with boundary integral equations’, *Engineering Analysis with Boundary Elements* 1996;**18**(1):1–8.
- [8] Dimitropoulos CD, Beris AN. ‘An efficient and robust spectral solver for nonseparable elliptic equations’, *Journal of Computational Physics* 1997;**133**(1):186–191.
- [9] Divo E, Kassab AJ. ‘Generalized boundary integral equation for heat conduction in non-homogeneous media: recent developments on the sifting property’, *Engineering Analysis with Boundary Elements* 1998;**22**(3):221–234.
- [10] Elman HC, Schultz MH. ‘Preconditioning by fast direct methods for non-selfadjoint nonseparable elliptic equations’, *SIAM Journal on Numerical Analysis* 1986;**23**(1):44–57.
- [11] Englund J, Helsing J. ‘Stress computations on perforated polygonal domains’ *Engineering Analysis with Boundary Elements* 2003;**27**(5):533–546.
- [12] Ethridge F, Greengard L. ‘A new fast-multipole accelerated Poisson solver in two dimensions’, *SIAM Journal on Scientific Computing* 2001;**23**(3):741–760.
- [13] Greengard L, Kropinski MC. ‘An integral equation approach to the incompressible Navier-Stokes equations in two dimensions’, *SIAM Journal on Scientific Computing* 1998;**20**(1):318–336.
- [14] Greengard L, Lee JY. ‘A Direct Adaptive Solver of Arbitrary Order Accuracy’, *Journal of Computational Physics* 1996;**125**(2):415–424.

- [15] Greengard L, Moura M. ‘On the numerical evaluation of electrostatic fields in composite materials’. In *Acta Numerica 1994*. Cambridge University Press: Cambridge, pp. 379–410 (1994).
- [16] Greengard L, Rokhlin V. ‘A fast algorithm for particle simulations’, *Journal of Computational Physics* 1987;**73**(2):325–348.
- [17] Guenther RB, Lee JW. *Partial Differential Equations of Mathematical Physics and Integral Equations*, New York: Dover Publications Inc, (1996).
- [18] Guillard H, Désidéri J-A. ‘Iterative methods with spectral preconditioning for elliptic equations’, *Computer Methods in Applied Mechanics and Engineering* 1990;**80**(1-3):305–312.
- [19] Helsing J. ‘A high-order accurate algorithm for electrostatics of overlapping disks’, *Journal of Statistical Physics* 1998;**90**(5-6):1461–1473.
- [20] Helsing J. ‘Corner singularities for elliptic problems: special basis functions versus ‘brute force’’, *Communications in Numerical Methods in Engineering* 2000;**16**(1):37–46.
- [21] Hrycak T, Rokhlin V. ‘An improved fast multipole algorithm for potential fields’, *SIAM Journal on Scientific Computing* 1998;**19**(6):1804–1826.
- [22] Huang J, Greengard L. ‘A fast direct solver for elliptic partial differential equations on adaptively refined meshes’, *SIAM Journal on Scientific Computing* 2000;**21**(4):1551–1566.
- [23] Huang J, Cheng HW, and Leiterman T. ‘An Adaptive Fast Solver for the Modified Helmholtz Equation in Two Dimensions’, (preprint, Department of Mathematics, University of North Carolina, Chapel Hill).
- [24] Mammoli AA. ‘Solution of non-linear boundary integral equations in complex geometries with auxiliary integral subtraction’, *International Journal for Numerical Methods in Engineering* 2002;**55**(9):1115–1128.
- [25] Mayo A. ‘The rapid evaluation of volume integrals of potential theory on general regions’, *Journal of Computational Physics* 1992;**100**(2):236–245.
- [26] McKenney A, Greengard L, Mayo A. ‘A fast poisson solver for complex geometries’, *Journal of Computational Physics* 1995;**118**(2):348–355.
- [27] Mikhailov SE. ‘Localized boundary-domain integral formulations for problem with variable coefficients’, *Engineering Analysis with Boundary Elements* 2002;**26**(8):681–690.
- [28] Nielsen IMB, Janssen CL. ‘A novel pseudospectral Fourier method for solving Poisson’s equation for a solute in a non-uniform dielectric’, *Computer Physics Communications* 2001;**136**(1-2),29–36.

- [29] Pares-Sierra A, Vallis GK. ‘A fast semi-direct method for the numerical solution of non-separable elliptic equations in irregular domains’, *Journal of Computational Physics* 1989;**82**(2):398–412.
- [30] Pickering WM, Harley PJ. ‘Numerical solution of nonseparable elliptic equations by the iterative application of FFT methods’, *International Journal of Computer Mathematics* 1995;**55**(3+4),211–222.
- [31] Rokhlin V. ‘Rapid solution of integral equations of classical potential theory’, *Journal of Computational Physics* 1985;**60**(2):187–207.
- [32] Rokhlin V. ‘Application of volume integrals to the solution of partial differential equations’, *Computers & Mathematics with Applications* 1985;**11**(7-8):667–679.
- [33] Saad Y, Schultz MH. ‘GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems’, *SIAM Journal on Scientific and Statistical Computing* 1986;**7**(3):856–869.
- [34] Simkin J, Trowbridge E. ‘On the use of the total scalar potential in the numerical solution of field problems in electromagnetics’ *International Journal for Numerical Methods in Engineering* 1979;**14**(3),423–440.
- [35] Strain J. ‘Fast spectrally-accurate solution of variable-coefficient elliptic problems’ *Proceedings of the American Mathematical Society* 1994;**122**(3),843–850.
- [36] User’s Manual FMM Toolbox (2D) for MATLAB Version 1.1 p. 53. Hamden: MadMax Optics Inc., (2002).
- [37] Yarvin N, Rokhlin V. ‘Generalized Gaussian quadratures and singular value decompositions of integral operators’, *SIAM Journal on Scientific Computing* 1998;**20**(2),699–718.
- [38] Zhao S, Yedlin MJ. ‘A new iterative Chebychev spectral method for solving the elliptic equation $\nabla \cdot (\sigma \nabla u) = f$ ’, *Journal of Computational Physics* 1994;**113**(2);215–223.
- [39] Zhu T, Zhang J, Atluri SN. ‘A meshless numerical method based on the local boundary integral equation (LBIE) to solve linear and non-linear boundary value problems’ *Engineering Analysis with Boundary Elements* 1999;**23**(5+6),375–389.