Integral equation methods for elliptic problems with boundary conditions of mixed type[☆]

Johan Helsing

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

Abstract

Laplace's equation with mixed boundary conditions, that is, Dirichlet conditions on parts of the boundary and Neumann conditions on the remaining contiguous parts, is solved on an interior planar domain using an integral equation method. Rapid execution and high accuracy is obtained by combining equations which are of Fredholm's second kind with compact operators on almost the entire boundary with a recursive compressed inverse preconditioning technique. Then an elastic problem with mixed boundary conditions is formulated and solved in an analogous manner and with similar results. This opens up for the rapid and accurate solution of several elliptic problems of mixed type.

Key words: mixed boundary value problem, second kind integral equation, potential theory, elasticity
PACS: 02.30.Rz, 46.15.-x

1. Introduction

The need to solve elliptic problems with different types of boundary conditions on different parts of a connected boundary often arises in computational physics. Elastic specimens partly held fixed and partly subjected to traction [2, 18] and loaded composites with interface cracks [6, 17] are common setups with mixed Dirichlet- and Neumann type conditions. More generally, elliptic problems for multiphase materials where some continuity

---

conditions hold on internal interfaces and loads are applied to a connected outer boundary belong to this class. Grain boundary diffusion in finite-size polycrystals [20, 24] and coupled Stokes and Darcy flow [22] are two examples. Solvers based on integral equations, which are superior for pure boundary conditions, are not always applicable for mixed conditions. When they do apply and the conditions vary on a connected boundary, see [11] for an overview, they are often less advantageous than for pure boundary conditions.

It is hard to find integral equations for mixed problems that are of Fredholm's second kind with operators that are compact on the entire boundary. This is the essential difficulty when boundary conditions vary on contiguous boundary parts [23]. The second-kind-compact-operator property is what makes integral equation methods competitive. This property helps in retaining the condition number of the underlying mathematical problem throughout the solution process.

Using primitive functions of Neumann data, one can sometimes find integral equations for mixed planar problems that are singular with discontinuous coefficients in the sense of Section 116 of [15]. Such equations may require *reduction*, that is, the application of a pseudo-inverse to the dominant operator, for well-posedness. A great advantage with reduction is that it transforms a singular integral equation into one of Fredholm's second kind with operators that are compact on the entire boundary, provided the boundary is smooth. Reduction is certainly efficient when high accuracy is of interest [6, 9]. Drawbacks include that it takes some effort to construct the pseudo-inverse and that it is hard to treat non-smooth boundaries [3] and boundary conditions that change type more than once. Mikhlin uses reduction to derive a second kind equation with compact operators for the mixed problem of the theory of elasticity, see Eq. (17) of Section 72 in [13]. This equation is not written out on explicit form and has, to our knowledge, never been used for numerics.

If one gives up the search for second kind equations with compact operators and is content with discretizing and solving just any integral equation, chiefly for the benefit of dimensionality reduction, the numerical results could suffer. Especially so in the vicinity of *singular boundary points*, that is, points where the boundary conditions change type and where the solution may have a complicated asymptotic behavior. Adaptive mesh refinement close to such points is often not a good idea since it can excite severe ill-conditioning. In general, any attempt at mesh refinement increases ill-conditioning in the absence of the second-kind-compact-operator property.

This paper takes a new approach to mixed boundary conditions. Like

the classic works [13, 15] we strive for integral equations that behave as if they were of Fredholm's second kind with compact operators everywhere. But while the classic works use reduction to achieve this, we use *recursive compressed inverse preconditioning* (a local multilevel technique developed to deal with weaker singularities stemming from boundary irregularities [8]). The advantages with trading reduction for recursive compressed inverse preconditioning are flexibility in modeling and simplicity in programming. Several types of complications can be treated within the same framework.

For brevity we only consider two problems: Laplace's equation in the plane, introduced in Section 3 and used to illustrate general ideas, and planar elasticity, chosen as to let these ideas work in a more challenging setting in Section 9. A key ingredient in the transition from Laplace's equation to elasticity is the particular choice of representation (55) and (56). Sections 2, 5, 6, and 7 discuss quadrature techniques for non-smooth kernels and review recursive compressed inverse preconditioning in the present environment. The computational process is straight-forward, once these issues are settled, and Sections 8 and 10 present very accurate results.

2. Discretization and quadrature

We use Nyström discretization for the integral equations and composite 16-point Gauss-Legendre quadrature as our basic quadrature tool. To keep the notation short we make no distinction between points or vectors in a real plane $\mathbb{R}^2$ and points in a complex plane $\mathbb{C}$. All points will be denoted z or τ. Let Γ be the smooth boundary of a simply connected domain Ω and let Γ be given orientation. Let $\tau(t)$, $t_\alpha < t \leq t_\beta$, be a parameterization of Γ and let there be n_Λ quadrature panels Λ_j, $j = 1, \ldots, n_\Lambda$, of approximately equal length placed on Γ. Then one can easily compute $N = 16n_\Lambda$ nodes t_j and weights w_j, $j = 1, \ldots, N$, associated with integration in t. Let f be a layer density on Γ. The parameterization allows us to view f both as function of position $f(\tau)$ and of parameter $f(t)$. The argument indicates which view is taken in a particular situation. Differentiation with respect to parameter t is indicated with a prime. The abbreviations $\tau_j = \tau(t_j)$, $f_j = f(t_j)$, $\tau'_j = \tau'(t_j)$, and $f'_j = f'(t_j)$ are used.

We shall discretize several integral operators on Γ. If the integral kernel $K(\tau, z)$ and layer density $f(\tau)$ are piecewise smooth, the basic quadrature

$$\int_\Gamma f(\tau) K(\tau, \tau_j) \, \mathrm{d}\tau = \int_{t_\alpha}^{t_\beta} f(t) K(\tau(t), \tau_j) \tau'(t) \, \mathrm{d}t \approx \sum_{k=1}^{N} f_k K(\tau_k, \tau_j) \tau'_k w_k \quad (1)$$

3

should be accurate. If $K(\tau, z)$ is singular for $\tau = z$, special techniques are needed to retain high accuracy. This section reviews such techniques. Note that Nyström discretization of an integral equation means discretization of the integral operators at each quadrature point τ_j, $j = 1, \ldots, N$. The result of the discratization is a linear system with a square system matrix.

2.1. The Cauchy singular operator

We begin with the Cauchy singular integral operator

$$M_{\mathrm{C}} f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_\Gamma \frac{f(\tau)\,\mathrm{d}\tau}{\tau - \tau_j}\,, \quad \tau_j \in \Gamma\,. \tag{2}$$

The integral is to be interpreted in the principal value sense. One option is to use *global regularization*

$$M_{\mathrm{C}} f(\tau_j) = f_j + \frac{1}{\pi \mathrm{i}} \int_\Gamma \frac{(f(\tau) - f_j)\,\mathrm{d}\tau}{\tau - \tau_j}\,, \quad \tau_j \in \Gamma\,. \tag{3}$$

The integral has a continuous integrand when $f(\tau)$ is continuous. It can be discretized with basic quadrature and differentiation of $f(t)$ based on panelwise polynomial interpolation at the Legendre nodes.

A drawback with global regularization is that it may involve a fair amount of row summation for the diagonal elements of the system matrix. A cheaper alternative, in this respect, is *local regularization*

$$M_{\mathrm{C}} f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_{\Gamma^\circ} \frac{f(\tau)\,\mathrm{d}\tau}{\tau - \tau_j} + \frac{f_j}{\pi \mathrm{i}} \log\left(\frac{\tau_b - \tau_j}{\tau_a - \tau_j}\right) + \frac{1}{\pi \mathrm{i}} \int_{\Gamma^\star} \frac{(f(\tau) - f_j)\,\mathrm{d}\tau}{\tau - \tau_j}\,, \tag{4}$$

where $\Gamma^\circ = \Gamma \setminus \Gamma^\star$ and the short boundary part $\Gamma^\star$ contains τ_j. It is assumed that $\Gamma^\star$ starts at $\tau_a = \tau(t_a)$, ends at $\tau_b = \tau(t_b)$, consists of three consecutive panels $\Lambda_1^\star$, $\Lambda_2^\star$, and $\Lambda_3^\star$, say, with τ_j situated on the middle panel $\Lambda_2^\star$. The integrand of the first integral of (4) is smooth when consecutive panels do not differ too much in length. The integrand of the second integral is smooth if $f(\tau)$ is smooth on $\Gamma^\star$. Note that the value of $\log(\cdot)$ in (4), produced by a computer, must be corrected with $\pm\pi\mathrm{i}$ so as to correspond to the principal value.

When $f(\tau)$ has a discontinuity at one end of $\Lambda_2^\star$, neither global- nor local regularization is efficient. Then we suggest *local panelwise evaluation*

$$M_{\mathrm{C}} f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_{\Gamma^\circ} \frac{f(\tau)\,\mathrm{d}\tau}{\tau - \tau_j} + \sum_{l=1}^3 \frac{1}{\pi \mathrm{i}} \int_{\Lambda_l^\star} \frac{f(\tau)\,\mathrm{d}\tau}{\tau - \tau_j}\,, \quad \tau_j \in \Lambda_2^\star\,, \tag{5}$$

4

where the integrand of the first integral is smooth and each term in the sum is computed individually via product integration. We specialize to the case of $\Lambda_l^\star$ being an open arc in the complex plane with starting point -1 and endpoint 1. General arcs can be given this property by translation, rotation, and scaling. 15th-degree accurate product integration weights for the integrals

$$\int_{\Lambda_l^\star} \frac{f(\tau)\, \mathrm{d}\tau}{\tau - \tau_j}\,, \quad \tau_j \in \Lambda_2^\star \tag{6}$$

can then be constructed from the observation that the quantities

$$p_k = \int_{-1}^{1} \frac{\tau^{k-1}\, \mathrm{d}\tau}{\tau - \tau_j}\,, \quad k = 1, \ldots, 16 \tag{7}$$

can be computed from the simple recursion

$$p_{k+1} = \tau_j p_k + c_k\,, \quad k = 1, \ldots, \tag{8}$$

where

$$c_k = \frac{1 - (-1)^k}{k}\,. \tag{9}$$

Once the p_k are available, the product integration weights can be obtained, for example, by solving a 16×16 Vandermonde system. See Section 5 of [7] for details and alternatives.

2.2. The hypersingular operator

Next, we discretize the hypersingular integral operator

$$M_{\mathrm{H}} f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_\Gamma \frac{f(\tau)\, \mathrm{d}\tau}{(\tau - \tau_j)^2}\,, \quad \tau_j \in \Gamma\,, \tag{10}$$

which is to be interpreted in the sense of the Hadamard finite part. The three evaluation techniques of Section 2.1 can in principle be applied here, too. Global regularization, however, becomes clumsy whenever $f(\tau)$ is not everywhere smooth and we shall not use it. Local regularization gives

$$M_{\mathrm{H}} f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_{\Gamma^\circ} \frac{f(\tau)\, \mathrm{d}\tau}{(\tau - \tau_j)^2} - \frac{f_j}{\pi \mathrm{i}} \left(\frac{1}{\tau_b - \tau_j} - \frac{1}{\tau_a - \tau_j} \right)$$

$$+ \frac{f_j'}{\tau_j' \pi \mathrm{i}} \log \left(\frac{\tau_b - \tau_j}{\tau_a - \tau_j} \right) + \frac{1}{\pi \mathrm{i}} \int_{t_a}^{t_b} \frac{\left(f(t) - f_j - f_j'/\tau_j'(\tau(t) - \tau_j) \right)}{(\tau(t) - \tau_j)^2} \tau'(t)\, \mathrm{d}t\,. \tag{11}$$

5

Product integration quadrature weights for local panelwise evaluation

$$M_{\mathrm{H}}f(\tau_j) = \frac{1}{\pi \mathrm{i}} \int_{\Gamma^\circ} \frac{f(\tau)\,\mathrm{d}\tau}{(\tau - \tau_j)^2} + \sum_{l=1}^{3} \frac{1}{\pi \mathrm{i}} \int_{\Lambda_l^\star} \frac{f(\tau)\,\mathrm{d}\tau}{(\tau - \tau_j)^2}\,, \quad \tau_j \in \Lambda_2^\star \qquad (12)$$

can be computed by translation, rotation, and scaling and from the observation that the quantities

$$r_k = \int_{-1}^{1} \frac{\tau^{k-1}\,\mathrm{d}\tau}{(\tau - \tau_j)^2} \quad k = 1, \dots, 16 \qquad (13)$$

can be expressed as

$$r_k = -\left(\frac{1}{1 - \tau_j} + \frac{(-1)^{k-1}}{1 + \tau_j} \right) + (k-1)p_{k-1}\,, \qquad (14)$$

where p_k is as in (7) and $p_0 = 0$.

2.3. The logarithmic operator

Lastly, we discretize the logarithmic integral operator

$$M_{\mathrm{L}}f(\tau_j) = \int_{\Gamma} f(\tau) \log |\tau - \tau_j|\,\mathrm{d}|\tau|\,, \quad \tau_j \in \Gamma\,. \qquad (15)$$

In a composite trapezoidal rule setting, (15) can be discretized using Fourier methods, see Chapter 7.2.2 of [1], or with modified weights, See Section 6 of [12]. In the composite Gauss-Legendre setting we suggest local panelwise evaluation

$$M_{\mathrm{L}}f(\tau_j) = \int_{\Gamma^\circ} f(\tau) \log |\tau - \tau_j|\,\mathrm{d}|\tau|$$

$$+ \sum_{l=1}^{3} \int_{t_{al}}^{t_{bl}} f(t) \log \left| \frac{(t_{bl} - t_{al})}{2} \frac{(\tau(t) - \tau_j)}{(t - t_j)} \right| |\tau'(t)|\,\mathrm{d}t$$

$$+ \sum_{l=1}^{3} \frac{(t_{bl} - t_{al})}{2} \int_{-1}^{1} f(t_l(x))|\tau'(t_l(x))| \log |x - x_{lj}|\,\mathrm{d}x\,, \qquad (16)$$

where t_{al} and t_{bl} are the starting point and endpoint, in parameter, of $\Lambda_l^\star$. Also, in (16) we have made use of the auxiliary parameterizations $t_l(x) = x(t_{bl} - t_{al})/2 + (t_{bl} + t_{al})/2$, so that $x_{lj} = (2t_j - t_{al} - t_{bl})/(t_{bl} - t_{al})$.

The integrands of the first four integrals of (16) are smooth and basic quadrature can be used. The integrands of the three last integrals are not

smooth, but 15-degree accurate product integration quadrature weights can be computed from the observation that the quantities

$$q_k = \int_{-1}^{1} x^{k-1} \log |x - x_j| \, \mathrm{d}x, \quad k = 1, \ldots, 16,\tag{17}$$

can be expressed as

$$q_k = \frac{1}{k} \log |1 - x_j| - \frac{(-1)^k}{k} \log |1 + x_j| - \frac{p_{k+1}}{k},\tag{18}$$

where p_k is as in (7) and (8), but with τ and τ_j replaced with x and x_j.

We remark that when integration in (15) is with respect to $\mathrm{d}\tau$ rather than to $\mathrm{d}|\tau|$, one should replace $|\tau'(t)|$ with $\tau'(t)$ in (16).

3. Integral equations for Laplace's equation

We seek a function $U(z)$, harmonic in Ω, such that

$$\lim_{\Omega \ni \tau \to z} U(\tau) = g_{\mathrm{D}}(z), \quad z \in \Gamma_{\mathrm{D}},\tag{19}$$

$$\lim_{\Omega \ni \tau \to z} n_z \cdot \nabla U(\tau) = g_{\mathrm{N}}(z), \quad z \in \Gamma_{\mathrm{N}},\tag{20}$$

where $g_{\mathrm{D}}(z)$ is Dirichlet data on the boundary part Γ_{D}, $g_{\mathrm{N}}(z)$ is Neumann data on the boundary part Γ_{N}, $\Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}} = \Gamma$, and n_z is the outward unit normal of Γ at z.

Let now $U(z)$, $z \in \Omega \cup \Gamma_{\mathrm{N}}$, be represented by a real density $\rho(z)$, $z \in \Gamma$,

$$U(z) = \frac{1}{\pi} \int_{\Gamma_{\mathrm{D}}} \rho(\tau) \Im \left\{ \frac{\mathrm{d}\tau}{\tau - z} \right\} - \frac{1}{\pi} \int_{\Gamma_{\mathrm{N}}} \rho(\tau) \log |\tau - z| \, \mathrm{d}|\tau|, \quad z \in \Omega \cup \Gamma_{\mathrm{N}}.\tag{21}$$

Insertion of (21) into (19) and (20) gives the system

$$\rho(z) + \frac{1}{\pi} \int_{\Gamma_{\mathrm{D}}} \rho(\tau) \Im \left\{ \frac{\mathrm{d}\tau}{\tau - z} \right\} - \frac{1}{\pi} \int_{\Gamma_{\mathrm{N}}} \rho(\tau) \log |\tau - z| \, \mathrm{d}|\tau| = g_{\mathrm{D}}(z), \quad z \in \Gamma_{\mathrm{D}},\tag{22}$$

$$\rho(z) + \frac{1}{\pi} \int_{\Gamma_{\mathrm{D}}} \rho(\tau) \Im \left\{ \frac{n_z \, \mathrm{d}\tau}{(\tau - z)^2} \right\} + \frac{1}{\pi} \int_{\Gamma_{\mathrm{N}}} \rho(\tau) \Re \left\{ \frac{n_z \, \mathrm{d}|\tau|}{\tau - z} \right\} = g_{\mathrm{N}}(z), \quad z \in \Gamma_{\mathrm{N}}.\tag{23}$$

This system is not of Fredholm's second kind with compact operators on the entire boundary. But away from the singular boundary points it is. More precisely, if the kernels of the integral operators are set to zero when τ and
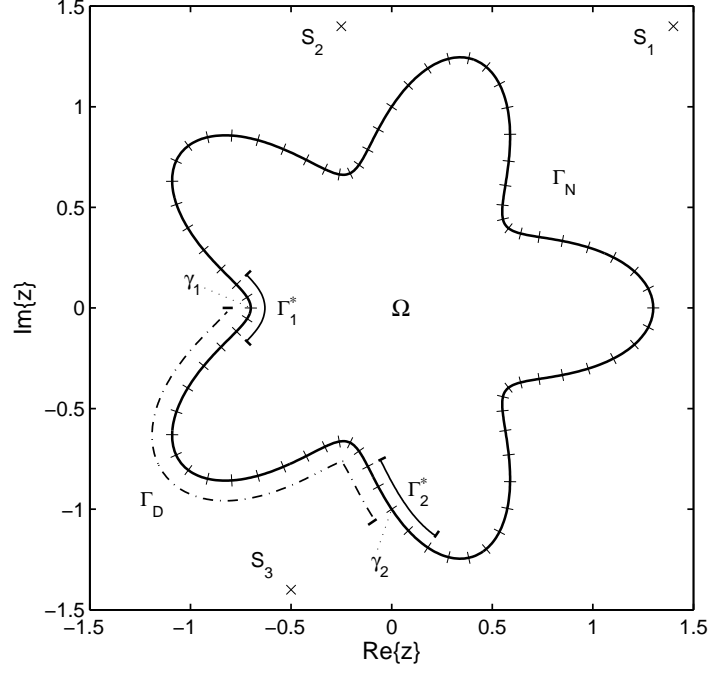
Figure 1: Test domain Ω with boundary $\Gamma = \Gamma_D \cup \Gamma_N$ given by (25) and (26). A coarse mesh of 84 quadrature panels is constructed on Γ. Two parts of the boundary, $\Gamma_1^\star$ and $\Gamma_2^\star$, cover the four coarse panels closest to the singular boundary points γ_1 and γ_2 where the boundary conditions change type. Three sources S_k, for the generation of some boundary conditions, are marked by 'x'.

z simultaneously lie close to a singular boundary point, then the integral operators are compact.

Once the system (22) and (23) is solved for $\rho(z)$, the function $U(z)$ can be computed for $z \in \Omega \cup \Gamma_N$ via (21). The gradient of $U(z)$ can be computed in Ω via differentiation of (21). The normal derivative of $U(z)$ at Γ_D is

$$n_z \cdot \nabla U(z) = \frac{1}{\pi} \int_{\Gamma_D} \rho(\tau) \Im \left\{ \frac{n_z \, d\tau}{(\tau - z)^2} \right\} + \frac{1}{\pi} \int_{\Gamma_N} \rho(\tau) \Re \left\{ \frac{n_z \, d|\tau|}{\tau - z} \right\} , \quad z \in \Gamma_D .$$

$$(24)$$

8

4. Test geometry, hardware, and software

We use the same geometry in all numerical experiments – an interior domain with boundary parameterization

$$\tau(t) = (1 + 0.3\cos 5t)e^{it}, \quad -\pi < t \le \pi, \tag{25}$$

see Fig. 1. This boundary is simple to produce, yet not trivial from a numerical viewpoint since its curvature is varying. The boundary parts Γ_D and Γ_N are

$$\tau(t) \in \Gamma_D, \quad -\pi < t < -\frac{\pi}{2}, \quad \text{and} \quad \tau(t) \in \Gamma_N, \quad -\frac{\pi}{2} < t < \pi. \tag{26}$$

Compressed inverses will be constructed in local coordinate systems centered around the singular boundary points $\gamma_1 = \tau(\pi)$ and $\gamma_2 = \tau(-\pi/2)$. Some reference solutions will be expressed in terms of sources at the points S_k.

All numerical experiments are performed in MATLAB version 7.6 and executed on an ordinary workstation equipped with an IntelCore2 Duo E8400 CPU at 3.00 GHz. Small linear systems are solved with MATLAB's built in solvers. The GMRES iterative solver [19] with a low-threshold stagnation avoiding technique [7] is used for larger systems. System matrices are formed explicitly, but the fast multipole method [4, 5], also implemented in MATLAB and therefore not particularly fast, is used for the evaluation of stress fields in the last example of Section 10.

5. Compressed discretization

The following three sections are about the discretization and solution of the system (22) and (23). The overall goal is stability and computational economy. The plan is to seek a right inverse preconditioner, called $\mathbf{R}$, and apply it analytically to the difficult parts of (22) and (23) – corresponding to interaction close to singular boundary points – and numerically to the remainder. In order to compute $\mathbf{R}$, we need a grid on a mesh that is highly refined close to the singular boundary points. But once $\mathbf{R}$ is obtained we shall solve (22) and (23) on a coarse grid. This section defines $\mathbf{R}$. Section 6 is about the fast construction of $\mathbf{R}$. Section 7 is on the reconstruction of $\rho(z)$ on the fine grid from computed values on the coarse grid. Much of this material is covered in greater detail in Ref. [8].

We discretize the system (22) and (23) on two meshes: a coarse mesh and a fine mesh. The coarse mesh has quadrature panels of approximately equal length and is arranged so that no panel has a singular boundary point

as an interior point, see Fig. 1. The fine mesh is constructed from the coarse mesh by repeated binary subdivision of coarse panels neighboring to singular boundary points. The subdivision is done with respect to length in parameter and in direction towards the singular boundary points. A mesh with n such subdivision is called an n-ply refined mesh. Let K denote the integral operator in (22) and (23). Using Nyström discretization we arrive at two grids and two linear systems

$$(\mathbf{I}_{\text{coa}} + \mathbf{K}_{\text{coa}}) \, \boldsymbol{\rho}_{\text{coa}} = \mathbf{g}_{\text{coa}} \,, \tag{27}$$

$$(\mathbf{I}_{\text{fin}} + \mathbf{K}_{\text{fin}}) \, \boldsymbol{\rho}_{\text{fin}} = \mathbf{g}_{\text{fin}} \,, \tag{28}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{K}$ is the discretization of K, and $\boldsymbol{\rho}$ and $\mathbf{g}$ is the discretized solution and the right hand side, respectively.

Let $K(\tau, z)$ denote the kernel of K. Split $K(\tau, z)$ into two functions

$$K(\tau, z) = K^{\star}(\tau, z) + K^{\circ}(\tau, z) \,, \tag{29}$$

where $K^{\star}(\tau, z)$ is zero except for when τ and z simultaneously lie on a part of Γ covering the four coarse panels closest to a singular boundary point. There $K^{\circ}(\tau, z)$ is zero. There are two boundary parts of this type in Fig. 1, denoted $\Gamma_1^{\star}$ and $\Gamma_2^{\star}$. The kernel split (29) corresponds to an operator split and a matrix split

$$K = K^{\star} + K^{\circ} \qquad \text{and} \qquad \mathbf{K} = \mathbf{K}^{\star} + \mathbf{K}^{\circ} \,. \tag{30}$$

The matrix elements of $\mathbf{K}_{\text{coa}}^{\star}$ will, for example, differ from zero only for two 64×64 blocks.

We point out that the integral operator K° is compact and should be easy to resolve. The operator $K^{\star}$, on the other hand, is not compact and is harder to resolve. It, therefore, seems inefficient to explicitly discretize K° and $K^{\star}$ on the same fine mesh. Indeed, for sufficiently resolved coarse meshes and for sufficiently large $\Gamma_1^{\star}$ and $\Gamma_2^{\star}$ one can express $\mathbf{K}_{\text{fin}}^{\circ}$ in terms of $\mathbf{K}_{\text{coa}}^{\circ}$ as

$$\mathbf{K}_{\text{fin}}^{\circ} = \mathbf{P}\mathbf{K}_{\text{coa}}^{\circ}\mathbf{P}_W^T \,. \tag{31}$$

Here $\mathbf{P}$ is a prolongation operator that performs panelwise 15th-degree polynomial interpolation in parameter from the coarse grid to the fine grid when acting on column vectors from the left. $\mathbf{P}_W$ is a weighted prolongation operator

$$\mathbf{P}_W = \mathbf{W}_{\text{fin}}\mathbf{P}\mathbf{W}_{\text{coa}}^{-1} \,, \tag{32}$$

whose transpose performs panelwise 15th-degree polynomial interpolation in the variable of integration when acting on discretized integral operators

from the right. $\mathbf{W}$ of (32) is a diagonal matrix containing the quadrature weights w_k. One can interpret (31) as follows: the matrix on the left hand side corresponds to an overresolved discretization of a compact integral operator and has low rank. Therefore it could be compressed. A truncated singular value decomposition is one option. The right hand side of (31), which corresponds to prolongation of off-diagonal matrix blocks, is a computationally cheaper alternative. It does not involve any computations at all, apart from the evaluation of matrix elements. We shall assume that the coarse mesh is sufficiently resolved for (31) to hold to high accuracy. We shall also assume that

$$\mathbf{g}_{\mathrm{fin}} = \mathbf{P}\mathbf{g}_{\mathrm{coa}} \tag{33}$$

holds to high accuracy.

Now one can find an equation, essentially on the coarse grid, which has the second-kind-compact-operator property and whose solution has the same discretization error as the solution to (28) on the fine grid. This compressed equation can be expressed in terms of a transformed discrete density $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$ and reads

$$\left(\mathbf{I}_{\mathrm{coa}} + \mathbf{K}_{\mathrm{coa}}^{\circ}\mathbf{R}\right) \tilde{\boldsymbol{\rho}}_{\mathrm{coa}} = \mathbf{g}_{\mathrm{coa}} \,, \tag{34}$$

see Section 6 of [8] for details in its derivation. In (34) only the compressed weighted inverse

$$\mathbf{R} = \mathbf{P}_W^T \left(\mathbf{I}_{\mathrm{fin}} + \mathbf{K}_{\mathrm{fin}}^{\star}\right)^{-1} \mathbf{P} \,, \tag{35}$$

needs the fine grid for its computation. The matrix $\mathbf{R}$ differs from the identity matrix for two diagonal 64×64 blocks. In view of $\mathbf{P}_W^T\mathbf{P} = \mathbf{I}_{\mathrm{coa}}$, which holds since the weights w_k are Gaussian, one may interpret $\mathbf{R}$ as a projection of an inverse right preconditioner to (28) onto a space of piecewise 15th degree polynomials on the coarse mesh.

The relation between the original and the transformed density is

$$\boldsymbol{\rho}_{\mathrm{coa}} = \mathbf{S}\tilde{\boldsymbol{\rho}}_{\mathrm{coa}} \,, \tag{36}$$

where the compressed un-weighted inverse

$$\mathbf{S} = \mathbf{Q} \left(\mathbf{I}_{\mathrm{fin}} + \mathbf{K}_{\mathrm{fin}}^{\star}\right)^{-1} \mathbf{P} \tag{37}$$

also has to be computed on the fine grid. Here $\mathbf{Q}$ is a restriction operator in the opposite direction of $\mathbf{P}$ and $\mathbf{Q}\mathbf{P} = \mathbf{I}_{\mathrm{coa}}$.

Note that $\rho(z)$ is needed in (21) for the evaluation $U(z)$. When z is away from Γ, the kernels of (21) are smooth and can be well resolved by
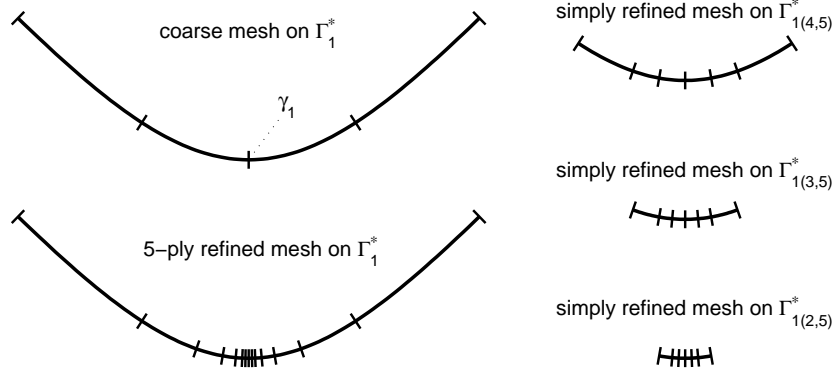
11

Figure 2: Placement of quadrature panels on the boundary part $\Gamma_1^\star$, surrounding the singular boundary point γ_1. Left: meshes on $\Gamma_1^\star$. Right: simply refined meshes on progressively smaller parts of $\Gamma_1^\star$ called $\Gamma_{1(i,n)}^\star$ with $i = 4, 3, 2$ and $n = 5$. See Fig. 1 for the entire coarse mesh.

polynomials. The density $\rho(z)$ is, however, not everywhere smooth. Therefore $\boldsymbol{\rho}_{\mathrm{coa}}$ of (36) is difficult to integrate. Rather, one should use the discrete weight-corrected density

$$\hat{\boldsymbol{\rho}}_{\mathrm{coa}} = \mathbf{R}\tilde{\boldsymbol{\rho}}_{\mathrm{coa}} \tag{38}$$

in the discretization of (21) and $\mathbf{S}$ is then not needed.

When $U(z)$ on Γ_{N} and $n_z \cdot \nabla U(z)$ on Γ_{D} are of interest, one could use a combination of (21) and (24) as post-processor. Now neither the integral kernels nor $\rho(z)$ are everywhere smooth, but accurate values on the coarse grid can still be obtained in terms of $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$. Let $\mathbf{L}\boldsymbol{\rho}$ be the discretization of (21) on Γ_{N} and (24). The quantities sought can then be expressed as

$$\mathbf{Q}\mathbf{L}_{\mathrm{fin}}\boldsymbol{\rho}_{\mathrm{fin}} = \mathbf{L}_{\mathrm{coa}}^\circ \hat{\boldsymbol{\rho}}_{\mathrm{coa}} + \mathbf{X}\tilde{\boldsymbol{\rho}}_{\mathrm{coa}} \,, \tag{39}$$

where $\mathbf{X}$ is the compressed block-diagonal matrix

$$\mathbf{X} = \mathbf{Q}\mathbf{L}_{\mathrm{fin}}^\star \left(\mathbf{I}_{\mathrm{fin}} + \mathbf{K}_{\mathrm{fin}}^\star\right)^{-1} \mathbf{P} \,. \tag{40}$$

6. Recursive construction of R

The definitions of the compressed matrices $\mathbf{R}$ of (35), $\mathbf{S}$ of (37), and $\mathbf{X}$ of (40) are similar. Their direct numerical construction is costly and ill-conditioned. Recursive compression is a better alternative. This multilevel technique, executed in local coordinates, is fast and avoids the increased

12

conditioning that normally follows from progressive refinement close to singular boundary points. This section summarizes and extends results from Section 7 of [8]. Differences between the present paper and [8] are:

- In [8], the singular behavior of $\rho(z)$ is induced by a non-smooth Γ. The integral operators themselves do not change along Γ. The present paper treats the reverse situation. The integral operators vary while Γ is smooth. This may induce stronger singularities.

- The boundary part around a singular boundary point γ_i on which compression takes place is denoted Γ_i in [8]. It typically covers two coarse panels. In the present paper this boundary part is denoted $\Gamma_i^\star$ and covers four coarse panels.

- Eqs. (69) and (70) of [8] are not generally valid and are not used in the present paper.

Although $\mathbf{R}$, $\mathbf{S}$, and $\mathbf{X}$ cannot be computed from one another, one can find useful synergies in their computation. The matrix $\mathbf{R}$ is the easiest to compute. This is so since it involves prolongation. Prolongation operators are easy to factorize and no information is lost in their repeated application on progressively finer grids. This helps in forming a fast recursion. Restriction operators, present in $\mathbf{S}$ and $\mathbf{X}$, are harder to deal with. Information is lost if they are applied repeatedly on progressively coarser grids. Therefore we begin with the construction of $\mathbf{R}$. By-products of this scheme will then be used for the action of $\mathbf{S}$ and $\mathbf{X}$ in Section 7. We specialize to the 64×64 block of $\mathbf{R}$ which describes interaction on $\Gamma_1^\star$. The block which describes interaction on $\Gamma_2^\star$ can be treated analogously.

Consider an n-ply refined mesh on $\Gamma_1^\star$, see the lower left image of Fig. 2. Let $\Gamma_{1(i,n)}^\star$, $1 \leq i \leq n$, denote boundary parts covering the $2(i+2)$ innermost panels of this n-ply refined mesh, see the right images of Fig. 2. Let $\mathcal{G}_{i\mathrm{b}}$ denote a grid on a simply refined mesh on $\Gamma_{1(i,n)}^\star$. Let K, as in Section 5, be the integral operator in (22) and (23). Let the 96×96 matrices $\mathbf{K}_i$ be the discretization of K on $\mathcal{G}_{i\mathrm{b}}$. Let $\mathbf{I}$ be a 96×96 identity matrix. Introduce the splits

$$\mathbf{K}_i = \mathbf{K}_i^\star + \mathbf{K}_i^\circ \qquad \text{and} \qquad \mathbf{I} = \mathbf{I}^\star + \mathbf{I}^\circ \,, \tag{41}$$

where superscript '$\star$' means that matrix elements with both indices in the set $\{17\!:\!80\}$ are carried over from the left hand side, the remaining elements being zero. Superscript '$\circ$' means that elements with at least one index in the set $\{1:16\}$ or $\{81:96\}$ are carried over from the left hand side, the remaining elements being zero.

13

A few more matrices are needed. Let the diagonal 96×96 matrix $\mathbf{W}_1$ contain quadrature weights associated with a grid on a simply refined mesh on $\Gamma_1^\star$. Let the 96×64 matrix $\mathbf{P}_{\mathrm{bc}}$ be a prolongation operator that performs panelwise 15th-degree polynomial interpolation in parameter from a grid on a coarse four-panel mesh to a grid on a simply refined six-panel mesh compatible with mesh refinement on $\Gamma_1^\star$. If the coarse panels on $\Gamma_1^\star$ are equisized in parameter this means that $\mathbf{P}_{\mathrm{bc}}$ prolongs from a mesh on $[-2, 2]$ with breakpoints $\{-2, -1, 0, 1, 2\}$ to a mesh on that same interval with breakpoints $\{-2, -1, -0.5, 0, 0.5, 1, 2\}$. Let the diagonal 64×64 matrix $\mathbf{W}_0$ contain the quadrature weights of the coarse grid on $\Gamma_1^\star$. Let

$$\mathbf{P}_{W\mathrm{bc}} = \mathbf{W}_1 \mathbf{P}_{\mathrm{bc}} \mathbf{W}_0^{-1}. \tag{42}$$

Now the 64×64 block of $\mathbf{R}$ which describes interaction on $\Gamma_1^\star$ can be expressed as $\mathbf{R}_n$ and computed from the simple recursion

$$\mathbf{R}_i = \mathbf{P}_{W\mathrm{bc}}^T \left(\mathbb{F}\{\mathbf{R}_{i-1}^{-1}\} + \mathbf{I}^\circ + \mathbf{K}_i^\circ \right)^{-1} \mathbf{P}_{\mathrm{bc}}, \quad i = 1, \dots, n, \tag{43}$$

where

$$\mathbb{F}\{\mathbf{R}_0^{-1}\} = \mathbf{I}^\star + \mathbf{K}_1^\star \tag{44}$$

is used for initialization. Here $\mathbb{F}\{\cdot\}$ is an operator which creates a frame of width 16 of zeros around its matrix argument. Note that the recursion (43) starts at $i = 1$ and that n steps give a $\mathbf{R}$ which, when used in (34), produces a solution with the same discretization error as the solution of (28) on an n-ply refined mesh. This will be further illustrated in the numerical examples of Sections 8 and 10, where we repeatedly compare (28) to (34) and see how they converge identically with n until (28) suddenly becomes unstable. Besides stability, the other advantage with (34) and (43) is computational economy. Each step in (43) amounts to setting up the 96×96 matrix $\mathbf{K}_i^\circ$, inverting the 64×64 matrix $\mathbf{R}_{i-1}$ available from the previous recursion step, solving a linear system with $\mathbf{P}_{\mathrm{bc}}$ as right hand side, and multiplying with $\mathbf{P}_{W\mathrm{bc}}^T$. The preconditioned system (34) has relatively few unknowns and converges rapidly in GMRES thanks to a clustered spectrum.

The condition number of $\mathbf{R}_i$ in (43) could be very high for $i \ll n$. This ill-conditioning is chiefly due to bad scaling and may be harmful to matrix inversion. It can be cured as follows: $\mathbf{R}_i$ has properties similar to $\mathbf{I} + \mathbf{K}_i$. This 96×96 matrix can be partitioned into four 48×48 blocks. The two diagonal blocks are dominated by elements of unit magnitude. The two off-diagonal blocks, which stem from hypersingular or logarithmic interaction between points on $\Gamma_{1(i,n)}^\star$, are dominated by elements that vary greatly with

recursion step i. The hypersingular block has elements proportional to $1/s_i$, where s_i is the length of $\Gamma^\star_{1(i,n)}$. The logarithmic block has elements proportional to $s_i \log(s_i)$. Therefore, to better balance $\mathbf{R}_i$, a diagonal similarity transformation is carried out prior to its inversion and afterwards carried out in the opposite direction. The transformation matrix has diagonal entries unity and $1/s_i$. A similar balancing is done with the matrix within parenthesis in (43). Despite this balancing, the condition number of $\mathbf{R}_i$ could be as high as 10^7. Fortunately, $\mathbf{R}$ is not used as system matrix in the compressed equation (34). It is only used in composition with $\mathbf{K}^\circ_{\mathrm{coa}}$ as a perturbation of the identity. The condition number of the system matrix in (34) is around 600 in our examples.

Remark: $\Gamma^\star_1$ contains four coarse panels. Simply refined meshes on $\Gamma^\star_{1(i,n)}$ contain six panels. These choices were made for two reasons. First, a four-panel $\Gamma^\star_1$ makes (31) hold to high accuracy also for Γ with more complicated behavior close to γ_1. Second, the derivation of (43) relies on a decomposition similar to (31), but involving $\mathbf{K}^\circ_i$. For that decomposition to hold to high accuracy when $K(\tau, z)^\circ$ comes from a singular operator, $\mathbf{K}^\circ_i$ must refer to a discretization on a mesh that is at least simply refined with respect to a uniform four-panel mesh. In Ref. [8], the underlying kernels were smoother and we worked with smaller meshes in the recursion.

7. Recursive reconstruction of ρ_{fin} and the action of X

After $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$ has been obtained from (34) one may wish to reconstruct $\boldsymbol{\rho}_{\mathrm{fin}}$ of (28). This can be achieved by, loosely speaking, running (43) backwards. Let $\tilde{\boldsymbol{\rho}}_{\mathrm{coa},n}$ be the restriction of $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$ to $\Gamma^\star_1$ and define

$$\tilde{\boldsymbol{\rho}}_{\mathrm{fin},n} = \mathbf{P}_n \tilde{\boldsymbol{\rho}}_{\mathrm{coa},n} \,, \tag{45}$$

where $\mathbf{P}_n$ is the restriction of $\mathbf{P}$ of Section 5 to $\Gamma^\star_1$. Then define

$$\begin{aligned}
\tilde{\boldsymbol{\rho}}_{\mathrm{fin},i-1} &= \left(\mathbf{I} + \mathbf{K}^\bullet_{i-1}\right)\left(\mathbf{I} + \mathbf{K}^\bullet_i\right)^{-1} \tilde{\boldsymbol{\rho}}_{\mathrm{fin},i} \\
&= \left[\mathbf{I} - \mathbf{K}^{\bullet\diamond}_i\left(\mathbf{I} + \mathbf{K}^\bullet_i\right)^{-1}\right] \tilde{\boldsymbol{\rho}}_{\mathrm{fin},i}, \quad i = n, \ldots, 1, \tag{46}
\end{aligned}$$

where all matrices have dimension $32(n+2) \times 32(n+2)$, $\mathbf{I}$ is the identity, $\mathbf{K}^\bullet_n$ is the restriction of $\mathbf{K}_{\mathrm{fin}}$ to $\Gamma^\star_1$, $\mathbf{K}^\bullet_i$ is $\mathbf{K}^\bullet_n$ with elements in a frame of width $16(n-i)$ set to zero, and $\mathbf{K}^{\bullet\diamond}_i = \mathbf{K}^\bullet_i - \mathbf{K}^\bullet_{i-1}$. The first and last $16(n-i)$ elements of the column vector $\tilde{\boldsymbol{\rho}}_{\mathrm{fin},i}$ coincide with values of $\boldsymbol{\rho}_{\mathrm{fin}}$ on $\Gamma^\star_1$. The remaining elements of $\tilde{\boldsymbol{\rho}}_{\mathrm{fin},i}$ correspond to a transformed discrete density

associated with a weighted inverse computed on an $(n-i)$-ply refined mesh on $\Gamma_{1(i,n)}^{\star}$, rather than on an n-ply refined mesh on $\Gamma_1^{\star}$.

Now (46) can be compressed using quantities from Section 6

$$\vec{\rho}_{\mathrm{coa},i} = \left[\mathbf{I} - \mathbf{K}_i^{\circ}\left(\mathbb{F}\{\mathbf{R}_{i-1}^{-1}\} + \mathbf{I}^{\circ} + \mathbf{K}_i^{\circ}\right)^{-1}\right]\mathbf{P}_{\mathrm{bc}}\tilde{\rho}_{\mathrm{coa},i}, \quad i = n, \ldots, 1, \quad (47)$$

where the 64 elements $\tilde{\rho}_{\mathrm{coa},i}$ are taken as elements $\{17:80\}$ of $\vec{\rho}_{\mathrm{coa},i+1}$ for $i < n$. The 32 elements $\{1\!:\!16\}$ and $\{81\!:\!96\}$ of $\vec{\rho}_{\mathrm{coa},i}$ are the reconstructed values of ρ_{fin} on the outermost panels of a simply refined mesh on $\Gamma_{1(i,n)}^{\star}$. The 64 elements $\{17\!:\!80\}$ of $\vec{\rho}_{\mathrm{coa},i}$ are the discrete values of a transformed density associated with a compressed weighted inverse computed on $\Gamma_{1(i,n)}^{\star}$. After the recursion is completed, the reconstructed values on the four innermost panels are obtained from

$$\mathbf{R}_0\tilde{\rho}_{\mathrm{coa},0}. \quad (48)$$

Should one wish to obtain the values of ρ_{coa}, that is, the action of $\mathbf{S}$ on $\tilde{\rho}_{\mathrm{coa}}$, see (36), one can simply apply $\mathbf{Q}$ of Section 5 to ρ_{fin}.

The action of $\mathbf{X}$ on $\tilde{\rho}_{\mathrm{coa}}$ in (39) can be computed via restriction and addition of the vectors

$$\mathbf{L}_i^{\circ}\left(\mathbf{I}^{\circ} + \mathbb{F}\{\mathbf{R}_{i-1}\}\right)\vec{\rho}_{\mathrm{coa},i}, \quad i = n, \ldots, 1. \quad (49)$$

8. Numerical examples for Laplace's equation

This section demonstrates the performance of n-step recursive compressed inverse preconditioning on a coarse grid (34) with (35) and (43). The goal is to compute $U(z)$ of (21) and $n_z \cdot \nabla U(z)$ of (24). The geometry and the mesh is that of Fig. 1, which means that (34) has 1344 unknowns. The underlying integral equation is the system (22) and (23). The hypersingular integral operator in (23) is discretized with local panelwise evaluation (12) for z on panels neighboring to γ_1 and γ_2 and with local regularization (11) otherwise. For comparison we also test non-compressed discretization on an n-ply refined mesh (28), which has $1344 + 64n$ unknowns. One could, perhaps, regard (28) as an instantiation of a standard integral equation scheme; in many ways typical of those used for mixed problems in computational materials science.

Two examples of boundary conditions $g_{\mathrm{D}}(z)$ and $g_{\mathrm{N}}(z)$ are used. The first example is constructed from a closed form solution

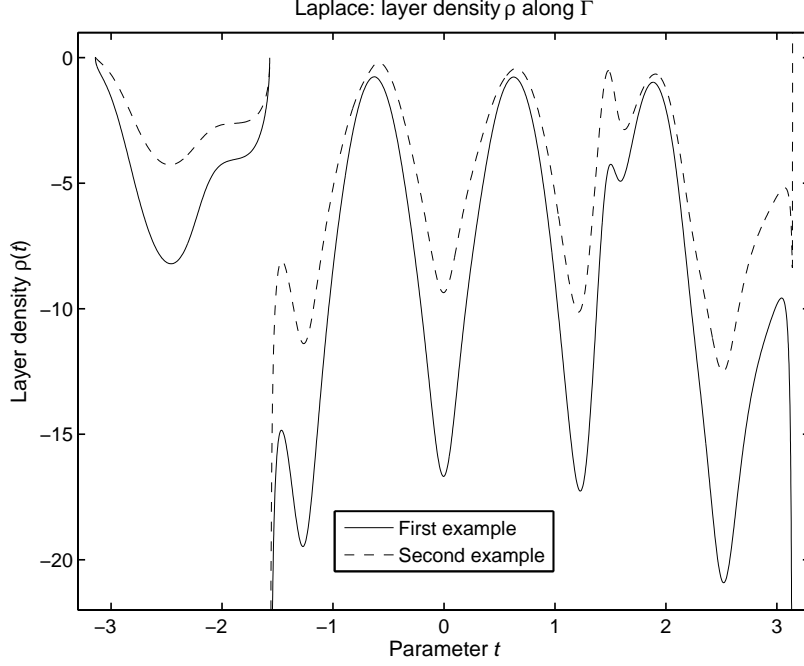$$U(z) = \Re\left\{\sum_{k=1}^{3}\frac{1}{z - S_k}\right\}, \quad (50)$$

16

Figure 3: Solution $\rho(t)$ to the system (22) and (23). The first example of boundary conditions correspond to a smooth solution $U(z)$ known in closed form. The second example of conditions corresponds to a solution where $n_z \cdot \nabla U(z)$ diverges on Γ_{D}. See Section 4 for a description of the geometry.

where $S_1 = 1.4 + 1.4\mathrm{i}$, $S_2 = -0.25 + 1.4\mathrm{i}$, and $S_3 = -0.5 - 1.4\mathrm{i}$ are sources outside of Ω, see Fig. 1. Obviously, these boundary conditions give a smooth $U(z)$. The second example is constructed by keeping $g_{\mathrm{N}}(z)$ compatible with (50) while setting $g_{\mathrm{D}}(z) = 0$. Then there is no closed form solution and $U(z)$ is not smooth. In fact, $n_z \cdot \nabla U(z)$ diverges on Γ_{D} close to γ_1 and γ_2. Fig. 3 depicts the density $\rho(t)$ along Γ for the two examples. Despite the differences in $U(z)$, the two $\rho(t)$ are similar. They both have $\sqrt{s}$ singularities on Γ_{D} and $1/\sqrt{s}$ singularities on Γ_{N}, where s is the distance to the closest singular boundary point. A difference is that the asymptotic regimes are much more narrow for the second example than for the first. Particularly so close to $t = \pm\pi$.

We compute $U(z)$ at 1000 evenly distributed points z in Ω, not closer than 1.5 panel lengths to Γ. Fig. 4 shows convergence in Euclidean norm. There are major differences between the solution schemes. Non-compressed
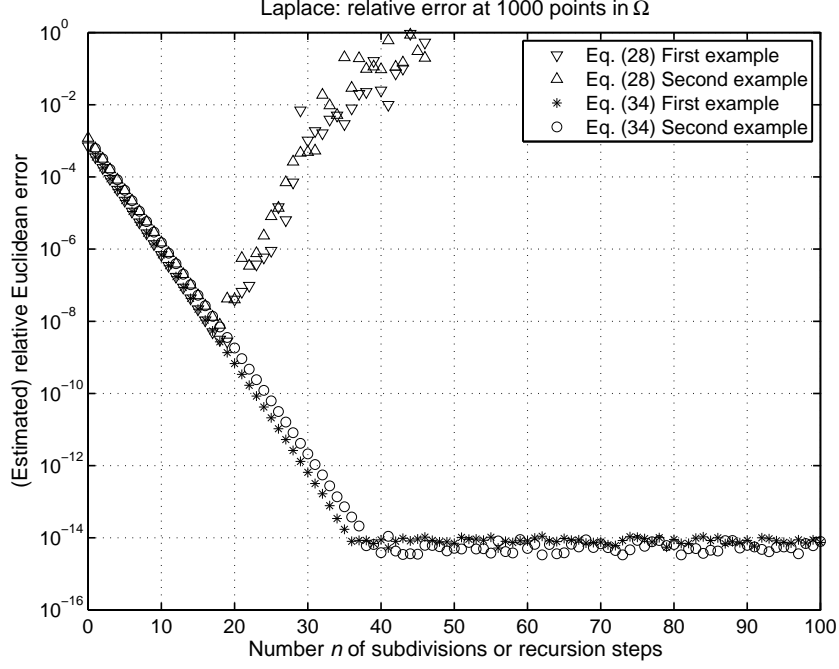
17

Figure 4: Convergence for $U(z)$ of (21) at 1000 points in Ω. The density $\rho(\tau)$ is obtained from (22) and (23) in two ways: via discretization on an n-ply refined mesh (28) and via recursive compressed inverse preconditioning (34) with n steps in (43). Two examples of boundary conditions are tested. Reference values are obtained analytically for the first example and via averaging of a large number of runs with overresolved grids for the second.

discretization (28) is unstable and produces converging results only up to a 20-ply refined mesh. Recursive compressed inverse preconditioning (34) is stable. Results accurate to $100\epsilon_{\mathrm{mach}}$ are produced for any number of recursion steps beyond 40. The differences between the two examples of boundary conditions are minor. This is so since neither scheme is derived under any explicit smoothness assumptions on $U(z)$. It is the resolution of $\rho(t)$, see Fig. 3, or its transformed counterparts, that controls convergence. Recursive compressed inverse preconditioning is derived under the assumption that the right hand side of the integral equation is piecewise smooth, which holds for both examples of boundary conditions.

Not only is (34) superior when it comes to convergence in $U(z)$, $z \in \Omega$. It also does a better job than (28) in terms of convergence in the GMRES solver. Fig. 5 shows that the number of iterations with (34) is constant and
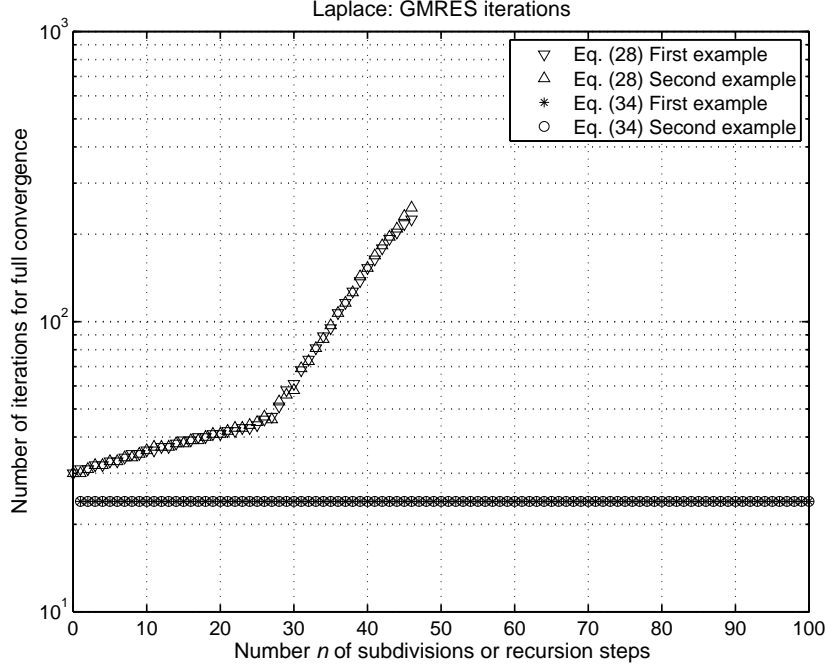
18

Figure 5: The number of GMRES iterations needed to reach a stopping criterion threshold of ϵ_{mach} in the relative residual when solving (28) and (34) as in Fig. 4.

independent of n. With (28) it grows with refinement. The computational cost for (34) is evenly distributed between the recursion (43) and the main system (34). It takes around 0.15 seconds to solve (34). With $n = 50$ steps in (43), the time spent on constructing $\mathbf{R}$ is around 0.25 seconds.

Now we compute $U(z)$ on Γ_{N} and $n_z \cdot \nabla U(z)$ on Γ_{D}. Recursive compressed inverse preconditioning (34) needs the post-processor (39), where the action of the compressed matrix $\mathbf{X}$ on $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$ is computed as outlined in Section 7. Fig. 6 illustrates convergence for the first example of boundary conditions. Comparison with Fig. 4 shows that the Neumann–Dirichlet map on Γ_{N} is as accurate and rapidly converging as the values of $U(z)$ at interior points. The Dirichlet–Neumann map on Γ_{D} is less well conditioned. Nevertheless, (34) outperforms (28) on both parts of the boundary. Non-compressed discretization (28) does, in fact, not converge at all in L^2-norm for the Dirichlet–Neumann map. The errors close to the singular boundary points diverge. That is why the L^1 norm is used in Fig. 6.
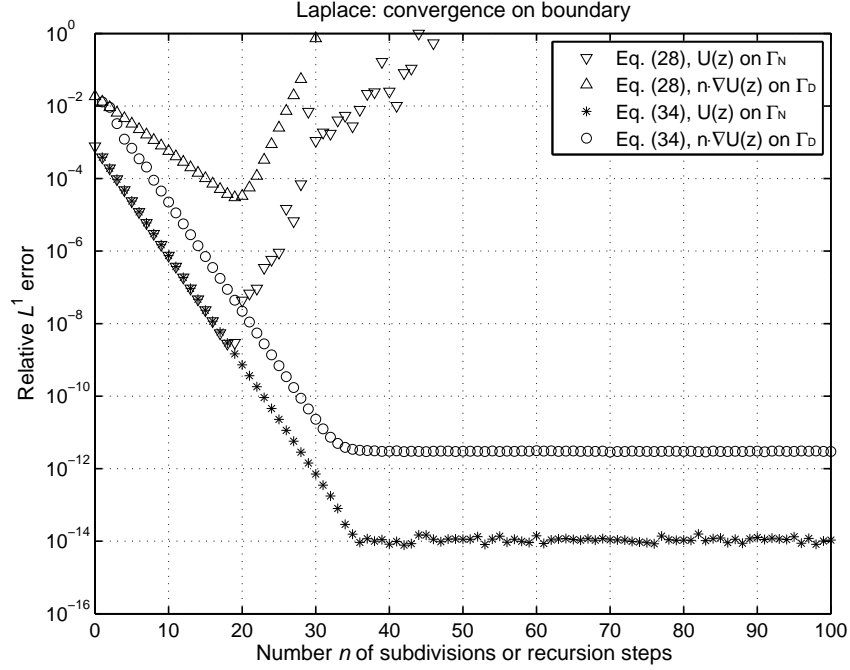
19

Figure 6: Convergence for $U(z)$ on Γ_{N} and for $n_z \cdot \nabla U(z)$ on Γ_{D}. The first example of boundary conditions, compatible with (50), is used.

9. Integral equations for planar elasticity

Let the domain Ω now consist of an elastic medium with shear modulus μ and a material parameter $\varkappa$ which under plane strain conditions and with Poisson's ratio ν is

$$\varkappa = 3 - 4\nu \geq 1\,, \tag{51}$$

see Section 32 of [14]. Displacement $\mathcal{U}(z) = (u_x, u_y)$ is prescribed along Γ_{D} and traction $\mathcal{T}(z) = (t_x, t_y)$ is prescribed along Γ_{N}, that is, $g_{\mathrm{D}}(z) = \mathcal{U}^{\mathrm{pr}}(z)$ and $g_{\mathrm{N}}(z) = \mathcal{T}^{\mathrm{pr}}(z)$. We want to determine the elastic fields inside Ω. The governing partial differential equation for this problem can formulated in several ways, see Section 39 of [13] or Section 2 of [5].

Efficient algorithms for planar elasticity can often be constructed via two analytic functions $\phi(z)$ and $\psi(z)$, called Goursat functions. These are simply related to displacement, stress, and traction. See Section 40 of [13].

20

In particular, on the boundary holds

$$\lim_{\Omega \ni \tau \to z} \frac{1}{2\mu} \left(\varkappa \phi(\tau) - \tau \overline{\phi'(\tau)} - \overline{\psi(\tau)} \right) = g_{\mathrm{D}}(z), \quad z \in \Gamma_{\mathrm{D}}, \tag{52}$$

$$\lim_{\Omega \ni \tau \to z} 2n_z \Re\{\Phi(\tau)\} - \bar{n}_z \left(z \overline{\Phi'(\tau)} + \overline{\Psi(\tau)} \right) = g_{\mathrm{N}}(z), \quad z \in \Gamma_{\mathrm{N}}, \tag{53}$$

where bar means complex conjugation, $\Phi(z) = \phi'(z)$, and $\Psi(z) = \psi'(z)$. In general, boundary integral equations can be obtained by first choosing integral representations for $\phi(z)$ and $\psi(z)$ and then enforcing the boundary conditions. If the representations are cleverly made, the resulting integral equation will be of Fredholm's second kind with compact operators or singular with a dominating non-singular part, that is, of the type

$$\omega(z) + \frac{B}{\pi \mathrm{i}} \int_\Gamma \frac{\omega(\tau)\,\mathrm{d}\tau}{\tau - z} + \frac{1}{\pi \mathrm{i}} \int_\Gamma \omega(\tau) K(\tau, z)\,\mathrm{d}\tau = g(z), \quad |B| < 1, \quad z \in \Gamma, \tag{54}$$

where $\omega(z)$ is an unknown complex layer density and the second integral operator is compact. Note that if $|B| > 1$, the Cauchy singular operator is dominating, the solution to (54) is not unique, additional constraints have to be added, and reduction is necessary for the integral equation to be numerically tractable, see Section 109 of [15]. Section 56 of [13] and [21] give examples of successful representations of $\phi(z)$ and $\psi(z)$ for some planar elastic problems with pure boundary conditions.

Natroshvili and Zazashvili recently suggested a new representation of the Goursat functions for a mixed boundary value problem on an elastic composite with interior cracks [16], that is, a generalization of our setup. The purpose of their work is to prove uniqueness of the solution. Here we suggest the following representation for the mixed problem of elasticity, better suited for numerics

$$\phi(z) = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{D}}} \frac{\omega(\tau)\,\mathrm{d}\tau}{\tau - z} - \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{N}}} \omega(\tau) \log(\tau - z)\,\mathrm{d}\tau, \quad z \in \Omega, \tag{55}$$

$$\psi(z) = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{D}}} \frac{\omega(\tau)\,\mathrm{d}\bar{\tau}}{\tau - z} - \frac{\varkappa}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{D}}} \frac{\overline{\omega(\tau)}\,\mathrm{d}\tau}{\tau - z} - \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{D}}} \frac{\omega(\tau)\bar{\tau}\,\mathrm{d}\tau}{(\tau - z)^2}$$

$$-\frac{\varkappa}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{N}}} \overline{\omega(\tau)} \log(\tau - z)\,\mathrm{d}\bar{\tau} - \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_{\mathrm{N}}} \frac{\omega(\tau)\bar{\tau}\,\mathrm{d}\tau}{\tau - z}, \quad z \in \Omega. \tag{56}$$

Insertion of (55) and (56) into (52) and (53) gives the system

$$\omega(z) + \frac{1}{\pi} \int_{\Gamma_{\mathrm{D}}} \omega(\tau) \Im\left\{ \frac{\mathrm{d}\tau}{\tau - z} \right\} + \frac{1}{2\varkappa \pi \mathrm{i}} \left[\int_{\Gamma_{\mathrm{D}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\tau}{\bar{\tau} - \bar{z}} - \frac{(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right) \right.$$

$$-\int_{\Gamma_{\mathrm{N}}} \frac{\overline{\omega(\tau)}(\tau - z)\,\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} \Bigg] - \frac{1}{\pi\mathrm{i}}\int_{\Gamma_{\mathrm{N}}} \omega(\tau) \log|\tau - z|\,\mathrm{d}\tau = \frac{2\mu}{\varkappa} g_{\mathrm{D}}(z)\,, \quad z \in \Gamma_{\mathrm{D}}\,,$$

$$(57)$$

$$\omega(z) + \frac{1}{(\varkappa + 1)\pi\mathrm{i}} \Bigg[\int_{\Gamma_{\mathrm{D}}} \omega(\tau) \left(\frac{\mathrm{d}\tau}{(\tau - z)^2} - \varkappa \frac{\bar{n}_z}{n_z} \frac{\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right)$$

$$- \int_{\Gamma_{\mathrm{D}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} - \frac{\bar{n}_z}{n_z} \frac{\mathrm{d}\tau}{(\bar{\tau} - \bar{z})^2} + \frac{\bar{n}_z}{n_z} \frac{2(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^3} \right)$$

$$+ (\varkappa - 1)\frac{\bar{n}_z}{n_z} \int_{\Gamma_{\mathrm{N}}} \frac{\omega(\tau)\,\mathrm{d}\tau}{\bar{\tau} - \bar{z}} + \int_{\Gamma_{\mathrm{N}}} \omega(\tau) \left(\frac{\mathrm{d}\tau}{\tau - z} + \frac{\bar{n}_z}{n_z}\frac{\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} \right)$$

$$- \int_{\Gamma_{\mathrm{N}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} + \frac{\bar{n}_z}{n_z}\frac{(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right) \Bigg] = \frac{2}{\varkappa + 1}\bar{n}_z g_{\mathrm{N}}(z)\,, \quad z \in \Gamma_{\mathrm{N}}\,. \quad (58)$$

This system may look complicated at first glance, but away from γ_1 and γ_2 it is of the type (54) with $B = 0$ for $z \in \Gamma_{\mathrm{D}}$ and with $|B| = (\varkappa - 1)/(\varkappa + 1) < 1$ for $z \in \Gamma_{\mathrm{N}}$; integral operators containing kernels within large parenthesis are either smooth or not evaluated at points where they would be singular. The logarithm in (57) is real valued. One does not have to worry about branches.

Once the system (57) and (58) is solved for $\omega(z)$, the displacement can be computed via

$$\mathcal{U}(z) = \frac{\varkappa}{2\mu\pi} \int_{\Gamma_{\mathrm{D}}} \omega(\tau) \Im\left\{ \frac{\mathrm{d}\tau}{\tau - z} \right\} + \frac{1}{4\mu\pi\mathrm{i}} \int_{\Gamma_{\mathrm{D}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} - \frac{(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right)$$

$$- \frac{1}{4\mu\pi\mathrm{i}} \int_{\Gamma_{\mathrm{N}}} \frac{\overline{\omega(\tau)}(\tau - z)\,\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} - \frac{\varkappa}{2\mu\pi\mathrm{i}} \int_{\Gamma_{\mathrm{N}}} \omega(\tau) \log|\tau - z|\,\mathrm{d}\tau\,, \quad z \in \Omega \cup \Gamma_{\mathrm{N}}\,.$$

$$(59)$$

The traction on Γ_{D} is

$$\mathcal{T}(z) = (1 - \varkappa)\frac{n_z}{2}\frac{\mathrm{d}\omega(z)}{\mathrm{d}z} + \frac{n_z}{2\pi\mathrm{i}} \Bigg[\int_{\Gamma_{\mathrm{D}}} \omega(\tau) \left(\frac{\mathrm{d}\tau}{(\tau - z)^2} - \varkappa \frac{\bar{n}_z}{n_z}\frac{\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right)$$

$$- \int_{\Gamma_{\mathrm{D}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} - \frac{\bar{n}_z}{n_z}\frac{\mathrm{d}\tau}{(\bar{\tau} - \bar{z})^2} + \frac{\bar{n}_z}{n_z}\frac{2(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^3} \right)$$

$$+ (\varkappa - 1)\frac{\bar{n}_z}{n_z} \int_{\Gamma_{\mathrm{N}}} \frac{\omega(\tau)\,\mathrm{d}\tau}{\bar{\tau} - \bar{z}} + \int_{\Gamma_{\mathrm{N}}} \omega(\tau) \left(\frac{\mathrm{d}\tau}{\tau - z} + \frac{\bar{n}_z}{n_z}\frac{\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} \right)$$

$$- \int_{\Gamma_{\mathrm{N}}} \overline{\omega(\tau)} \left(\frac{\mathrm{d}\bar{\tau}}{\bar{\tau} - \bar{z}} + \frac{\bar{n}_z}{n_z}\frac{(\tau - z)\,\mathrm{d}\bar{\tau}}{(\bar{\tau} - \bar{z})^2} \right) \Bigg]\,, \quad z \in \Gamma_{\mathrm{D}}\,, \quad (60)$$

where the differentiation in the first term is taken along Γ.

At this point we wish to emphasize the strong analogies in structure between planar elasticity and Laplace's equation in our presentation. The boundary conditions (52) and (53) correspond to (19) and (20). The representations (55) and (56) correspond to (21) with $z \in \Omega$. The system (57) and (58) correspond to (22) and (23). The expressions (59) and (60) correspond to (21) and (24). We could continue to write out analogous equations corresponding to all equations of Sections 5, 6, and 7, but refrain from doing so since the process is straight-forward from now on.

10. Numerical examples for planar elasticity

This section parallels Section 8, but with Laplace's equation replaced with the equations of planar elasticity. When the setups and the numerical results are similar to those of Section 8, we shall be brief.

Integral operators with Cauchy singular kernels are discretized with local panelwise evaluation (5) for z on panels neighboring to γ_1 and γ_2, with local regularization (4) for z on other panels in $\Gamma_1^\star$ and $\Gamma_2^\star$, and with global regularization (3) otherwise. Integral operators with hypersingular kernels are treated as in Section 8. We take advantage of the fact that $(\tau - z)/(\bar{\tau} - \bar{z})$ is smooth on smooth boundaries. Non-smooth boundaries would, in addition, require quadrature for the kernel $1/(\tau - z)^3$. A first example of boundary conditions $g_{\mathrm{D}}(z)$ and $g_{\mathrm{N}}(z)$ comes from a closed form solution

$$\phi(z) = \sum_{k=1}^{3} \frac{1}{z - S_k} \quad \text{and} \quad \psi(z) = z^2 \,, \tag{61}$$

where the S_k are as in (50). A second example is constructed by keeping $g_{\mathrm{N}}(z)$ compatible with (61) while setting $g_{\mathrm{D}}(z) = 0$. Elastic parameters are chosen as $\mu = 0.5$ and $\varkappa = 2.5$.

Fig. 7 shows convergence for the displacement $\mathcal{U}(z)$ at 1000 points $z \in \Omega$, placed as in Section 8. The convergence is very similar to that of $U(z)$ for Laplace's equation in Fig. 4, although one more digit is lost for the second example of boundary conditions. The number of GMRES iterations, not shown, exhibits the same dependence on the number n of subdivisions or recursion steps as in Fig. 5, but is twice as large for a given n.

Fig. 8, analogous to Fig. 6, shows convergence for the Neumann-Dirichlet map on Γ_{N} and the Dirichlet–Neumann map on Γ_{D}. Again, recursive compressed inverse preconditioning on a coarse grid (34) outperforms the standard scheme (28) in virtually every respect.
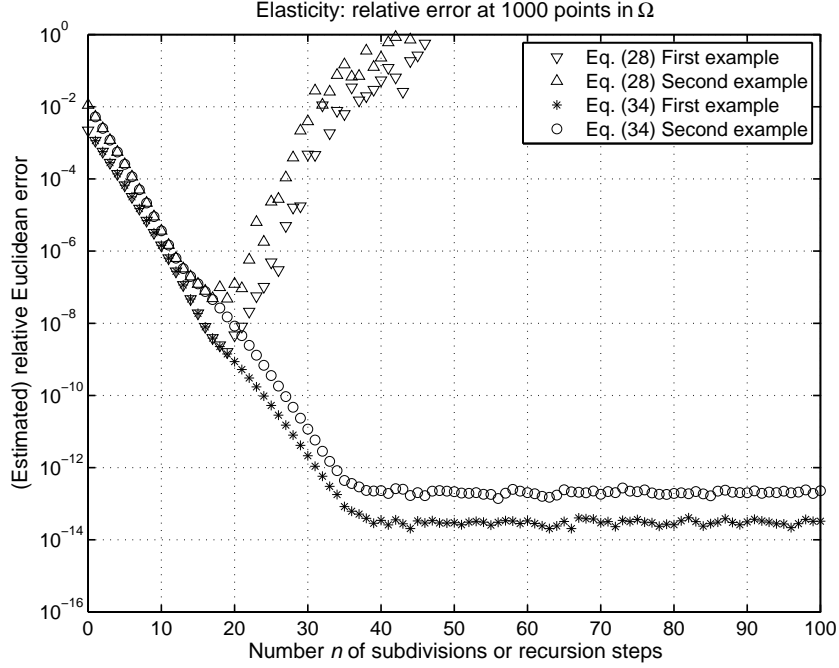
Figure 7: Convergence for $\mathcal{U}(z)$ of (59) at 1000 points in Ω. The density $\omega(\tau)$ is obtained from (57) and (58) in two ways: via discretization on an n-ply refined mesh and via n-step recursive compressed inverse preconditioning. Two examples of boundary conditions are tested, see (61) and compare Fig. 4.

Strictly speaking, a boundary value problem is not properly solved until the solution can be accurately evaluated in the *entire* computational domain. So far we have presented convergence of solutions on Γ and in Ω but not too close to Γ. Accurate evaluations very close to, but off, Γ are more demanding programming wise. With a little effort, however, excellent results can be obtained in several ways [7]. As an example, let us compute the trace of the stress tensor

$$\sigma_{xx} + \sigma_{yy} = 4\Re\{\Phi(z)\}, \quad z \in \Omega. \tag{62}$$

This problem is sufficiently simple for *globally compensated quadrature* to apply, which is the least intricate of the methods discussed in [7], see also [10]. From (55) follows

$$\Phi(z) = \frac{1}{2\pi\mathrm{i}} \int_\Gamma \frac{\theta(\tau)\,\mathrm{d}\tau}{\tau - z}, \quad z \in \Omega, \tag{63}$$
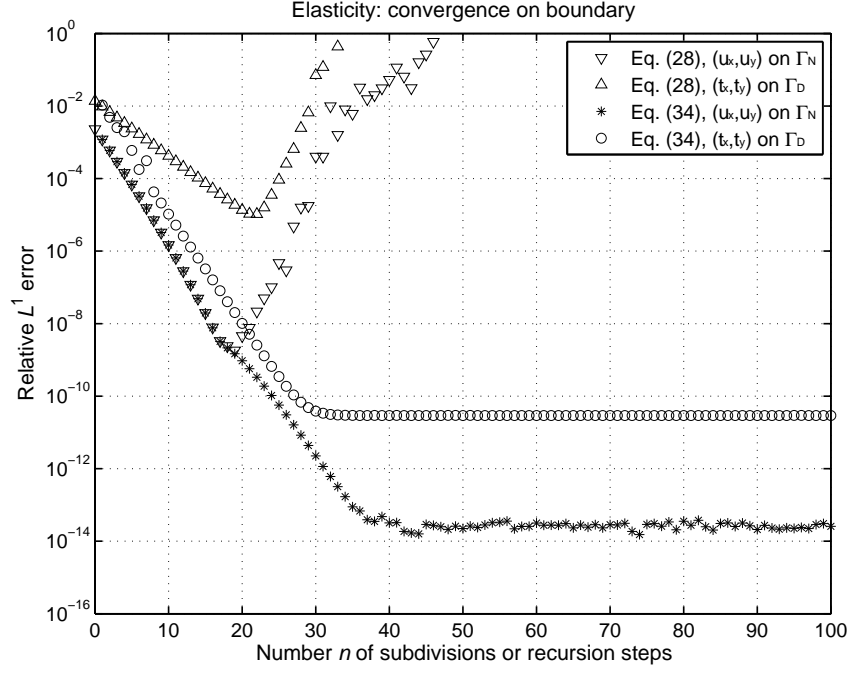
24

Figure 8: Convergence for $\mathcal{U}(z)$ of (59) on Γ_N and for $\mathcal{T}(z)$ of (60) on Γ_D. The first example of boundary conditions, see (61), is used.

where

$$\theta(\tau) = \frac{d\omega(\tau)}{d\tau}, \quad \tau \in \Gamma_D, \quad \text{and} \quad \theta(\tau) = \omega(\tau), \quad \tau \in \Gamma_N, \qquad (64)$$

and we have used

$$\lim_{\tau \in \Gamma_D \to \gamma_i} \omega(\tau) = 0, \quad i = 1, 2. \qquad (65)$$

The limit value on Γ (from the interior) is

$$\Phi(z)^+ = \frac{\theta(z)}{2} + \frac{1}{2\pi i} \int_\Gamma \frac{\theta(\tau)\,d\tau}{\tau - z}, \quad z \in \Gamma, \qquad (66)$$

and one can now evaluate $\Phi(z)$ arbitrarily close to Γ via

$$\Phi(z) = \frac{\sum_{k=1}^{N} \frac{\Phi^+(\tau_k)\tau_k' w_k}{\tau_k - z}}{\sum_{k=1}^{N} \frac{\tau_k' w_k}{\tau_k - z}}, \quad z \in \Omega. \qquad (67)$$
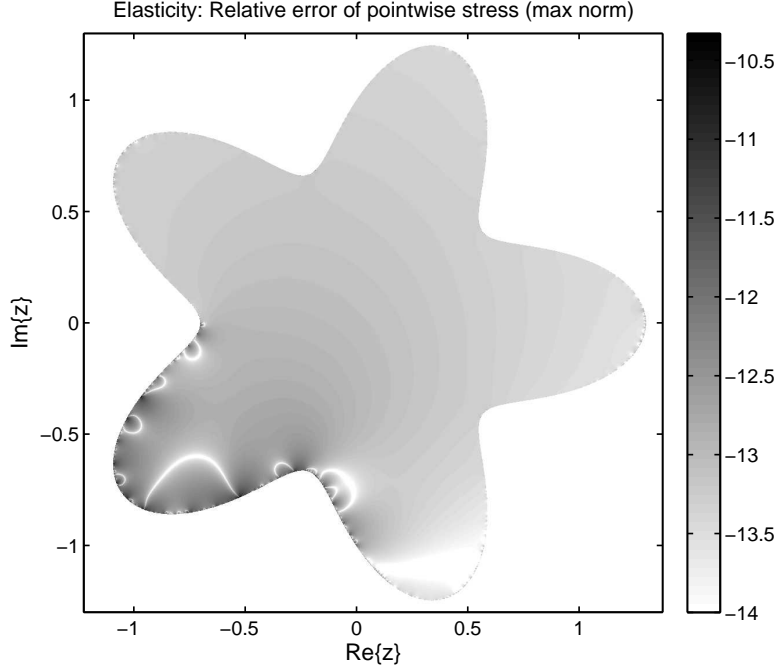
25

Figure 9: The 10-logarithm of the pointwise relative error when computing the trace of the stress tensor $\sigma_{xx} + \sigma_{yy}$ at 484670 points in Ω. The first example of boundary conditions is used. The maximum pointwise error is $4.7 \cdot 10^{-11}$.

A cartesian grid of 1000×1000 points $z = x + \mathrm{i}y$ is created in the square $x \in [-1.225, 1.375]$, $y \in [-1.3, 1.3]$. Out of these, 484670 points are located in Ω and some are very close to Γ. The first example of boundary conditions is chosen and the elasticity analogue of (34) is solved as above. For maximum accuracy and to illustrate stability with respect to overresolution we take 96 panels for the coarse mesh on Γ and $n = 100$ steps in all recursions. The Cauchy integral in (66) is computed from values of $\theta(\tau)$ on the fine grid and by global regularization (3), where the fast multipole method is used with the origin placed both at γ_1 and at γ_2 (for different parts of the integral) as to reduce cancellation. The reconstruction of ω_{fin}, needed for $\theta(\tau)$ on the fine grid in (64), is computed via the elasticity analogue of (47). The final field evaluation (67) is done with two regular fast multipole calls. The MATLAB code is far from ideal, speed wise, and the total running time for the full computation is around 24 seconds. About 75% of this time is spent in the post-processor.

26

Fig. 9 shows the pointwise relative error, in max norm, at the 484670 internal points. The error is typically around 10^{-12} and nowhere greater than $4.7 \cdot 10^{-11}$. Not even close to γ_1 and γ_2, where $\theta(z)$ diverges. This is satisfying and in accordance with the error in traction on Γ_D, see Fig. 8.

11. Discussion

This paper demonstrates that the combination of

- integral equations which are of Fredholm's second kind with compact operators except for close to a finite number of boundary points

- high-order accurate composite quadrature

- multilevel compression

produces fast and accurate results for elliptic problems with mixed boundary conditions. If one or more of these items are lacking, an integral equation scheme is less likely to excel compared to other techniques such as the finite element method. Its stability may even be in jeopardy.

Particular features of our paper are unified treatment and robust solution strategies; the same techniques are used irrespective of whether a singularity stems from a change of boundary conditions or from the boundary itself, the same recursions are used for a wide range of singular operators. We believe that our techniques could be extended to many problems, also in three dimensions. What needs to be worked out fresh is, essentially, integral equations of the right type. If this pays off in difficult situations is another question, whose answer may depend on what is of primary concern – accuracy and stability or storage and speed. We refrain from further speculations with one exception: The extension of the present scheme to multiple portions of the boundary with different boundary conditions is straight-forward. Nowhere in the derivation of the scheme we have used that Γ of Fig. 1 only has two singular boundary points. Furthermore, Section 11 of Ref. [8] gives an example where recursive compressed inverse preconditioning is successfully applied to a connected boundary containing over ten thousand points with geometric singularities.

We hope that these findings will contribute to greater flexibility and applicability of integral equation methods, primarily for mixed planar problems in elasticity and Stokes flow.

References

[1] K.E. Atkinson, *The numerical Solution of Integral Equations of the Second Kind*, Cambridge University Press, Cambridge, (1997).

[2] S.C. Brenner and L. Ridgway Scott, *The Mathematical Theory of Finite Element Methods, 3rd ed.*, Section 11.1, Springer, New York (2008).

[3] J. Englund, Stable algorithm for the stress field around a multiply branched crack, Int. J. Num. Meth. Engng, **63**(6), 926–946 (2005).

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

[5] L. Greengard, M.C. Kropinski, and A. Mayo, Integral equation methods for Stokes flow and isotropic elasticity in the plane, J. Comput. Phys., **125**(2), 403–414 (1996).

[6] J. Helsing, On the numerical evaluation of stress intensity factors for an interface crack of a general shape, Int. J. Num. Meth. Engng, **44**(5), 729–741 (1999).

[7] J. Helsing and R. Ojala, On the evaluation of layer potentials close to their sources, J. Comput. Phys., **227**(5), 2899–2921 (2008).

[8] J. Helsing and R. Ojala, Corner singularities for elliptic problems: Integral equations, graded meshes, quadrature, and compressed inverse preconditioning, J. Comput. Phys., **227**(20), 8820–8840 (2008).

[9] J. Helsing and G. Peters, An efficient numerical algorithm for cracks partly in frictionless contact, SIAM J. Appl. Math., **61**(2), 551–566 (2000).

[10] N.I. Ioakimidis, K.E. Papadakis, and E.A. Perdios, Numerical evaluation of analytic functions by Cauchy's theorem, BIT Numer. Math., **31**(2), 276–285 (1991).

[11] A.E. Martínez-Castro and R. Gallego, Tangential residual as error estimator in the boundary element method, Comput. Struct., **83**(10/11), 685–699 (2005).

[12] P.G. Martinsson and V. Rokhlin, A fast direct solver for boundary integral equations in two dimensions, J. Comput. Phys., **205**(1), 1–23 (2005).

[13] S.G. Mikhlin, *Integral Equations and their applications to certain problems in mechanics, mathematical physics and technology, 2nd ed.*, Pergamon Press, London, (1964).

[14] N.I. Muskhelishvili, *Some Basic Problems of the Mathematical Theory of Elasticity, 3rd Ed.*, P. Noordhoff Ltd, Groningen (1953).

[15] N.I. Muskhelishvili, *Singular Integral Equations, 2nd Ed.*, P. Noordhoff N.V., Groningen (1953).

[16] D. Natroshvili and S. Zazashvili, Mixed type boundary value problems in the linear theory of elastic mixtures for bodies with interior cuts, Mem. Differential Equations Math. Phys., **42**, 69–91 (2007).

[17] N.A. Noda and C.H. Xu, Controlling parameter of the stress intensity factors for a planar interfacial crack in three-dimensional bimaterials, Int. J. Solids Struct., **45**(3/4), 1017–1031 (2008).

[18] N. Ottosen and H. Petersson, *Introduction to the Finite Element Method*, Section 16.4, Pearson Educational Ltd., Harlow (1992).

[19] Y. Saad and M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comp., **7**(3), 856–869 (1986).

[20] J.A. Sethian and J. Wilkening, A numerical model of stress driven grain boundary diffusion, J. Comput. Phys., **193**(1), 275–305 (2004).

[21] D.I. Sherman, On the problem of plane strain in non-homogeneous media. In *Non-Homogeneity in elasticity and plasticity*, ed. W. Olszag. (Proceedings of I.U.T.A.M. Symposium, Warsaw, 1958). Pergamon Press, London (1959)

[22] S. Tlupova and R. Cortez, Boundary integral solutions of coupled Stokes and Darcy flows, J. Comput. Phys., **228**(1), 158–179 (2009).

[23] T. Tran-Cong, T. Nguyen-Thien, and N. Phan-Thien, Boundary element method based on a new second kind integral equation formulation, Engng Anal. Bound. Elem., **17**(4), 313–320 (1996).

[24] Y.J. Wei, A.F. Bower, and H.J. Gao, Recoverable creep deformation and transient local stress concentration due to heterogeneous grain-boundary diffusion and sliding in polycrystalline solids, J. Mech. Phys. Solids, **56**(4), 1460–1483 (2008).