ON INTEGRAL EQUATION METHODS FOR THE FIRST DIRICHLET PROBLEM OF THE BIHARMONIC AND MODIFIED BIHARMONIC EQUATIONS IN NON-SMOOTH DOMAINS*

JOHAN HELSING† AND SHIDONG JIANG‡

Abstract. Despite important applications in unsteady Stokes flow, a Fredholm second kind integral equation formulation modeling the first Dirichlet problem of the modified biharmonic equation in the plane has been derived only recently. Furthermore, this formulation becomes very ill-conditioned when the boundary is not smooth, say, having corners. The present work demonstrates numerically that a method called recursively compressed inverse preconditioning (RCIP) can be effective when dealing with this geometrically induced ill-conditioning in the context of Nyström discretization. The RCIP method not only reduces the number of iterations needed in iterative solvers, but also improves the achievable accuracy in the solution. Adaptive mesh refinement is only used in the construction of a compressed inverse preconditioner, leading to an optimal number of unknowns in the linear system in the solve phase.

Key words. second kind integral equation, biharmonic equation, modified biharmonic equation, the RCIP method.

AMS subject classifications. 35G15, 45B05, 65F99, 65R20

1. Introduction.

1.1. Biharmonic and modified biharmonic Dirichlet problems.

This work is about solving the first Dirichlet problem of the biharmonic and the modified biharmonic equations,

$$\Delta^2 u = 0 \qquad \text{and} \qquad \Delta(\Delta - \lambda^2)u = 0\,, \tag{1}$$

in planar domains using integral equation techniques. Since both the biharmonic and the modified biharmonic equations are fourth order elliptic partial differential equations, two boundary conditions are needed for the solution to be unique. By the first Dirichlet problem we mean that both the solution u itself and its first normal derivative $\partial u/\partial \nu$ are specified on the domain boundary Γ.

The first Dirichlet problem of the biharmonic equation models the clamped plate problem in linear elasticity. The first Dirichlet problem of the modified biharmonic equation has its applications in solving unsteady Stokes flow via semidiscretization schemes. While solving the modified biharmonic problem is the primary motivation for the present work, the less involved biharmonic problem captures all the essential difficulties encountered when solving the modified biharmonic problem. This is why we discuss both problems. We refer the reader to [18] for a detailed discussion.

1.2. Second kind integral equations and their kernels.

In [10], a system of Fredholm second kind integral equations (SKIE) is constructed for the first Dirichlet problem of the biharmonic equation. In [18], the construction is extended to solve the first Dirichlet problem of the modified biharmonic equation. In [10, 18], the solution

*J. Helsing was supported by the Swedish Research Council under contract 621-2014-5159. S. Jiang was supported by NSF under grant DMS-1720405 and by the Flatiron Institute, a division of the Simons Foundation.

†Centre for Mathematical Sciences, Lund University, Box 118, 221 00 Lund, Sweden (helsing@maths.lth.se).

‡Department of Mathematical Sciences, New Jersey Institute of Technology, Newark, New Jersey, USA 07102 (shidong.jiang@njit.edu).

u is represented via a sum of layer potentials whose kernels are judiciously selected so that all integral operators are compact and the diagonal terms from the jump relations have a non-zero determinant. The difficulty in constructing these SKIEs lies in the requirement that one integral operator must have the property that both the operator itself and its normal derivative are compact and that the corresponding potential has a non-zero jump across the boundary. Extensive search indicates that there seems to be only one such operator (modulo lower order terms) for both the biharmonic and the modified biharmonic equations. Recently, [24] constructed a better conditioned SKIE system for the first Dirichlet problem of the biharmonic equation that is applicable for the case of multiply connected domains as well. The construction in [24] is based on a Goursat function representation of the solution to a related Stokes problem. However, it is difficult to extend to the modified biharmonic problem and it appears that there is no clear alternative to the SKIEs in [18] for the modified biharmonic problem.

The SKIEs of [10] and [18] contain kernels that behave very differently on smooth and on non-smooth boundaries. To illustrate this, let D be a planar domain in $\mathbb{R}^2$ with boundary Γ. Let $\mathbf{p}$ be a point on Γ, $\mathbf{x}$ and $\mathbf{y}$ two points on Γ on opposite sides of $\mathbf{p}$, $\boldsymbol{\nu}_x$ and $\boldsymbol{\nu}_y$ the outward unit normal on Γ at these points, $\boldsymbol{r} = \mathbf{x}-\mathbf{y}$, and $r = |\mathbf{x}-\mathbf{y}|$. In this notation, the kernels of the SKIEs contain terms such as

$$\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu}_y)^2(\boldsymbol{\nu}_x\cdot\boldsymbol{\nu}_y)}{r^4} \qquad \text{and} \qquad \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)(\boldsymbol{r}\cdot\boldsymbol{\nu}_y)^3}{r^6}\,. \tag{2}$$

When Γ is smooth, both $(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)$ and $(\boldsymbol{r}\cdot\boldsymbol{\nu}_y)$ are of the order $O(r^2)$ [9, Theorem 2.2], which makes the terms in (2) integrable and leads to the compactness of the associated integral operators. When Γ has a corner vertex at $\mathbf{p}$, on the other hand, the terms in (2) behave like $1/r^2$ as $\mathbf{x}$ and $\mathbf{y}$ approach $\mathbf{p}$ from different sides. This is in stark contrast to the kernel of the Neumann–Poincaré operator (the double layer potential operator for Laplace's equation), which contains only the term $(\boldsymbol{r}\cdot\boldsymbol{\nu}_y)/r^2$ that behaves like $1/r$ as $\mathbf{x}$ and $\mathbf{y}$ approach a corner vertex $\mathbf{p}$ from different sides. Indeed, it is shown in [26] that the Neumann–Poincaré operator is a singular bounded operator on Lipschitz domains.

We remark here that many compact integral operators on smooth curves undergo similar changes as the Neumann–Poincaré operator in the vicinity of corners. However, as mentioned, some terms in the kernels of the SKIEs of [10, 18] become nearly hypersingular and the associated operators are unbounded around corner vertices. This is a much more difficult situation than what usually is encountered for corner problems, where kernels merely become nearly singular.

1.3. Numerical challenges. Solving SKIEs numerically on non-smooth boundaries involves a number of challenges. Layer densities representing the solution are often non-smooth close to boundary singularities. Such behaviors are difficult to resolve by polynomials, which underlie most approximation schemes. Adaptive mesh refinement towards corners is often needed. This is costly and may lead to artificial ill-conditioning and the loss of accuracy. Iterative solvers also converge slowly. All these problems are caused by the loss of compactness and only recently there have emerged efficient numerical treatments [3, 4, 5, 6, 7, 12, 13, 15, 25] in the context of operators that become nearly singular in corners. The problems are exacerbated in the presence of operators that become nearly hypersingular around corners.

Some particular difficulties that we face in the present work are:

- The numerical evaluation of kernel terms such as the first term in (2) may suffer from severe cancellation when r is small and $\mathbf{x}$ and $\mathbf{y}$ are on the same

side of a corner vertex.
- The leading integral operators in the SKIEs are not scale invariant on wedges. Scale invariance on wedges is often a helpful property when solving SKIEs numerically on Γ with corners. The lack of scale invariance poses an additional challenge.
- The presence of corners on Γ makes the Fredholm index of the SKIEs positive. To be more precise, our numerical observations indicate that the index of the Fredholm operator is increased by one for each corner.
- The presence of nearly hypersingular operators in the representation of the gradient field ∇u may lead to severe loss of accuracy in the numerical evaluation of the gradient field very close to Γ.

1.4. Our approach and outline of the paper. Solving the SKIEs of [10, 18] numerically in non-smooth domains is of interest for modeling unsteady Stokes flow. The RCIP method [12, 13, 15, 16], which is a lossless compression and acceleration scheme in the context of Nyström discretization, has been effectively used for simpler SKIEs than these ones. Central to the RCIP method is mesh refinement in corners and a recursive inversion formula based on kernel splitting. Applying RCIP to the SKIEs of [10, 18] is nontrivial because the kernels become nearly hypersingular near corners and the SKIEs themselves become non-invertible. This leads to challenges, related to exponential ill-conditioning of discretized system matrices and the loss of floating point accuracy, irrespective of what numerical solution method is used.

In this paper we develop a modified RCIP method suitable for nearly hypersingular kernels. First, we apply row-column equilibration to a particular form of block matrix inversion for increased stability. Second, we use scaling of hypersingular integral operators to devise a good initializer for the recursion. Third, we derive a more intricate kernel splitting procedure designed to improve the accuracy of the entries of intermediate matrices. Combined with an interpolation scheme for the accurate evaluation of gradient fields close to the boundary, our approach can be used to efficiently evaluate both u and its gradient down to machine precision for the (modified) biharmonic problem in the entire computational domain.

The rest of the paper is organized as follows. Section 2 reviews the integral equation formulations in [10, 18]. Section 3 is about explicit kernel-split Nyström discretization of the SKIEs. The RCIP method is reviewed in section 4 and the modifications particular to the present paper appear in section 5. Section 6 discusses accurate evaluation of the solution and its gradient. Section 7 illustrates the performance of the entire numerical scheme via several examples. Finally, we conclude the paper with a short discussion on the extensions and applications of the work.

2. SKIE formulations. This section reviews the SKIEs of [10, 18], for the first Dirichlet problem of the biharmonic and the modified biharmonic equations, in a uniform exposition.

2.1. Notation. We shall frequently use Green's functions $G(\mathbf{x}, \mathbf{y})$ whose arguments $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$ are points in $\mathbb{R}^2$ and are called *target* point and *source* point, respectively. The unit normal vector at $\mathbf{x}$ on Γ is $\boldsymbol{\nu}_x$, but the unit normal vector at $\mathbf{y}$ is simply denoted by $\boldsymbol{\nu}$. We use $\boldsymbol{r} = \mathbf{x} - \mathbf{y}$, $r = |\mathbf{x} - \mathbf{y}|$, and $\mathrm{d}s_y$ as an element of arc length. We use the same boldfaced letter to denote an integral operator, its kernel, and the associated interaction matrix after discretization. This should not cause any confusion as the meaning of the symbol is clear from the context.

To simplify expressions, we shall at times use complex variable notation alongside

with real notation. We then identify the point (x_1, x_2) in $\mathbb{R}^2$ with the point $\tau = x_1 + \mathrm{i}x_2$ in $\mathbb{C}$. Furthermore, we identify $\mathbf{x}$ with z_{t}, $\mathbf{y}$ with z_{s}, $\boldsymbol{\nu}$ with n_{s}, $\mathbf{y} - \mathbf{x}$ with z, and ∇u with $u_{x_1} + \mathrm{i}u_{x_2}$. The conjugation operator is denoted by Conj, or indicated with an overbar symbol. The real part of an expression is abbreviated Re. Note that

$$n_{\mathrm{s}} = -\mathrm{i}\dot{z}_{\mathrm{s}}(t)/|\dot{z}_{\mathrm{s}}(t)|\,, \tag{3}$$

where $z_{\mathrm{s}}(t)$ is a parameterization of Γ and $\dot{z}_{\mathrm{s}}(t) = \mathrm{d}z_{\mathrm{s}}(t)/\mathrm{d}t$.

2.2. The first Dirichlet problem of the biharmonic equation. Consider the following boundary value problem:

$$\Delta^2 u = 0 \quad \text{in} \quad D\,, \tag{4}$$

$$u = f_1 \quad \text{on} \quad \Gamma\,, \tag{5}$$

$$\frac{\partial u}{\partial \nu} = f_2 \quad \text{on} \quad \Gamma\,. \tag{6}$$

The fundamental solution to the biharmonic equation is

$$G(\mathbf{x}, \mathbf{y}) = \frac{1}{8\pi} r^2 \log(r)\,. \tag{7}$$

In [10], u is represented via a sum of two biharmonic multi-layer potentials

$$u(\mathbf{x}) = \int_\Gamma \left[G_1(\mathbf{x}, \mathbf{y})\sigma_1(\mathbf{y}) + G_2(\mathbf{x}, \mathbf{y})\sigma_2(\mathbf{y})\right] \mathrm{d}s_y\,, \tag{8}$$

where G_1 and G_2 are defined in (12). The boundary conditions (5) and (6) together with jump relations lead to the following SKIE system for the unknown layer densities σ_1 and σ_2:

$$\mathbf{D}(\mathbf{x})\boldsymbol{\sigma}(\mathbf{x}) + \int_\Gamma \widetilde{\mathbf{K}}(\mathbf{x}, \mathbf{y})\boldsymbol{\sigma}(\mathbf{y})\,\mathrm{d}s_y = \mathbf{f}(\mathbf{x})\,, \quad \mathbf{x} \in \Gamma\,, \tag{9}$$

where

$$\mathbf{D}(\mathbf{x}) = \begin{bmatrix} \frac{1}{2} & 0 \\ -\kappa(\mathbf{x}) & \frac{1}{2} \end{bmatrix}, \quad \mathbf{x} \in \Gamma\,, \tag{10}$$

$\kappa(\mathbf{x})$ is the signed curvature at $\mathbf{x}$, and

$$\boldsymbol{\sigma}(\mathbf{x}) = \begin{bmatrix} \sigma_1(\mathbf{x}) \\ \sigma_2(\mathbf{x}) \end{bmatrix}, \quad \mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \end{bmatrix}, \quad \widetilde{\mathbf{K}}(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} G_{11}(\mathbf{x}, \mathbf{y}) & G_{12}(\mathbf{x}, \mathbf{y}) \\ G_{21}(\mathbf{x}, \mathbf{y}) & G_{22}(\mathbf{x}, \mathbf{y}) \end{bmatrix}. \tag{11}$$

The kernels G_i and G_{ij}, $i, j = 1, 2$, are

$$\begin{aligned}
G_{11}(\mathbf{x}, \mathbf{y}) &= G_1(\mathbf{x}, \mathbf{y}) = -\frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})^3}{\pi r^4}\,, \\
G_{12}(\mathbf{x}, \mathbf{y}) &= G_2(\mathbf{x}, \mathbf{y}) = \frac{1}{4\pi} - \frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})^2}{2\pi r^2}\,, \\
G_{21}(\mathbf{x}, \mathbf{y}) &= \frac{\partial G_1}{\partial \nu_x} = -3\frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})^2(\boldsymbol{\nu} \cdot \boldsymbol{\nu}_x)}{\pi r^4} + 4\frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})^3(\boldsymbol{r} \cdot \boldsymbol{\nu}_x)}{\pi r^6}\,, \\
G_{22}(\mathbf{x}, \mathbf{y}) &= \frac{\partial G_2}{\partial \nu_x} = -\frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})(\boldsymbol{\nu} \cdot \boldsymbol{\nu}_x)}{\pi r^2} + \frac{(\boldsymbol{r} \cdot \boldsymbol{\nu})^2(\boldsymbol{r} \cdot \boldsymbol{\nu}_x)}{\pi r^4}\,.
\end{aligned} \tag{12}$$

Since many applications require the evaluation of the gradient of the solution we use the last two equations in (12) to obtain

$$\nabla u(\mathbf{x}) = \int_\Gamma \left[G_{1c}(\mathbf{x},\mathbf{y})\sigma_1(\mathbf{y}) + G_{2c}(\mathbf{x},\mathbf{y})\sigma_2(\mathbf{y})\right] \mathrm{d}s_y \,, \tag{13}$$

with kernels

$$\begin{aligned} G_{1c}(\mathbf{x},\mathbf{y}) &= -3\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 n_s}{\pi r^4} - 4\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3 z}{\pi r^6} \,, \\ G_{2c}(\mathbf{x},\mathbf{y}) &= -\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu}) n_s}{\pi r^2} - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 z}{\pi r^4} \,. \end{aligned} \tag{14}$$

2.3. The first Dirichlet problem of the modified biharmonic equation. Consider the following boundary value problem:

$$\Delta(\Delta - \lambda^2)u = 0 \quad \text{in} \quad D \,, \tag{15}$$

$$u = f_1 \quad \text{on} \quad \Gamma \,, \tag{16}$$

$$\frac{\partial u}{\partial \nu} = f_2 \quad \text{on} \quad \Gamma \,. \tag{17}$$

The fundamental solution to the modified biharmonic equation is

$$G(\mathbf{x},\mathbf{y}) = -\frac{1}{2\pi\lambda^2}\left(\log(r) + K_0(\lambda r)\right) \,, \tag{18}$$

where λ is assumed to be real and positive and K_0 is the modified Bessel function of the second kind of order zero [23]. In [18], an SKIE is derived for (15)–(17). The representation for u and the consequent SKIE system are formally identical to (8)–(11), but the kernels G_i and G_{ij} now become

$$\begin{aligned} G_{11}(\mathbf{x},\mathbf{y}) &= G_1(\mathbf{x},\mathbf{y}) = -\frac{\boldsymbol{r}\cdot\boldsymbol{\nu}}{\pi r^2}\mathcal{C}_1(\lambda r) + \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3}{\pi r^4}\mathcal{C}_2(\lambda r) \,, \\ G_{12}(\mathbf{x},\mathbf{y}) &= G_2(\mathbf{x},\mathbf{y}) = -\left(\frac{1}{2\pi} - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2}{\pi r^2}\right)\mathcal{C}_0(\lambda r) \,, \\ G_{21}(\mathbf{x},\mathbf{y}) &= -\frac{(\boldsymbol{\nu}\cdot\boldsymbol{\nu}_x)}{\pi r^2}\mathcal{C}_1(\lambda r) + 3\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2(\boldsymbol{\nu}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\mathcal{C}_2(\lambda r) \\ &\quad + \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\mathcal{C}_3(\lambda r) - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^6}\mathcal{C}_4(\lambda r) \,, \\ G_{22}(\mathbf{x},\mathbf{y}) &= 2\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})(\boldsymbol{\nu}\cdot\boldsymbol{\nu}_x)}{\pi r^2}\mathcal{C}_0(\lambda r) + \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{2\pi r^2}\mathcal{C}_5(\lambda r) \\ &\quad - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\mathcal{C}_2(\lambda r) \,, \end{aligned} \tag{19}$$

where the functions $\mathcal{C}_k$ $(k = 0, \ldots, 5)$ are

$$\begin{aligned} \mathcal{C}_0(x) &= K_0(x) + \frac{2}{x}K_1(x) - \frac{2}{x^2} \,, \\ \mathcal{C}_1(x) &= 3\mathcal{C}_0(x) + xK_1(x) + \frac{1}{2} \,, \\ \mathcal{C}_2(x) &= 4\mathcal{C}_0(x) + xK_1(x) \,, \\ \mathcal{C}_3(x) &= 12\mathcal{C}_0(x) + 5xK_1(x) + x^2K_0(x) + 1 \,, \\ \mathcal{C}_4(x) &= 24\mathcal{C}_0(x) + 8xK_1(x) + x^2K_0(x) \,, \\ \mathcal{C}_5(x) &= 2\mathcal{C}_0(x) + xK_1(x) \,, \end{aligned} \tag{20}$$

and K_1 is the modified Bessel function of the second kind of order one.

The gradient of the solution to the modified biharmonic problem can be computed via (13) but with the kernels in (14) replaced with

$$
\begin{aligned}
G_{1\mathrm{c}}(\mathbf{x},\mathbf{y}) = & -\frac{n_{\mathrm{s}}}{\pi r^2}\mathcal{C}_1(\lambda r) + 3\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 n_{\mathrm{s}}}{\pi r^4}\mathcal{C}_2(\lambda r) - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})z}{\pi r^4}\mathcal{C}_3(\lambda r) \\
& + \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3 z}{\pi r^6}\mathcal{C}_4(\lambda r)\,, \\
G_{2\mathrm{c}}(\mathbf{x},\mathbf{y}) = & \ 2\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\mathrm{s}}}{\pi r^2}\mathcal{C}_0(\lambda r) - \frac{z}{2\pi r^2}\mathcal{C}_5(\lambda r) + \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 z}{\pi r^4}\mathcal{C}_2(\lambda r)\,.
\end{aligned}
\tag{21}
$$

We note that neither (9) with (12) nor (9) with (19) has a nontrivial null space for interior Dirichlet problems in a simply connected domain with smooth boundary.

3. Discretization of the SKIE system. The biharmonic SKIE system (9) with (12) is discretized using a standard panel-based Nyström scheme and n_{gl}-point Gauss–Legendre quadrature. The quadrature weight at a node z_{sk} is denoted w_k.

The modified biharmonic SKIE system (9) with (19) is discretized using that same Nyström scheme, but supplemented with a product integration scheme [14, Section 6.1] which is activated when $\mathbf{y}$ is close to $\mathbf{x}$. This requires that each kernel is split into a smooth part and a logarithmically singular part

$$
G_{ij}(\mathbf{x},\mathbf{y}) = G_{ij}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) + G_{ij}^{(\mathrm{L})}(\mathbf{x},\mathbf{y})\log(r)\,,
\tag{22}
$$

where the factor $G_{ij}^{(\mathrm{L})}$ is known in analytic form and both $G_{ij}^{(\mathrm{S})}$ and $G_{ij}^{(\mathrm{L})}$ are smooth in the sense that, for a fixed $\mathbf{x}$, their dependence on $\mathbf{y}$ is well approximated panelwise by polynomials of degree $n_{\mathrm{gl}} - 1$. Without ambition of being exhaustive, the rest of this section summarizes some results needed to implement the discretizations of (9) with (12) and (19).

Limits of the kernels in (12), as $\mathbf{y} \to \mathbf{x}$, are computed using

$$
\lim_{\mathbf{y}\to\mathbf{x}} \frac{\boldsymbol{r}\cdot\boldsymbol{\nu}}{r^2} = -\frac{\kappa(\mathbf{x})}{2}
\tag{23}
$$

and read

$$
\begin{aligned}
&\lim_{\mathbf{y}\to\mathbf{x}} G_{11}(\mathbf{x},\mathbf{y}) = 0\,, && \lim_{\mathbf{y}\to\mathbf{x}} G_{12}(\mathbf{x},\mathbf{y}) = \frac{1}{4\pi}\,, \\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{21}(\mathbf{x},\mathbf{y}) = -\frac{3}{4\pi}\kappa^2(\mathbf{x})\,, && \lim_{\mathbf{y}\to\mathbf{x}} G_{22}(\mathbf{x},\mathbf{y}) = \frac{1}{2\pi}\kappa(\mathbf{x})\,.
\end{aligned}
\tag{24}
$$

For the kernels in (19), we need $G_{ij}^{(\mathrm{L})}$ of (22). These functions can be constructed from splittings

$$
\mathcal{C}_k(x) = \mathcal{C}_k^{(\mathrm{S})}(x) + \mathcal{C}_k^{(\mathrm{L})}(x)\log(x)\,, \qquad k = 0,\ldots,5\,,
\tag{25}
$$

which, in turn, rely on splittings

$$
K_n(x) = K_n^{(\mathrm{S})}(x) + K_n^{(\mathrm{L})}(x)\log(x)\,, \qquad n = 0, 1\,.
\tag{26}
$$

From the definition of K_n [23] it follows

$$
K_0^{(\mathrm{L})}(x) = -J_0(\mathrm{i}x)\,, \qquad K_1^{(\mathrm{L})}(x) = -\mathrm{i}J_1(\mathrm{i}x)\,,
\tag{27}
$$

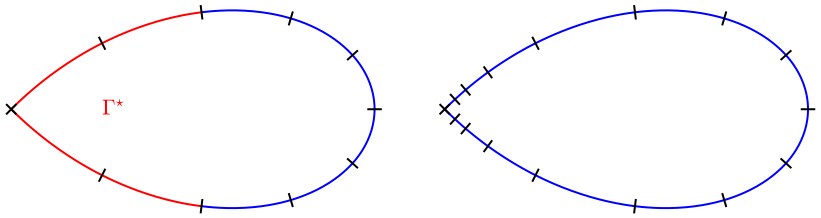

FIG. 1. *Left: the coarse mesh used in the solve phase of the RCIP method. The boundary subset* $\Gamma^\star$ *contains the four panels closest to the corner. These panels constitute a type* c *mesh on* $\Gamma^\star$. *Right: the fine mesh obtained from the coarse mesh by* $n_{\text{sub}} = 3$ *times dyadically refining the panels closest to the corner.*

where J_n are Bessel functions of the first kind. Furthermore, it is straightforward to compute the limits

$$
\begin{aligned}
&\lim_{\mathbf{y}\to\mathbf{x}} G_{11}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) = 0\,,\\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{12}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) = \frac{1}{4\pi}\,,\\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{21}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) = -\frac{3}{4\pi}\kappa^2(\mathbf{x}) - \frac{\lambda^2}{8\pi}\left(\log\left(\frac{\lambda}{2}\right) + \frac{1}{4} + \gamma\right)\,,\\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{22}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) = \frac{1}{2\pi}\kappa(\mathbf{x}),\\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{11}^{(\mathrm{L})}(\mathbf{x},\mathbf{y}) = \lim_{\mathbf{y}\to\mathbf{x}} G_{12}^{(\mathrm{L})}(\mathbf{x},\mathbf{y}) = \lim_{\mathbf{y}\to\mathbf{x}} G_{22}^{(\mathrm{L})}(\mathbf{x},\mathbf{y}) = 0,\\
&\lim_{\mathbf{y}\to\mathbf{x}} G_{21}^{(\mathrm{L})}(\mathbf{x},\mathbf{y}) = -\frac{\lambda^2}{8\pi}\,,
\end{aligned}
\tag{28}
$$

where γ is the Euler–Mascheroni constant.

When $x = \lambda r$ is small, we evaluate the $\mathcal{C}_k^{(\mathrm{L})}(x)$ of (25) using series expansions of the order $O(x^{16})$, or higher, to avoid numerical cancellation.

4. An overview of the RCIP method. This section gives a brief overview of the RCIP method for solving integral equations on curves Γ with various types of isolated point singularities. The exposition follows [13], which in turn builds on [12, 15, 16]. We assume that a system of n_{eq} integral equations takes the standard form

$$
(\mathbf{I} + \mathbf{K})\boldsymbol{\rho}(\mathbf{x}) = \mathbf{f}(\mathbf{x})\,, \tag{29}
$$

where $\mathbf{I}$ is the identity operator, $\mathbf{K}$ contains integral operators that are compact when Γ is smooth, and $\boldsymbol{\rho}$ contains unknown layer densities. For the ease of discussion, and without loss of generality, we will use the one-corner curve Γ from [13, Eq. (1)] with parameterization

$$
z_{\mathrm{s}}(t) = \sin(\pi t) e^{\mathrm{i}\theta(t-0.5)}\,, \quad t \in [0,1]\,, \tag{30}
$$

and with $\theta = \pi/2$ (in our illustrations).

A standard numerical technique for resolving point singularities is adaptive mesh refinement toward the singular point, as shown in Figure 1. That is, the linear system to be solved after Nyström discretization is

$$
(\mathbf{I}_{\text{fin}} + \mathbf{K}_{\text{fin}})\,\boldsymbol{\rho}_{\text{fin}} = \mathbf{f}_{\text{fin}}\,, \tag{31}
$$

where the system size is $n_{\text{fin}} \times n_{\text{fin}}$. With n_{pan} panels on the coarse mesh and n_{sub} refinement levels on the coarse panels closest to the corner we have $n_{\text{fin}} = n_{\text{eq}}(n_{\text{pan}} + 2n_{\text{sub}})n_{\text{gl}}$.

There are several issues associated with (31). First, $\mathbf{K}$ is not compact when Γ has corners and (29) is not a second kind equation. The linear system (31) becomes increasingly ill-conditioned as n_{sub} grows and iterative solvers converge slowly. Second, the system size of (31) increases as n_{sub} grows. Third, it is difficult to determine *a priori* the number n_{sub} needed to meet a given desired accuracy. Under-refinement will fail to resolve $\boldsymbol{\rho}$, while over-refinement often leads to an excessive computational cost, instability, and the loss of accuracy due to the multiscale nature of the graded mesh.

The RCIP method resolves these issues. It starts from the splitting of the interaction matrix in the form

$$\mathbf{K} = \mathbf{K}^{\circ} + \mathbf{K}^{\star}, \tag{32}$$

where $\mathbf{K}^{\star}$ is nonzero only when both the target and source points are on the boundary subset $\Gamma^{\star}$ containing the four coarse panels closest to the corner vertex, see Figure 1. One can say that $\mathbf{K}^{\star}$ is the singular and near-singular part of the interaction matrix that is capable of capturing the singularities in $\boldsymbol{\rho}$ around the corner, while $\mathbf{K}^{\circ}$ contains the smooth part. We remark here that $\star$ and $\circ$ as superscripts on any other matrix always refer to the aforementioned splitting. The final linear system in the RCIP method is

$$(\mathbf{I}_{\text{coa}} + \mathbf{K}^{\circ}_{\text{coa}}\mathbf{R})\,\tilde{\boldsymbol{\rho}}_{\text{coa}} = \mathbf{f}_{\text{coa}}\,, \tag{33}$$

where $\mathbf{I}_{\text{coa}}$ is the identity matrix on the coarse mesh, $\mathbf{K}^{\circ}_{\text{coa}}$ is $\mathbf{K}^{\circ}$ discretized on the coarse mesh, $\mathbf{R}$ is the *recursively compressed inverse preconditioner*, and $\tilde{\boldsymbol{\rho}}_{\text{coa}}$ is a transformed layer density discretized on the coarse mesh. The density $\tilde{\boldsymbol{\rho}}_{\text{coa}}$ may be used directly with $\mathbf{R}$ and smooth quadratures to calculate many quantities of interest in the post-processing stage. The linear system (33) has size $n_{\text{coa}} \times n_{\text{coa}}$ with $n_{\text{coa}} = n_{\text{eq}}n_{\text{pan}}n_{\text{gl}}$, independent of n_{sub}.

The matrix $\mathbf{R}$ coincides with the identity matrix except at those entries where $\mathbf{K}^{\star}_{\text{coa}}$ is nonzero. We denote that part of $\mathbf{R}$ by $\mathbf{R}^{\star}$. The size of $\mathbf{R}^{\star}$ is $n_{\text{c}} \times n_{\text{c}}$ with $n_{\text{c}} = 4n_{\text{eq}}n_{\text{gl}}$. In other words, the matrix $\mathbf{K}^{\circ}_{\text{coa}}$ has a zero block of size $n_c \times n_c$, and $\mathbf{R}^{\star}$ is exactly located at the position of the zero block of $\mathbf{K}^{\circ}_{\text{coa}}$. In practice, this zero block may be distributed in various places in $\mathbf{K}^{\circ}_{\text{coa}}$ depending on the ordering of the discretization points. In [13, Appendix D], it is shown that $\mathbf{R}^{\star}$ can be computed as the matrix $\mathbf{R}_{n_{\text{sub}}}$ obtained via the recursion

$$\mathbf{R}_i = \mathbf{P}^{T}_{W\text{bc}} \left(\mathbb{F}\{\mathbf{R}^{-1}_{(i-1)}\} + \mathbf{M}^{\circ}_i \right)^{-1} \mathbf{P}_{\text{bc}}\,, \quad i = 1, \ldots, n_{\text{sub}}\,, \tag{34}$$

with

$$\mathbf{M}_i = \mathbf{I}_{\text{b}} + \mathbf{K}_{i\text{b}}\,. \tag{35}$$

Here $\mathbf{R}_0$ is an initializer wich can be set to $\mathbf{I}^{\star}_{\text{b}}$. $\mathbf{P}_{\text{bc}}$ is a prolongation matrix of size $n_{\mathsf{b}} \times n_{\text{c}}$ ($n_{\text{b}} = 6n_{\text{eq}}n_{\text{gl}}$) carrying out interpolation from points on a *type* c mesh to points on a *type* b mesh on the same boundary part. $\mathbf{P}_{W\text{bc}} = \mathbf{W}_{\text{b}}\mathbf{P}_{\text{bc}}\mathbf{W}^{-1}_{\text{c}}$, where $\mathbf{W}_{\text{b}}$ and $\mathbf{W}_{\text{c}}$ are two diagonal matrices with quadrature weights for *type* b and *type* c meshes on the diagonal, respectively. The operator $\mathbb{F}\{\cdot\}$ expands its matrix argument

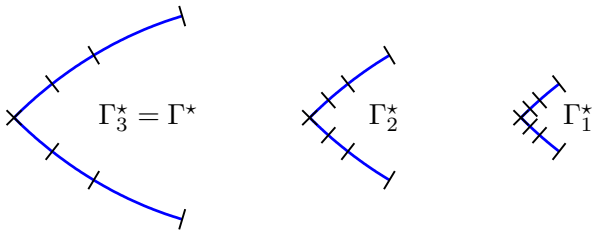

FIG. 2. Three boundary subsets $\Gamma_3^\star$, $\Gamma_2^\star$, and $\Gamma_1^\star$, used for the discretization of $\mathbf{K}_{i\mathsf{b}}$ in (35), along with their corresponding type b meshes for $n_{\text{sub}} = 3$. Note that $i = 1$ is the finest level and that $\Gamma_{n_{\text{sub}}}^\star$ coincides with $\Gamma^\star$ in Figure 1.

by zero-padding, and both the target and source points of $\mathbf{K}_{i\mathsf{b}}$ are on a type b mesh on the boundary subset $\Gamma_i^\star$. The reader is referred to Figures 1 and 2 for a few examples of subsets $\Gamma_i^\star$ and of the so-called type b and type c meshes at different refinement levels and to [13, Appendix D] for a detailed explanation of (34) and the construction of the meshes involved. Here we simply note that at each recursion step one only needs to invert an $n_{\text{b}} \times n_{\text{b}}$ matrix.

The recursion (34) starts from the smallest six panels in the fine mesh, gradually moves up in the hierarchy, and finally reaches $\Gamma^\star$ in the coarse mesh. This way, $\mathbf{R}^\star$ encapsulates all the information about the singularities of the unknowns around the corner in the fine mesh and the computational cost grows only linearly with n_{sub}. The overall accuracy is essentially independent of the choice of n_{sub} as long as n_{sub} is sufficiently large. We remark that the RCIP method captures the singularities of $\boldsymbol{\rho}$ in a purely numerical and algorithmic fashion. That is, it is kernel-independent or even problem-independent and there is no need to find the singular behavior of $\boldsymbol{\rho}$ analytically in advance.

5. Difficulties associated with the SKIE system (9) and modifications of the RCIP method. The SKIE system (9) has $n_{\text{eq}} = 2$. We choose $n_{\text{gl}} = 16$ in the discretization and thus $n_{\text{c}} = 128$ and $n_{\text{b}} = 192$. The system (9) can easily be converted into the form (29) via the change of variables

$$\boldsymbol{\rho} \equiv \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} = \mathbf{D}\boldsymbol{\sigma} \quad \text{and} \quad \mathbf{K} = \widetilde{\mathbf{K}}\mathbf{D}^{-1}. \tag{36}$$

As pointed out in subsection 1.1, the biharmonic problem captures the essential difficulties of the modified biharmonic problem. Hence, we will use the system matrix $\mathbf{I} + \mathbf{K}$ with kernels from (12) as an illustration in this section, unless the modified biharmonic problem is mentioned specifically. We observe that the stability of the recursion (34) is tightly related to the properties of the matrix $\mathbf{M}_i$ of (35). Thus, we present numerical evidence on $\mathbf{M}_i$, with Γ as in (30) and $n_{\text{pan}} = 24$, to elaborate the challenges outlined in subsection 1.3 and discuss modifications of the RCIP method needed to overcome these challenges.

5.1. Avoiding inversion of ill-conditioned matrices. The matrix $\mathbf{M}_i$ stemming from (9) via (36), for a fixed i, becomes more and more ill-conditioned as n_{sub} increases. Figure 3 shows the singular values of $\mathbf{M}_1$ for $n_{\text{sub}} = 20$ and the largest and the smallest singular values of $\mathbf{M}_i$ for $i = 1, \ldots, n_{\text{sub}}$. We observe that the largest singular value is doubled and the smallest singular value is halved at each consecutive refinement level, $i = n_{\text{sub}}, \ldots, 1$. Already with $n_{\text{sub}} = 20$ – a number of refinement

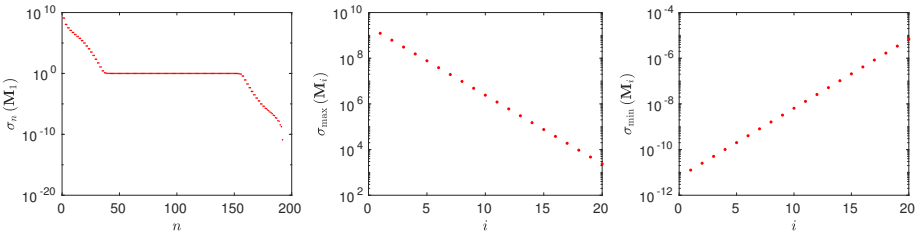

FIG. 3. *Left: singular values σ_n of $\mathbf{M}_1$ versus n, $n = 1, \ldots, 192$. Middle: largest singular values of $\mathbf{M}_i$ versus refinement level i. Right: smallest singular values of $\mathbf{M}_i$ versus refinement level i. Here $n_{\text{sub}} = 20$, $i = 1, \ldots, n_{\text{sub}}$, and $i = 1$ is the finest level.*

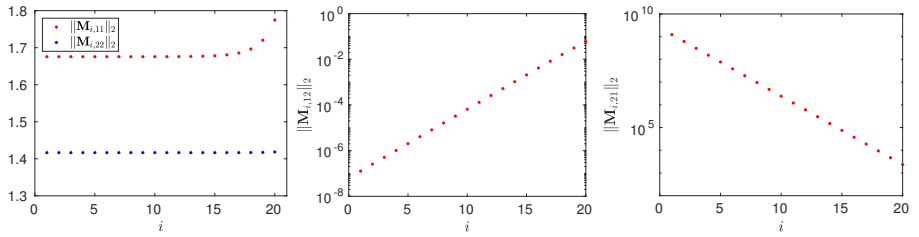

FIG. 4. *Left: 2-norms of diagonal blocks $\mathbf{M}_{i,11}$ and $\mathbf{M}_{i,22}$ versus refinement level i. Middle: 2-norms of $\mathbf{M}_{i,12}$ versus refinement level i. Right: 2-norms of $\mathbf{M}_{i,21}$ versus refinement level i. Here $n_{\text{sub}} = 20$, $i = 1, \ldots, n_{\text{sub}}$, and $i = 1$ is the finest level.*

levels usually insufficient for resolving the singularities of the unknown density – the condition number of $\mathbf{M}_1$ is greater than 10^{20}, making inversion risky in double precision arithmetic. The severe ill-conditioning of $\mathbf{M}_i$ leads to the ill-conditioning of $\mathbf{R}_i$. As pointed out in the Introduction, the ill-conditioning of $\mathbf{M}_i$ is closely related to the fact that the kernel $G_{21}(\mathbf{x}, \mathbf{y})$ in (12) behaves like $O\left(1/\|\mathbf{x} - \mathbf{y}\|^2\right)$ and that the associated integral operator becomes hypersingular around a corner. The more common situation is when a compact operator becomes a bounded singular operator around a corner and then $\mathbf{R}_i$ is well-conditioned.

Fortunately, it is not necessary to compute the recursion (34) exactly as it stands and the inversion of $\mathbf{R}_{i-1}$ can be avoided using block matrix inversion techniques. We implement $\left(\mathbb{F}\{\mathbf{R}_{i-1}^{-1}\} + \mathbf{M}_i^{\circ}\right)^{-1}$ using [17, Eq. (8)] as in [16, Appendix B]. This implementation leads to stabilization.

5.2. Inversion of poorly balanced matrices. The matrix $\mathbf{M}_i$ of (35), stemming from (9), is not only ill-conditioned but also poorly balanced. To see this, we partition $\mathbf{M}_i$ into 2×2 blocks of equal size

$$\mathbf{M}_i = \begin{bmatrix} \mathbf{M}_{i,11} & \mathbf{M}_{i,12} \\ \mathbf{M}_{i,21} & \mathbf{M}_{i,22} \end{bmatrix}. \tag{37}$$

The partitioning follows naturally from the discretization of G_{11}, G_{12}, etc. and the associated diagonal blocks. Figure 4 shows the 2-norms of these four blocks. We observe that the norms of the diagonal blocks $\mathbf{M}_{i,11}$ and $\mathbf{M}_{i,22}$ are $O(1)$ and more or less independent of the refinement level i. However, the norm of $\mathbf{M}_{i,12}$ is roughly halved and the norm of $\mathbf{M}_{i,21}$ is roughly doubled at each consecutive refinement level, $i = n_{\text{sub}}, \ldots, 1$. This is so since the kernel G_{12} remains bounded and the kernel G_{21} scales like $O(1/r^2)$ around the corner, and the element of arclength $\mathrm{d}s$ is reduced by a factor of 2 from level i to level $i - 1$.

The poor balance of $\mathbf{M}_i$ will lead to severe loss of accuracy when solving related linear systems or computing the inverse of certain related matrices in the recursion (34). Here we use a simple row-column equilibration [11, Section 3.5.2] as follows: Suppose that we try to solve a linear system $\mathbf{A}_i x = b$ with $\mathbf{A}_i$ having similar 2×2 block structure as $\mathbf{M}_i$. By row-column equilibration we mean that instead of solving $\mathbf{A}_i x = b$, we solve $(\mathbf{D}_i^{-1}\mathbf{A}_i\mathbf{D}_i)y = \mathbf{D}_i^{-1}b$ and set $x = \mathbf{D}_i y$. $\mathbf{D}_i$ is a diagonal matrix

$$\mathbf{D}_i = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & s_i\mathbf{I} \end{bmatrix} \tag{38}$$

with $\mathbf{I}$ the identity matrix, $\mathbf{0}$ the zero matrix, and the scaling factor $s_i = 2^{n_{\text{sub}}-i}$. It is straightforward to verify that $\mathbf{D}_i^{-1}\mathbf{A}_i\mathbf{D}_i$ has no effect on diagonal blocks of $\mathbf{A}_i$, but compensates the poor balance of two off-diagonal blocks of $\mathbf{A}_i$ so that their entries are of order $O(1)$, i.e., about the same order as diagonal blocks. Similarly, the inverse of $\mathbf{A}_i$ is computed via the formula $\mathbf{A}_i^{-1} = \mathbf{D}_i(\mathbf{D}_i^{-1}\mathbf{A}_i\mathbf{D}_i)^{-1}\mathbf{D}_i^{-1}$. This technique makes the recursion much more accurate and robust for the SKIEs studied in this paper.

5.3. Initiating the recursion. The choice of the initializer $\mathbf{R}_0$ in (34) has a great influence on the number of refinement levels n_{sub} needed to resolve an SKIE. Let, for a given i, $\mathbf{R}_i^{\text{c}}$ denote the matrix to which $\mathbf{R}_{n_{\text{sub}}-i}$ converges as $n_{\text{sub}} \to \infty$ in (34), that is, as the mesh is infinitely refined. The best initializer on a mesh that is n_{sub} times refined (n_{sub} now finite) is $\mathbf{R}_0 = \mathbf{R}_{n_{\text{sub}}}^{\text{c}}$. In the present work, we use an approximation of that initializer that is asymptotically correct for large n_{sub}.

If $\mathbf{M}_i$ of (35) is scale invariant on wedges, then the series of matrices $\mathbf{R}_i^{\text{c}}$, $i = 0, 1, \ldots$, converges rapidly to a limit $\mathbf{R}^{\text{c}}$ and the initializer can be set to $\mathbf{R}_0 = \mathbf{R}^{\text{c}}$. Often only the first 40 or 50 $\mathbf{R}_i^{\text{c}}$ are disctinct in double precision arithmetic. This means that when n_{sub} is large, the majority of the steps in the recursion (34) can be viewed as a fixed-point iteration for $\mathbf{R}^{\text{c}}$ defined by

$$\mathbf{R}^{\text{c}} = \mathbf{P}_{W\text{bc}}^T \left(\mathbb{F}\{(\mathbf{R}^{\text{c}})^{-1}\} + \mathbf{M}_{\text{w}}^{\circ}\right)^{-1} \mathbf{P}_{\text{bc}}\,. \tag{39}$$

Here $\mathbf{M}_{\text{w}}$ is $\mathbf{I}_{\text{b}} + \mathbf{K}_{\text{b}}$ on a finite wedge with the same opening angle as the corner in (30), for example

$$z_{\text{s}}(t) = \begin{cases} -\pi t e^{0.5\text{i}\theta}\,, & t \in [-2, 0]\,, \\ \pi t e^{-0.5\text{i}\theta}\,, & t \in [0, 2]\,. \end{cases} \tag{40}$$

The fixed-point property can be used to find $\mathbf{R}^{\text{c}}$ without knowing precisely what n_{sub} are needed for the $\mathbf{R}_i^{\text{c}}$ to converge or at what index i the $\mathbf{R}_i^{\text{c}}$ cease to be distinct. One just iterates (39) until convergence is reached. See, further, [13, Section 11].

For the SKIEs of the present work, $\mathbf{M}_i$ is not scale invariant on wedges and the scheme of the preceding paragraph needs to be modified. We observe that

$$\mathbf{M}_{i+1} = \mathbf{D}_1^{-1}\mathbf{M}_i\mathbf{D}_1\,, \qquad n_{\text{sub}} - i \gg 1\,, \tag{41}$$

$$\mathbf{R}_i^{\text{c}} = \mathbf{D}_1^{-1}\mathbf{R}_{i+1}^{\text{c}}\mathbf{D}_1\,, \qquad i \gg 1\,, \tag{42}$$

where $\mathbf{D}_1$ are square diagonal matrices of appropriate dimensions with diagonal entries either unity or 2. Now, let $2\Delta t$ be the side lengths, in parameter, of the legs of the type b mesh on which $\mathbf{M}_1$ is discretized in (34) and let $\mathbf{M}_{\text{w}}$ be defined on the finite wedge (40). One can then, for large n_{sub}, find a matrix $\mathbf{R}^{\text{c}}$ such that

$$\mathbf{R}_{n_{\text{sub}}}^{\text{c}} = \mathbf{D}_{n_{\text{sub}}}\mathbf{R}^{\text{c}}\mathbf{D}_{n_{\text{sub}}}^{-1}\,, \tag{43}$$

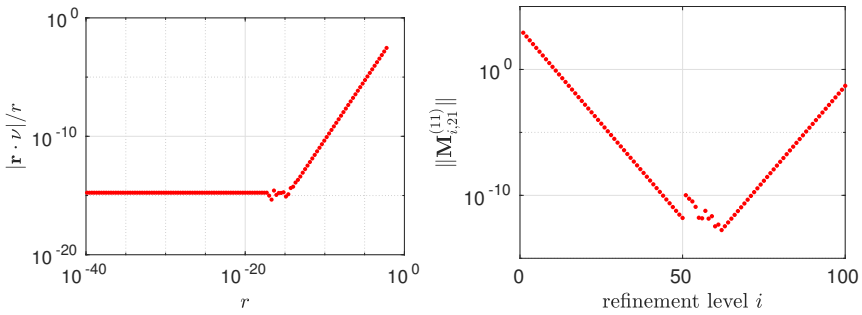

FIG. 5. *Effect of round-off errors for small r. Left: scaling of $\boldsymbol{r}\cdot\boldsymbol{\nu}/r$ in double precision floating-point arithmetic. Right: instability when calculating smooth blocks of $\mathbf{M}_{i,21}$ – the norm of $\mathbf{M}_{i,21}^{(11)}$ versus refinement level i for $n_{\rm sub}=100$. Here $\mathbf{M}_{i,21}^{(11)}$ is the 11-block of $\mathbf{M}_{i,21}$ upon further partition of that matrix into 2×2 blocks of equal size.*

where the diagonal matrix $\mathbf{D}_{n_{\rm sub}}$ has diagonal entries either unity or $(\Delta t)^{-1}$ and

$$\mathbf{R}^{\rm c}=\mathbf{D}_1\mathbf{P}_{W\rm bc}^T\left(\mathbb{F}\{(\mathbf{R}^{\rm c})^{-1}\}+\mathbf{M}_{\rm w}^{\circ}\right)^{-1}\mathbf{P}_{\rm bc}\mathbf{D}_1^{-1}\,. \tag{44}$$

Fixed-point iteration on (44) until convergence gives $\mathbf{R}^{\rm c}$. Then the initializer $\mathbf{R_0}=\mathbf{R}^{\rm c}_{n_{\rm sub}}$ is obtained from (43).

5.4. Reducing the effect of round-off errors. In the original RCIP method, the splitting (32) is done on a strict geometric basis: the interaction between the four panels closest to a corner vertex is accounted for in $\mathbf{K}^{\star}$, while the remaining interaction is accounted for in $\mathbf{K}^{\circ}$. Since local coordinates with origin at the corner vertices are used for kernel evaluation, there are typically no problems related to numerical cancellation.

In the present work the situation is different. Round-off errors can be amplified and make the recursion (34) unstable, particularly for small indices i with large $n_{\rm sub}$. When the target point $\mathbf{x}$ and the source point $\mathbf{y}$ are on the same side of a corner vertex and $r=\|\mathbf{x}-\mathbf{y}\|$ is very small, then $\boldsymbol{r}\cdot\boldsymbol{\nu}$ or $\boldsymbol{r}\cdot\boldsymbol{\nu}_x$ will not scale like $O(r^2)$ as they should [9, Theorem 2.2]. To see this, let $\boldsymbol{r}\cdot\boldsymbol{\nu}=r_1-r_2$ with $r_1=(\mathbf{x}-\mathbf{y})_1\nu_1=\sigma\bar{r}_1\cdot 10^e$ and $r_2=-(\mathbf{x}-\mathbf{y})_2\nu_2=\sigma\bar{r}_2\cdot 10^e$, where $\sigma=\pm 1$ is the sign, e is the exponent, and $\bar{r}_1$, $\bar{r}_2$ are the significands of r_1 and r_2, respectively. Now $\bar{r}_1$ will be very close to $\bar{r}_2$ when r is very small. However, with large probability, $\bar{r}_1$ will not be exactly equal to $\bar{r}_2$ due to round-off error. Instead, we have $r_1-r_2=c_1\epsilon_{\rm mach}\cdot 10^e$ with c_1 an $O(1)$ constant. Since 10^e is about the same order as r, we have

$$\boldsymbol{r}\cdot\boldsymbol{\nu}=O(\epsilon_{\rm mach}r)\,, \tag{45}$$

rather than the correct value $O(r^2)$, when r is very small.

The error in (45) does not have much effect on refined discretizations of SKIEs if $\mathbf{K}(\mathbf{x},\mathbf{y})$ contains only the term $\boldsymbol{r}\cdot\boldsymbol{\nu}/r^2$. Consider, for example, the off-diagonal jk-entry in the corresponding matrix $\mathbf{K}_{\rm ib}$ of (35). As both the quadrature weight $w_{i,k}$ and the distance $r_{i,jk}$ are roughly halved at each consecutive refinement level, $i=n_{\rm sub},\ldots,1$, we have $w_{i,k}=O(r_{i,jk})$. Hence

$$\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})_{i,jk}}{r_{i,jk}^2}|\dot{z}_{{\rm s}i,k}|w_{i,k}=O(\epsilon_{\rm mach})\,, \tag{46}$$

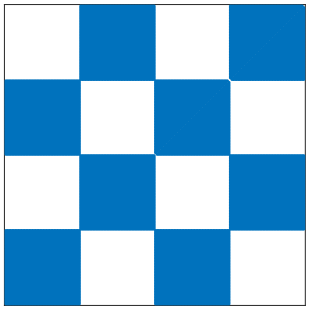

FIG. 6. *The new definition of* $\mathbf{K}^\star$ *– white blocks are set to zero in* $\mathbf{K}^\star$ *and transferred to* $\mathbf{K}^\circ$ *in (32) in the RCIP method for the SKIE system (9) with (12) and (36).*

rather than the correct value $O(r_{i,jk})$, when $r_{i,jk}$ is very small. The effect of the error in (46) is negligible as the identity will dominate smooth blocks of the corresponding matrix $\mathbf{M}_i$ in (35).

When $\mathbf{K}(\mathbf{x},\mathbf{y})$ contains the term $(\boldsymbol{r}\cdot\boldsymbol{\nu})^2/r^4$, the effect of round-off error is much more severe. A similar finite-precision analysis as above leads to

$$\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2_{i,jk}}{r^4_{i,jk}}|\dot{z}_{si,k}|w_{i,k} = O\left(\frac{\epsilon^2_{\text{mach}}}{r_{i,jk}}\right) \tag{47}$$

rather than to the correct value $O(r_{i,jk})$. The magnitude of the right hand side in (47) increases as $r_{i,jk}$ decreases and it will become $O(1)$ when $r_{i,jk} \approx \epsilon^2_{\text{mach}}$. This occurs in double-precision arithmetic, say, when $n_{\text{sub}} = 100$ and $i = 1$ since then $r_{i,jk} \approx 2^{-100}/16 \approx 5\cdot 10^{-32}$. Numerical experiments show that the error in (47) destroys the stability of the recursion (34). Figure 5 shows the round-off effect on calculating $\boldsymbol{r}\cdot\boldsymbol{\nu}$ when $\mathbf{x}$ and $\mathbf{y}$ are on the same side of the corner and the instability, due to the term $(\boldsymbol{r}\cdot\boldsymbol{\nu})^2/r^4$, when calculating smooth blocks of $\mathbf{M}_{i,21}$ in (37) for the biharmonic problem.

The second term $(\boldsymbol{r}\cdot\boldsymbol{\nu})^3(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)/r^6$ in (2), while also being hypersingular when $\mathbf{x}$ and $\mathbf{y}$ are on the opposite side of the corner, leads to negligible error in the calculation of matrix entries when $\mathbf{x}$ and $\mathbf{y}$ are on the same side. This is because

$$\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3_{i,jk}(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)_{i,jk}}{r^6_{i,jk}}|\dot{z}_{si,k}|w_{i,k} = O\left(\frac{\epsilon^4_{\text{mach}}}{r_{i,jk}}\right) \tag{48}$$

in floating-point arithmetic. This quantity will remain small (say, less than ϵ_{mach}) as long as $r_{i,jk} < \epsilon^3_{\text{mach}} \approx 10^{-47}$, which is true even when $n_{\text{sub}} = 120$, $i = 1$ for the curve defined by (30).

5.4.1. Stable splitting for the biharmonic problem. To overcome the problem with round-off errors diverging with refinement for smooth integrands in (9), we note that the splitting (32) does not have to be done on a strict geometric basis. The important property of $\mathbf{K}^\star$ is that it captures the non-smooth part of $\mathbf{K}$ near corners. We therefore abandon strict geometric splitting criteria and let as much as possible of the kernel $\mathbf{K}(\mathbf{x},\mathbf{y})$ be accounted for in $\mathbf{K}^\circ$.

To be more specific, for the biharmonic problem (9) with (12) and (36), $\mathbf{K}^\star$ is nonzero only if both the target and source points are on $\Gamma^\star$ but lie on different sides of the corner vertex. That is, if we partition $\mathbf{K}^\star$ into 4×4 blocks (there are 2 equations

in our SKIE system), then every other block is set to zero in $\mathbf{K}^\star$ and transferred to $\mathbf{K}^\circ$. We illustrate the new splitting in Figure 6. The new splitting reduces the cancellation error in the calculation of $\mathbf{K}_{i\mathrm{b}}$, makes half of the entries of $\mathbf{K}_{i\mathrm{b}}$ zero, and stabilizes the recursion (34) completely.

5.4.2. Stable splitting for the modified biharmonic problem. For the modified biharmonic problem (9) with (19) and (36), operators with logarithmically singular kernels are not smooth and thus need to be kept in $\mathbf{K}^\star$. It is straightforward to split those operators into two parts – $\mathbf{K}^{(\mathrm{S})}$ and $\mathbf{K}^{(\mathrm{L})}$, with $\mathbf{K}^{(\mathrm{S})}$ containing smooth terms, which would lead to large cancellation error unless they are included in $\mathbf{K}^\circ$, and $\mathbf{K}^{(\mathrm{L})}$ containing the rest. Therefore, $\mathbf{K}^{(\mathrm{S})}$ is split as in subsection 5.4.1, while $\mathbf{K}^{(\mathrm{L})}$ is split as in the original RCIP method. Here we observe that $\mathbf{K}^{(\mathrm{S})} = \widetilde{\mathbf{K}}^{(\mathrm{S})}\mathbf{D}^{-1}$ with $\widetilde{\mathbf{K}}^{(\mathrm{S})}$ given by the following expressions:

$$\widetilde{\mathbf{K}}^{(\mathrm{S})} = \begin{bmatrix} G_{11}^{(\mathrm{S})} & G_{12}^{(\mathrm{S})} \\ G_{21}^{(\mathrm{S})} & G_{22}^{(\mathrm{S})} \end{bmatrix}, \tag{49}$$

and

$$\begin{aligned}
G_{11}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) &= \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3}{\pi r^4}\left(-1+\frac{\lambda^2 r^2}{8}\right), \\
G_{12}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) &= \frac{1}{2\pi}\left(\frac{1}{2}-\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2}{r^2}\right), \\
G_{21}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) &= 3\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2(\boldsymbol{\nu}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\left(-1+\frac{\lambda^2 r^2}{8}\right) - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\frac{\lambda^2 r^2}{8} \\
&\quad - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^6}\left(-4+\frac{\lambda^2 r^2}{4}-\frac{\lambda^4 r^4}{48}\right), \\
G_{22}^{(\mathrm{S})}(\mathbf{x},\mathbf{y}) &= -\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})(\boldsymbol{\nu}\cdot\boldsymbol{\nu}_x)}{\pi r^2} - \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2(\boldsymbol{r}\cdot\boldsymbol{\nu}_x)}{\pi r^4}\left(-1+\frac{\lambda^2 r^2}{8}\right).
\end{aligned} \tag{50}$$

6. Accurate evaluation of the solution and its gradient. Once the discretized system (9) is solved for pointwise values of the unknown layer density $\boldsymbol{\sigma}$, the solution $u(\mathbf{x})$ and its gradient $\nabla u(\mathbf{x})$ can be obtained in D via discretization of (8) and (13). When $\mathbf{x}$ is some distance away from Γ, the underlying standard quadrature is sufficient to give high accuracy.

When $\mathbf{x}$ is close to Γ, special evaluation methods are needed. See [1, 20] for different options. We shall again resort to product integration, extending the scheme developed in [12, 14]. This time we need to split the kernels concerned into several terms which are all products of a given type of near-singularity and a smooth function. The explicit kernel-split needed for a kernel G_1, G_2, $G_{1\mathrm{c}}$, or $G_{2\mathrm{c}}$ reads in general form

$$G_i(\mathbf{x},\mathbf{y}) = G_i^{(\mathrm{S})}(z_\mathrm{t}, z_\mathrm{s}) + G_i^{(\mathrm{L})}(z_\mathrm{t}, z_\mathrm{s})\log|z| + \sum_{m=1}^{3} \mathrm{Re}\left\{G_i^{(m)}(z_\mathrm{t}, z_\mathrm{s})\frac{n_\mathrm{s}}{z^m}\right\}, \tag{51}$$

where $G_i^{(1)}$ is a smooth factor for the near-singularity, $G_i^{(2)}$ is a smooth factor for a near-hypersingularity, and $G_i^{(3)}$ is a smooth factor for a near-supersingularity. For the complex kernels $G_{1\mathrm{c}}$ and $G_{2\mathrm{c}}$ of (14) and (21) we may need to omit Re in (51), or replace it with Conj in certain situations.

At the heart of our product integration scheme lies panelwise polynomial approximation of layer densities, coefficients determined via the solution of Vandermonde systems, and the evaluation of integrals over intervals of Γ of the type

$$
\begin{aligned}
s_k^{(m)} &= \int_a^b \tau^{k-1} \log(\tau - z_{\mathrm{t}})\, \mathrm{d}\tau\,, \qquad && m = \mathrm{L}\,, \\
s_k^{(m)} &= \int_a^b \frac{\tau^{k-1}\, \mathrm{d}\tau}{(\tau - z_{\mathrm{t}})^m}\,, && m = 1, 2, 3\,,
\end{aligned}
\tag{52}
$$

where $k = 1, \ldots, n_{\mathrm{gl}}$ and a and b are consecutive breakpoints of the quadrature panels on Γ. The integrals in (52) can readily be evaluated in a semi-analytic fashion via translation, rotation, scaling of the plane, and forward recursion. See [12, Section 2] for recursion formulas for $s_k^{(m)}$, $m = 1, 2$, and [14, Appendix B] for a MATLAB code that implements the entire product integration scheme for $m = \mathrm{L}, 1, 2$. The extension to $m = 3$ is straight-forward.

When $\mathbf{x}$ is very close to Γ, the product integration just discussed may suffer from numerical cancellation. Subsection 6.3, below, shows how to deal with that problem when ∇u is known on Γ.

6.1. Kernel splitting for the evaluation of u. This section shows that the kernels G_1 and G_2 of (8) with (12) or (19) can be split in the form (51). First, one can show that

$$
\begin{aligned}
\boldsymbol{r}\cdot\boldsymbol{\nu} &= -\mathrm{Re}\left\{n_{\mathrm{s}}\bar{z}\right\}\,, && (\boldsymbol{r}\cdot\boldsymbol{\nu})^2 = \frac{1}{2}(z\bar{z} + \mathrm{Re}\left\{n_{\mathrm{s}}^2\bar{z}^2\right\})\,, \\
\frac{\boldsymbol{r}\cdot\boldsymbol{\nu}}{r^2} &= -\mathrm{Re}\left\{\frac{n_{\mathrm{s}}}{z}\right\}\,, && \frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^3}{r^4} = -\mathrm{Re}\left\{\frac{n_{\mathrm{s}}}{2z}\right\} + \mathrm{Re}\left\{\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\mathrm{s}}^2}{2z^2}\right\}\,.
\end{aligned}
\tag{53}
$$

For the biharmonic kernels G_1 and G_2 of (12) it then follows

$$
G_1^{(1)} = \frac{1}{2\pi}\,, \qquad G_1^{(2)} = -\frac{1}{2\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\mathrm{s}}\,, \qquad G_2^{(1)} = \frac{1}{2\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})\,.
\tag{54}
$$

We emphasize that the term $(\boldsymbol{r}\cdot\boldsymbol{\nu})^2/r^2$, occurring in G_2, is smooth and bounded in the analytical sense, but cannot be well approximated by a low degree polynomial when the target point is close to the source panel. Therefore, it is to be considered as near-singular within our product integration scheme.

For the modified biharmonic kernels G_1 and G_2 of (19), the factors $G_1^{(\mathrm{L})}$ and $G_2^{(\mathrm{L})}$ are the same as for the on-boundary evaluation in section 3. The smooth factors of the other singularities are

$$
G_1^{(1)} = \frac{1}{2\pi} - \frac{\lambda^2}{8\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})^2\,, \qquad G_1^{(2)} = -\frac{1}{2\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\mathrm{s}}\,, \qquad G_2^{(1)} = \frac{1}{2\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})\,.
\tag{55}
$$

The extra term in $G_1^{(1)}$ of (55), not present in $G_1^{(1)}$ of (54), is due to that the first two terms in the asymptotic expansion of $\mathcal{C}_2(\lambda r)$ are $-1 + \lambda^2 r^2/8$. We note that the splittings of G_1 and G_2 in singular terms are clean in the sense that they do not contain any cross terms that multiply a near-logarithmic singularity with other types of near singularity.

6.2. Kernel splitting for the evaluation of ∇u. This section shows that the complex kernels of (13) with (14) or (21) can be split in the form (51), with Re omitted or replaced by Conj.

We first split the kernels of (14). Using

$$
\begin{aligned}
4\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 z}{r^6} &= \text{Conj}\left\{\frac{n_{\text{s}}^2}{z^3}\right\} + \frac{n_{\text{s}}^2}{z^2\bar{z}} + \frac{2}{z\bar{z}^2}\,, \\
3\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\text{s}}}{r^4} &= -\frac{3n_{\text{s}}^2}{2z^2\bar{z}} - \frac{3}{2z\bar{z}^2}\,, \\
\frac{(\boldsymbol{r}\cdot\boldsymbol{\nu})z}{r^4} &= -\text{Conj}\left\{\frac{n_{\text{s}}}{2z^2}\right\} - \frac{n_{\text{s}}}{2r^2}\,,
\end{aligned}
\tag{56}
$$

we obtain

$$
\begin{aligned}
G_{1\text{c}}(\mathbf{x},\mathbf{y}) &= -\frac{1}{2\pi}\left(\text{Conj}\left\{2(\boldsymbol{r}\cdot\boldsymbol{\nu})\frac{n_{\text{s}}^2}{z^3} - \frac{n_{\text{s}}}{2z^2}\right\} + \frac{n_{\text{s}}^3}{2z^2}\right), \\
G_{2\text{c}}(\mathbf{x},\mathbf{y}) &= \frac{1}{2\pi}\left((\boldsymbol{r}\cdot\boldsymbol{\nu})\text{Conj}\left\{\frac{n_{\text{s}}}{z^2}\right\} + n_{\text{s}}\text{Re}\left\{\frac{n_{\text{s}}}{z}\right\}\right).
\end{aligned}
\tag{57}
$$

The above expressions split $G_{1\text{c}}$ and $G_{2\text{c}}$ into a sum of near-singular, near-hypersingular, and near-supersingular terms with smooth prefactors.

We then split the kernels of (21). The resulting factors $G_{1c}^{(\text{L})}$ and $G_{2c}^{(\text{L})}$ are similar to $G_{21}^{(\text{L})}$ and $G_{22}^{(\text{L})}$ of the on-boundary evaluation in section 3. For the near singular terms, we note that the leading terms in the asymptotic expansion of the $\mathcal{C}_k$ are

$$
\begin{aligned}
&\mathcal{C}_0(\lambda r) = -\frac{1}{2} + O(r^2\log r)\,, \quad \mathcal{C}_1(\lambda r) = O(r^2\log r)\,, \\
&\mathcal{C}_2(\lambda r) = -1 + \frac{1}{8}\lambda^2 r^2 + O(r^4\log r)\,, \quad \mathcal{C}_3(\lambda r) = -\frac{1}{8}\lambda^2 r^2 + O(r^4\log r)\,, \\
&\mathcal{C}_4(\lambda r) = -4 + \frac{1}{4}\lambda^2 r^2 - \frac{1}{48}\lambda^4 r^4 + O(r^6\log r)\,, \quad \mathcal{C}_5(\lambda r) = O(r^2\log r)\,.
\end{aligned}
\tag{58}
$$

Combining (21) and (58), we obtain the near-singular parts $G_{1\text{c}}^{(\text{N})}$ and $G_{2\text{c}}^{(\text{N})}$ of $G_{1\text{c}}$ and $G_{2\text{c}}$ which can be arranged in accordance with (51) as

$$
\begin{aligned}
G_{1\text{c}}^{(\text{N})}(\mathbf{x},\mathbf{y}) &= -\frac{1}{2\pi}\left(\text{Conj}\left\{2(\boldsymbol{r}\cdot\boldsymbol{\nu})\frac{n_{\text{s}}^2}{z^3} - \frac{n_{\text{s}}}{2z^2}\right\} + \frac{n_{\text{s}}^3}{2z^2}\right) \\
&\quad + \frac{\lambda^2}{8\pi}(\boldsymbol{r}\cdot\boldsymbol{\nu})z\text{Re}\left\{\frac{n_{\text{s}}^2}{z^2}\right\} \\
&\quad - \frac{\lambda^2}{8\pi}\left(2z + 3(\boldsymbol{r}\cdot\boldsymbol{\nu})n_{\text{s}} - \frac{\lambda^2}{6}(\boldsymbol{r}\cdot\boldsymbol{\nu})^2 z\right)\text{Re}\left\{\frac{n_{\text{s}}}{z}\right\}, \\
G_{2\text{c}}^{(\text{N})}(\mathbf{x},\mathbf{y}) &= \frac{1}{2\pi}\left((\boldsymbol{r}\cdot\boldsymbol{\nu})\text{Conj}\left\{\frac{n_{\text{s}}}{z^2}\right\} + \left(n_{\text{s}} - \frac{\lambda^2}{4}(\boldsymbol{r}\cdot\boldsymbol{\nu})z\right)\text{Re}\left\{\frac{n_{\text{s}}}{z}\right\}\right).
\end{aligned}
\tag{59}
$$

We observe that setting $\lambda = 0$ in (59) gives back (57).

6.3. An interpolation scheme for very close evaluation. The product integration scheme will suffer from cancellation in terms like $1/(z_{\text{s}} - z_{\text{t}})^m$ when the target point z_{t} is away from the origin but very close to the boundary Γ. The evaluation of ∇u is particularly affected by this loss of precision. To fix the problem, we propose a simple interpolation scheme for ∇u. We invoke this scheme whenever $d(z_{\text{t}},\Gamma)/l_k < 0.04$, where $d(z_{\text{t}},\Gamma)$ is the distance from z_{t} to Γ and l_k is the length of the panel on which the boundary point nearest to z_{t} is situated. Typically, very few

TABLE 1
Strengths and positions of sources generating f_1.

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
q_k	0.171	0.720	0.918	0.825	0.078
$\mathrm{Re}\{z_k\}$	1.280	0.413	−0.235	−0.145	1.050
$\mathrm{Im}\{z_k\}$	0.177	0.795	0.315	−0.474	−0.580

target points will need this special treatment. Therefore, its use will not increase the cost of field evaluations in any significant way.

The interpolation scheme goes as follows. First, we find a boundary point z_s whose normal vector passes through z_t. This is done via Müller's method [22], which converges to machine precision in about four iterations. We then place $n_\mathrm{ip}+1$ interpolation points z_i on the line passing through z_s and z_t. The first point is $z_1 = z_\mathrm{s}$. The remaining points are chosen to be the transformed Chebyshev nodes of the second kind, as implemented by the following MATLAB code

```
x=cos(pi*(n_ip:-1:0)/n_ip);
z_i=z_s+(x+1)*l_k/2*(z_t-z_s)/abs(z_t-z_s);
```

In our implementation, we set $n_\mathrm{ip} = 8$ and evaluate ∇u at z_i, $i = 2, \ldots, 9$, using the product integration scheme of section 6. It is assumed that the value of ∇u at z_1 can be obtained directly from boundary data. Finally, barycentric interpolation is used to evaluate the gradient field at z_t. We observe, in numerical experiments, that the scheme is fairly robust and achieves better than 11-digit accuracy for ∇u and arbitrarily small $d(z_\mathrm{t}, \Gamma)$.

7. Numerical results. We now demonstrate the performance of our algorithms for (9), (8), and (13) with numerical examples. The codes are implemented in MATLAB, release 2017a. Output and timings are obtained on a 64 bit LINUX laptop with a 2.10GHz INTEL i7-4600U CPU with 4MB of cache in single core mode.

EXAMPLE 1. *The one-corner curve of (30) with $\theta = \pi/2$.*

In this example, D is the domain bounded by the curve Γ of (30) with $\theta = \pi/2$. Boundary data f_1 and f_2 are generated by a linear combination of fundamental solutions with strengths q_k at points z_k outside D. The combination of fundamental solutions for the two problems is

$$f_1(\mathbf{x}) = \sum_{k=1}^{5} q_k r_k^2 \log(r_k) \quad \text{and} \quad f_1(\mathbf{x}) = \sum_{k=1}^{5} q_k \left(\log(r_k) + K_0(\lambda r_k)\right), \tag{60}$$

where $r_k = |z_\mathrm{t} - z_k|$. q_k and z_k are chosen "randomly" as in Table 1. The errors are computed against analytical solutions. The linear system, corresponding to (33), is solved iteratively using GMRES with a stopping criterion threshold of ϵ_mach in the estimated relative residual. n_gl is set to 16. Other than part (E), the constant λ in (15) is set to 20; the curve is divided into 28 and 45 panels for the biharmonic and modified biharmonic problems, respectively.

(A). STABILITY ANALYSIS. The top row of Figure 7 illustrates how the relative error in the solution u and the number of GMRES iterations needed depend on the number of refinement levels n_sub used in (34) for the biharmonic problem. The bottom row of Figure 7 illustrates analogous dependencies for the modified biharmonic problem. The errors in u are measured at a single interior point $z_\mathrm{t} = 0.3 + 0.1\mathrm{i}$.

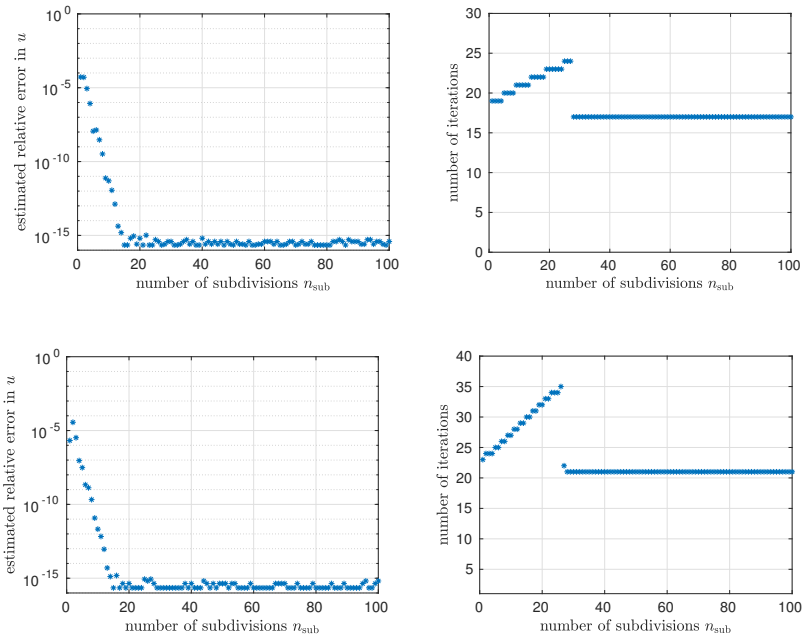

FIG. 7. *Stability for Example 1. Left column: convergence of u with n_{sub}. Right column: number of GMRES iterations needed as a function of n_{sub}. Top row: the biharmonic problem. Bottom row: the modified biharmonic problem.*

Figure 7 shows that the RCIP method is robust and accurate and leads to linear systems whose system matrices have nice spectral properties. The number of GMRES iterations needed is low, ranging from 17 to 35. The achieved precision is more or less independent of n_{sub} and nearly optimal, as long as n_{sub} is sufficiently large.

(B). ACCURACY ANALYSIS. We now seek u and ∇u in the entire domain D. A Cartesian grid of 300×300 equispaced points is placed on the rectangle $[-0.1, 1.1] \times [-0.5353, 0.5353]$ and evaluations are carried out at those $27,332$ grid points that are in D. Here $7,344$ target points activate the product integration scheme of section 6; $4,678$ target points close to the corner vertex require that $\boldsymbol{\rho}$ is reconstructed on a partially refined grid, see [13, Section 9]; 464 target points activate the interpolation scheme of subsection 6.3. The number of refinement levels is set to $n_{\text{sub}} = 40$. Figure 8 shows the absolute error in u and ∇u for the biharmonic and modified biharmonic problems, respectively.

From these results, and from similar experiments with other strengths q_k and points z_k than those in Table 1 and also with other corner opening angles θ, we conclude that our scheme achieves high accuracy in the entire computational domain.

(C). COMPUTATIONAL COST ON VARIOUS PARTS OF THE NUMERICAL SCHEME. The breakdown of the computational cost for the biharmonic problem is as follows. 0.88 seconds of CPU time is spent on the initializer (44) for $\mathbf{R}^{\text{c}}$; 0.21 seconds of CPU time is spent on the recursion (34) for $\mathbf{R}^{\star}$; 0.02 seconds of CPU time is spent on solving (33) for $\tilde{\boldsymbol{\rho}}_{\text{coa}}$ with GMRES; 23 seconds of CPU time is spent on computing u and ∇u from $\tilde{\boldsymbol{\rho}}_{\text{coa}}$ via (36) and discretizations of (8) and (13). The breakdown of the computational cost for the modified biharmonic problem is similar, although it generally takes longer time in the recursion and evaluation phases since the kernels

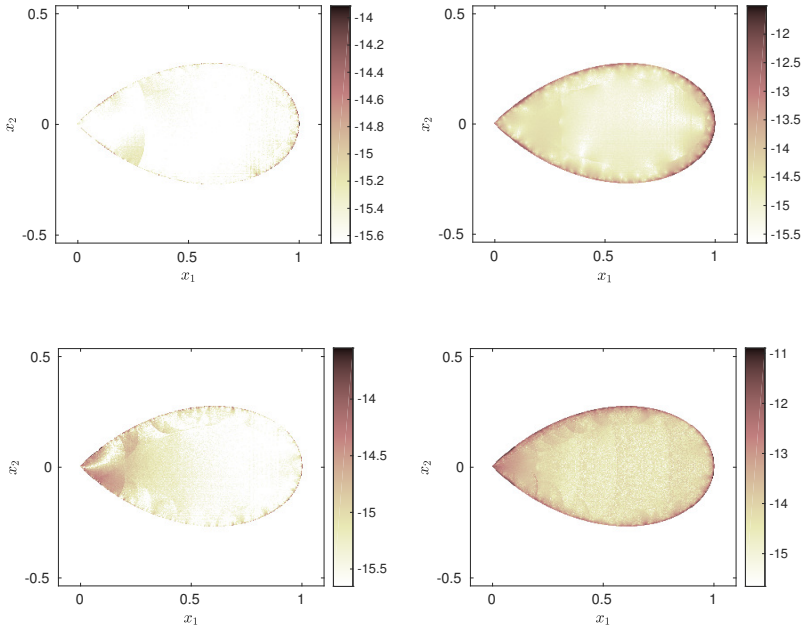

FIG. 8. *Accuracy in D for Example 1. Left column:* Log_{10} *of pointwise absolute error in* u. *Right column:* Log_{10} *of pointwise absolute error in* $|\nabla u|$. *Top row: the biharmonic problem. Bottom row: the modified biharmonic problem.*

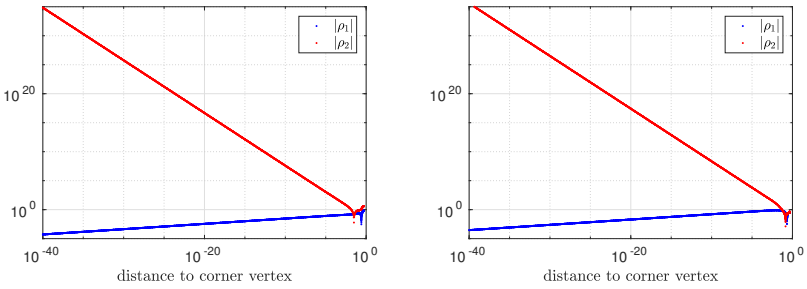

FIG. 9. *Asymptotic behavior of the layer densities* ρ_1 *and* ρ_2 *close to the corner vertex for Example 1. Left: the biharmonic problem. Right: the modified biharmonic problem.*

are more complicated. The time spent on GMRES is negligible compared to the time spent on the recursion (34) and, even more so, on evaluations of u and ∇u. We expect the time spent on evaluations of u and ∇u to be reduced if fast algorithms are constructed and incorporated.

(D). ASYMPTOTIC BEHAVIOR OF THE DENSITY. We emphasize that the RCIP method can use $\tilde{\boldsymbol{\rho}}_{\mathrm{coa}}$ and the matrices $\mathbf{R}_i$ of (34) to reconstruct $\boldsymbol{\rho}_{\mathrm{fin}}$ at all discretization points on the fine mesh [13, Section 9]. This makes it easy to zoom in on layer densities in regions where they vary rapidly. Figure 9 shows the asymptotic behavior of $\boldsymbol{\rho}$, related to $\boldsymbol{\sigma}$ via (36), close to the corner vertex. For both our problems, the asymptotic behavior is

$$\rho_i \propto d^{\beta_i} \quad i = 1, 2\,, \tag{61}$$

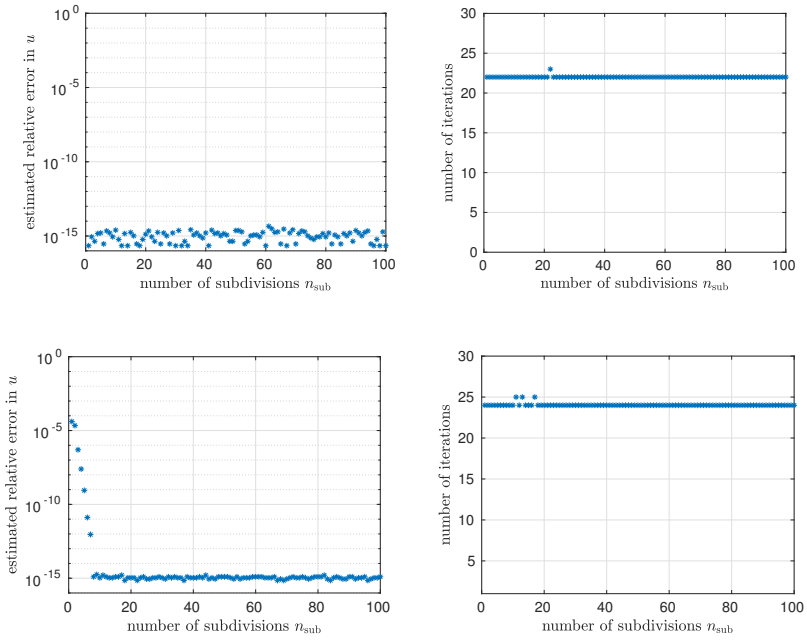

FIG. 10. *Stability for Example 2. Left column: convergence of u with n_{sub}. Right column: number of GMRES iterations needed as a function of n_{sub}. Top row: the biharmonic problem. Bottom row: the modified biharmonic problem.*

where d is the distance to the corner vertex and the singularity exponents are numerically determined to $\beta_1 = 0.0914708101539$ and $\beta_2 = -0.9085291898461$.

(E). DEPENDENCE ON λ FOR THE MODIFIED BIHARMONIC PROBLEM. We have also investigated the dependence on λ for the modified biharmonic problem. For $\lambda = 1,\ 10,\ 100$, we need 24, 32, 50 panels to fully resolve the density, and 16, 19, 22 number of iterations for GMRES to converge with stopping criterion of $10\epsilon_{\text{mach}}$, respectively. This shows that the number of GMRES iterations increases very mildly as λ increases.

EXAMPLE 2. *Triangle.*

It is clear that our numerical algorithms are applicable to curves with multiple corners. The construction of the preconditioner $\mathbf{R}^\star$ needs to be done only once for each different type of corner, which reduces the computational cost for certain geometries. For example, only one distinct $\mathbf{R}^\star$ is needed for all corners of a regular polygon. Here we illustrate the applicability of our algorithms to polygons by considering the domain D whose boundary Γ is given by a nonequilateral triangle with vertices at $v_j = e^{i\theta_j}$ $(j = 1, 2, 3)$. The angles θ_j are chosen "randomly" to be [0.258 2.20 4.27], n_{gl} is set to 18, and the total number of discretization points on the coarse mesh is 450 for the biharmonic problem and 684 for the modified biharmonic problem with $\lambda = 20$. Boundary data is generated by a linear combination of fundamental solutions with strengths q_k at points z_k outside D, where the q_k are the same as in Table 1, and $z_k = 1.5e^{i2\pi(k-1)/5}$ $(k = 1, \ldots, 5)$. The GMRES stopping criterion threshold is set to $20\epsilon_{\text{mach}}$. The errors in u are measured at a single interior point $z_{\text{t}} = 0.3 + 0.1\text{i}$ against analytical solutions.

The top row of Figure 10 illustrates how the relative error in the solution u and

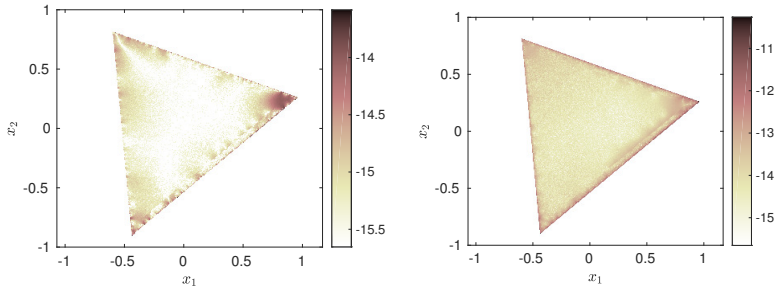

FIG. 11. *Accuracy in D for the modified biharmonic problem for Example 2. Left column:* Log_{10} *of pointwise absolute error in u. Right column:* Log_{10} *of pointwise absolute error in $|\nabla u|$.*

the number of GMRES iterations needed depend on the number of refinement levels n_{sub} used in (34) for the biharmonic problem. The bottom row of Figure 10 illustrates analogous dependencies for the modified biharmonic problem. The number of GMRES iterations is almost constant. For the biharmonic problem, the achieved precision is more or less independent of n_{sub} due to the presence of the initializer and the fact that all sides are straight line segments. For the modified biharmonic problem, the achieved precision also reaches its limit in double precision arithmetic quickly with respect to n_{sub}.

An accuracy study is carried out in the entire domain D. A Cartesian grid of 300×300 equispaced points is placed on the rectangle $[-1,\ 1.1] \times [-1,\ 1]$ and evaluations are carried out at those $27,401$ grid points that are in D. Here $4,388$ target points activate the product integration scheme of section 6; $1,500$ target points close to the corner vertex require that $\boldsymbol{\rho}$ is reconstructed on a partially refined grid; 239 target points activate the interpolation scheme of subsection 6.3. The number of refinement levels is set to $n_{\text{sub}} = 40$. Figure 11 shows the absolute error in u and ∇u for the modified biharmonic problem. The accuracy for the biharmonic problem is similar and therefore omitted.

8. Conclusions and further discussions. We have developed a numerical scheme to solve the SKIEs for the first Dirichlet problem of the biharmonic and modified biharmonic equations [10, 18] in non-smooth domains. The scheme modifies the RCIP method [13, 15], so that it can handle certain hypersingular integral kernels around corners in a stable fashion.

In [24] it was pointed out that the condition number of the discretized system matrix of the SKIE (9) grows linearly with the maximum curvature of the boundary, as illustrated in [24, Figure 2]. The modified RCIP method presented in this paper uses the preconditioned equation (33) which, for domains with corners, allows for an economical discretization, rapid convergence of iterative solvers, and high accuracy in both the solution and its gradient.

For large-scale problems, say, with boundary curves of many corners, our scheme can be coupled with fast algorithms (see, for example, [2, 8, 19, 21]) to achieve optimal complexity in both the solve phase and the evaluation phase. We plan to extend our work and combine it with proper fast algorithms to develop an accurate and efficient solver for the unsteady Stokes flow [18] in non-smooth domains.

Acknowledgements. The authors would like to thank the anonymous referees for their useful comments and suggestions, many of which have been incorporated in

the paper to enhance the presentation.

REFERENCES

[1] L. af Klinteberg and A.-K. Tornberg, *Adaptive quadrature by expansion for layer potential evaluation in two dimensions*, SIAM J. Sci. Comput., 40 (2018), pp. A1225–A1249.

[2] T. Askham, *A stabilized separation of variables method for the modified biharmonic equation*, arXiv:1710.05408 [math.NA], (2018).

[3] J. Bremer, *A fast direct solver for the integral equations of scattering theory on planar curves with corners*, J. Comput. Phys., 231 (2012), pp. 1879–1899.

[4] J. Bremer, *On the Nyström discretization of integral equations on planar curves with corners*, Appl. Comput. Harmon. Anal., 32 (2012), pp. 45–64.

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

[6] J. Bremer, V. Rokhlin, and I. Sammis, *Universal quadratures for boundary integral equations on two-dimensional domains with corners*, J. Comput. Phys., 229 (2010), pp. 8259–8280.

[7] O. P. Bruno, J. S. Ovall, and C. Turc, *A high-order integral algorithm for highly singular PDE solutions in Lipschitz domains*, Computing, 84 (2009), pp. 149–181.

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

[9] D. Colton and R. Kress, *Integral equation methods in scattering theory*, SIAM, 2013.

[10] P. Farkas, *Mathematical foundations for fast algorithms for the biharmonic equation*, PhD thesis, University of Chicago, Chicago, 1989.

[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, 4th ed., 2012.

[12] J. Helsing, *Integral equation methods for elliptic problems with boundary conditions of mixed type*, J. Comput. Phys., 228 (2009), pp. 8892–8907.

[13] ———, *Solving integral equations on piecewise smooth boundaries using the RCIP method: a tutorial*, arXiv:1207.6737 [physics.comp-ph], (2017).

[14] J. Helsing and A. Holst, *Variants of an explicit kernel-split panel-based Nyström discretization scheme for Helmholtz boundary value problems*, Adv. Comput. Math., 41 (2015), pp. 691–708.

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

[16] ———, *Elastostatic computations on aggregates of grains with sharp interfaces, corners, and triple-junctions*, Int. J. Solids. Struct., 46 (2009), pp. 4437–4450.

[17] H. V. Henderson and S. R. Searle, *On deriving the inverse of a sum of matrices*, SIAM Review, 23 (1981), pp. 53–60.

[18] S. Jiang, M. C. A. Kropinski, and B. Quaife, *Second kind integral equation formulation for the modified biharmonic equation and its applications*, J. Comput. Phys., 249 (2013), pp. 113–126.

[19] S. Jiang, B. Ren, P. Tsuji, and L. Ying, *Second kind integral equations for the first kind Dirichlet problem of the biharmonic equation in three dimensions*, J. Comput. Phys., 230 (2011), pp. 7488–7501.

[20] A. Klöckner, A. Barnett, L. Greengard, and M. O'Neil, *Quadrature by expansion: a new method for the evaluation of layer potentials*, J. Comput. Phys., 252 (2013), pp. 332–349.

[21] V. Minden, K. L. Ho, A. Damle, and L. Ying, *A recursive skeletonization factorization based on strong admissibility*, SIAM Multiscale Model. Simul., 15 (2017), pp. 768–796.

[22] D. E. Müller, *A method for solving algebraic equations using an automatic computer*, Math. Tables Aids Comput., 10 (1956), pp. 208–215.

[23] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, eds., *NIST Handbook of Mathematical Functions*, Cambridge University Press, May 2010.

[24] M. Rachh and T. Askham, *Integral equation formulation of the biharmonic Dirichlet problem*, J. Sci. Comput., 75 (2018), pp. 762–781.

[25] K. Serkh and V. Rokhlin, *On the solution of elliptic partial differential equations on regions with corners*, J. Comput. Phys., 305 (2016), pp. 150 – 171.

[26] G. Verchota, *Layer potentials and boundary value problems for Laplace's equation in Lipschitz domains*, J. Funct. Anal., 59 (1984), pp. 572–611.