Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

May the singular integrals in BEM be replaced by Zero?

by

Steffen Börm, Nico Krzebek, and Stefan Sauter

Preprint no.: 86 2003

May the singular integrals in BEM be replaced by Zero?

S. Börm

(sbo@mis.mpg.de), Max-Planck-Institut Mathematik in den Naturwissenschaften*, D-04103 Leipzig, Inselstr. 22-26, Germany*

N. Krzebek

(krzebek@amath.unizh.ch), Institut für Mathematik, Universität Zürich, Winterthurerstr 190, CH-8057 Zürich, Switzerland

S.A. Sauter *

(stas@amath.unizh.ch), Institut für Mathematik, Universität Zürich, Winterthurerstr 190, CH-8057 Zürich, Switzerland

Abstract

When approximating the singular integrals arising in the boundary element method by quadrature techniques, it is important to keep the quadrature error consistent with the discretization error in order to reach the optimal order of convergence. In classical approaches, this means that the order of the quadrature grows logarithmically in the number of degrees of freedom.

We present a quadrature scheme based on alternative representations of the singular integrands that allows us to use a *constant* quadrature order without giving up consistency.

1 Introduction

The integral equation method is an elegant tool to transform homogeneous linear boundary value problems with constant coefficients into boundary integral equations (BIE) on the boundary of the domain (see, e.g., [8], [13]). The boundary element method (BEM) is a flexible discretisation technique for solving these equations numerically.

* Corresponding author

However, the implementation of the Galerkin BEM is non-trivial since, in order to keep the computational costs moderate, the use of efficient quadrature methods for computing the entries of the system matrix is essential. Furthermore, fast operator compression techniques [9], [14], [17], [10], [18] such as panel clustering, multipole, $\mathcal{H}$-matrices, or wavelets have to be applied to avoid the storage of the n^2 matrix entries by computing instead $n \log^\kappa n$ alternative quantities for, e.g., the panel clustering algorithm. (Here and in the following n denotes the dimension of the system matrix.)

The reason for the $(\log^\kappa n)$-term in the complexity estimates for the fast compression algorithms is two-fold:

(1) The numerical computation of the nearfield entries of the system matrix, i.e., the entries lying close to the diagonal, requires quadrature schemes where the *order* has to be increased logarithmically with the number of unknowns.
(2) The expansion order for the *farfield approximations* for, e.g., the panel-clustering method, has to be increased logarithmically with the number of unknowns.

The power κ in the poly-logarithmic term typically ranges between $4-7$ and, for practical problem sizes with, say, $10^3 - 10^5$ unknowns, spoils significantly the "almost linear" computational complexity of the overall discretisation.

In this paper, we will present a numerical quadrature method which can be applied to all classical boundary integral operators corresponding to Laplace-type problems and the order m of the quadrature method may be chosen very low $m \sim 1$, independent of the number of unknowns.

We will not discuss in this paper the combination of the proposed quadrature method with panel-clustering but refer for the variable order panel-clustering method to [1], [17]. However, we emphasize that the implementation of the new quadrature method uses the same data structures as the panel-clustering method and therefore can be considered as a basic building block for incorporating this fast compression technique.

The focus of this paper lies in the presentation of the new quadrature techniques in a self-contained way which allows their straightforward implementation. We will report on the results of numerical experiments which shows that all quadrature and approximation orders can be chosen $O(1)$, independent of the number of unknowns.

The error analysis will be presented in part II of this paper.

2 Galerkin discretization of integral operators

Throughout this paper, $\Omega \subset \mathbb{R}^3$ denotes a bounded Lipschitz domain with boundary Γ and normal vector field n (oriented to the exterior of Ω). We define the Sobolev space $H^s(\Gamma)$, $s \geq 0$, in the usual way (see, e.g., [13]). Note that the range of s for which $H^s(\Gamma)$ is defined may be limited, depending on the global smoothness of the surface Γ. For $s < 0$, the spaces $H^s(\Gamma)$ are the duals of $H^{-s}(\Gamma)$.

We will consider the general integral equation

$$(\lambda I + \mathcal{K})\, u(x) := \lambda u(x) + \int_\Gamma k(x,y)\, u(y)\, ds_y = f(x), \qquad x \in \Gamma,$$

for some given scalar $\lambda \in \mathbb{R}$ kernel function k and sufficiently smooth right-hand side f. The corresponding weak form is

$$\text{Find } u \in H \text{ such that } a(u,v) := ((\lambda I + \mathcal{K})u, v) = (f,v) \quad \text{for all } v \in H. \tag{2.1}$$

Here H equals $H^\mu(\Gamma)$ or is a closed subspace of $H^\mu(\Gamma)$ for some $\mu \in \{-1/2, 0, 1/2\}$. (The bracket $(\cdot,\cdot)$ denotes the continuous extension of the $L^2(\Gamma)$ scalar product to the $H^{-\mu}(\Gamma) \times H^\mu(\Gamma)$ duality pairing.) A typical example is the hypersingular operator

$$a_H(u,v) := \frac{1}{4\pi} \int_\Gamma v(x) \int_\Gamma u(y) \frac{\partial}{\partial n_x} \frac{\partial}{\partial n_y} \frac{1}{\|x-y\|}\, ds_y\, ds_x$$

with $H = H^{1/2}(\Gamma)/\mathbb{R}$.

In the standard, conforming Galerkin method we select a subspace $\mathcal{S} \subset H$ and approximate (2.1) by seeking $u_\mathcal{S} \in \mathcal{S}$, such that

$$a(u,v) = (f,v) \qquad \text{for all } v \in \mathcal{S}. \tag{2.2}$$

In the context of the boundary element method, these subspaces are finite element spaces lifted to the surface Γ.

Definition 2.1 *(a) The* master element $\hat{\tau} \subseteq \mathbb{R}^2$ *is the open triangle with vertices* $(0,0)^\intercal$, $(0,1)^\intercal$ *and* $(1,1)^\intercal$.

(b) A set $\mathcal{G} = \{\tau_1, \tau_2, \ldots, \tau_n\}$ *consisting of open and disjoint (possibly curved) triangles in* $\mathbb{R}^3$ *such that there is a* C^∞*-diffeomorphism* $\Psi_\tau : \overline{\hat{\tau}} \to \overline{\tau}$ *for each* $\tau \in \mathcal{G}$ *is a* surface triangulation *of* Γ *if it satisfies* $\Gamma = \bigcup_{\tau \in \mathcal{T}} \overline{\tau}$ *and if the intersection* $e := \overline{\tau} \cap \overline{t}$ *of non-identical triangles* $t, \tau \in \mathcal{G}$ *is either empty, a common vertex of a common edge and, in the case that* e *is an edge, there exists an affine mapping* $\gamma : \overline{\hat{\tau}} \to \overline{\hat{\tau}}$ *such that* $\Psi_\tau^{-1}|_e = (\Psi_t \circ \gamma)^{-1}|_e$.

The stepsize of a grid is given by $h = \max_{\tau\in\mathcal{G}} \operatorname{diam}(\tau)$. For $k \in \{0,1\}$ and $p \in \mathbb{N}_0$, the finite element space $\mathcal{S}(k,p,\mathcal{G})$ is defined by $\mathcal{S}(k,p,\mathcal{G}) := \{u \in H^k(\Gamma) \mid \forall \tau \in \mathcal{G} : u \circ \Psi_\tau \in \mathbb{P}_p\}$.

We will consider boundary elements of lowest order for the discretisation of the integral operators, i.e.,

- $\mathcal{S} = \mathcal{S}(0,0,\mathcal{G})$ for the single layer operator and
- $\mathcal{S} = \mathcal{S}(1,1,\mathcal{G})/\mathbb{R}$ for the hypersingular operator.
- The boundary element space for the double layer potential operator is a subset of $\mathcal{S}(0,0,\mathcal{G})$ which contains all functions vanishing in a certain neighbourhood of the corners and edges of the surface Γ. For the detailed definition of the *finite section method* we refer to [6].

Let $(b_i)_{i=1}^n$ be the local basis of $\mathcal{S}(k,p,\mathcal{G})$. The standard Galerkin method requires us to compute the matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ given by $\mathbf{K}_{ij} := a(b_i,b_j)$ for $i,j \in \{1,\ldots,n\}$. The direct numerical realisation of this approach suffers from two bottlenecks: (a) the computation of the matrix entries requires the evaluation of singular, nearly singular and regular surface integrals over pairs of panels, (b) the system matrix is not sparse but fully populated and the computational and storage costs are at least of order n^2.

In this paper, we will only address the first point, i.e., the question how quadrature of singular and nearly-singular integrals can be performed efficiently, but our approach can also be used to construct a variant of the panel-clustering method [9] with linear complexity [1].

3 Alternative representations

In [1], the following alternative representations have been derived for the classical boundary integral operators

$$a_D(u,v) = \left(\lambda - \frac{1}{2}\right)(u,v) + \int_{\Gamma\times\Gamma} v(x)(u(y)-u(x))\frac{\partial G_1}{\partial n_y}(x,y)\,ds_y\,ds_x, \tag{3.1}$$

$$\begin{aligned} a_H(u,v) = {} & \int_\Gamma\int_\Gamma \langle \overrightarrow{\operatorname{curl}}_\Gamma u(y), \overrightarrow{\operatorname{curl}}_\Gamma v(x)\rangle \Delta_{\Gamma,xy} G_2(x,y)\,ds_y\,ds_x \\ & + \int_\Gamma\int_\Gamma \langle \overrightarrow{\operatorname{curl}}_\Gamma u(y), \overrightarrow{\operatorname{curl}}_\Gamma v(x)\rangle \langle n_y, y-x\rangle \frac{\partial G_1}{\partial n_y}(x,y)\,ds_y\,ds_x \quad \text{and} \end{aligned} \tag{3.2}$$

$$\begin{aligned} a_S(u,v) = {} & \int_\Gamma\int_\Gamma v(x)u(y)\Delta_{\Gamma,xy} G_2(x,y)\,ds_y\,ds_x \\ & + \int_\Gamma\int_\Gamma v(x)(u(y)-u(x))\langle n_y, y-x\rangle \frac{\partial G_1}{\partial n_y}(x,y)\,ds_y\,ds_x \\ & + (\rho u, v)\,. \end{aligned} \tag{3.3}$$

Here, we have used the notation

$$G_1(x,y) := \frac{1}{4\pi\|x-y\|}, \quad G_2(x,y) := \frac{\|x-y\|}{4\pi}, \quad \nabla_\Gamma u := (\nabla u^\star)|_\Gamma\,,$$
$$\nabla^t_\Gamma u := \nabla_\Gamma u - n\frac{\partial}{\partial n}u, \quad \overrightarrow{\mathrm{curl}}_\Gamma u := -n\times\nabla_\Gamma u \quad \text{and} \quad \Delta_{\Gamma,xy}G := -\langle\nabla_{\Gamma,x},\nabla^t_{\Gamma,y}\rangle G,$$

where n is the exterior unit normal vector field and $u^\star$ is a Lipschitz continuous extension of u to a tubular neighbourhood of Γ. In the case of the single layer potential, we assume that Γ is the union of q plane polygons Γ_i and define

$$\rho(x) := -\sum_{i=1}^{q}\int_{\Gamma_i}\langle n_y, x-y\rangle\frac{\partial}{\partial n_y}G_1(x,y)\,ds_y \tag{3.4}$$

(note that, due to orthogonality, $\langle n_y, x-y\rangle$ depends only on x and the polygon Γ_i, but not on the value of y).

The equations (3.1), (3.2) and (3.3) imply that we have to deal with the following kernel functions:

$$k_1(x,y) := \frac{\partial G_1}{\partial n_y}(x,y), \quad k_2(x,y) := \langle n_y, x-y\rangle\frac{\partial G_1}{\partial n_y}(x,y) \quad \text{and} \tag{3.5}$$
$$k_3(x,y) := \Delta_{\Gamma,xy}G_2(x,y). \tag{3.6}$$

4 Algorithm

4.1 Quadrature techniques for the kernels k_1 and k_2

In this section, we will recall quadrature techniques (cf. [16], [12], [7]) for integrals of the form

$$I_{\tau\times t} := \int_{\tau\times t} k(x,y)H(x,y)ds_y ds_x, \tag{4.1}$$

where $k\in\{k_1,k_2\}$ is one of the kernel functions in (3.5) and $H(x,y)$ is the combination of the basis functions as in (3.1), (3.2) and (3.3).

The quadrature technique for the kernel k_3 is different from that and will be discussed in Subsection 4.2.

For the quadrature, we have to distinguish four cases:

(1) Identical panels: $\tau = t$
(2) Common edge: $\overline{\tau}\cap\overline{t}$ is a common edge,

(3) Common point: $\overline{\tau} \cap \overline{t}$ is a common point,
(4) Positive distance: $\operatorname{dist}(\tau, t) > 0$.

The integration method in the singular cases (1-3) consists of two steps: (i) the application of regularising coordinate transforms and (ii) numerical quadrature based on tensor Gauß formulae.

4.1.1 Regularizing coordinate transforms

Let $\tau, t \in \mathcal{G}$ be triangles. By definition, there are C^∞-diffeomorphisms $\Psi_\tau : \hat{\tau} \to \tau$ and $\Psi_t : \hat{\tau} \to t$ which map the unit triangle $\hat{\tau}$ onto τ and t, respectively.

This implies

$$I_{\tau\times t} = \int_{\hat{\tau}} \int_{\hat{\tau}} \hat{K}(\hat{x}, \hat{y})\, d\hat{y}\, d\hat{x}$$

with $\hat{K}(\hat{x}, \hat{y}) = g_\tau(\hat{x}) g_t(\hat{y})\, k(\Psi_\tau(\hat{x}), \Psi_t(\hat{y}))\, H(\Psi_\tau(\hat{x}), \Psi_t(\hat{y}))$ for surface elements

$$g_\tau(\hat{x}) = \sqrt{\det(D\Psi_\tau(\hat{x})^\top D\Psi_\tau(\hat{x}))} \quad \text{and} \quad g_t(\hat{x}) = \sqrt{\det(D\Psi_t(\hat{x})^\top D\Psi_t(\hat{x}))}.$$

4.1.1.1 Identical Panels We assume that $\tau = t$. Then $\Psi_\tau = \Psi_t$ and $g_\tau = g_t$ hold.

The following coordinate transforms are taken from [7], [15]. We have

$$\begin{aligned} I_{\tau\times t} = \int_{(0,1)^4} \xi^3 \eta_1^2 \eta_2 \Bigg\{ & \hat{K}\left(\xi \begin{pmatrix} 1 \\ 1-\eta_1+\eta_1\eta_2 \\ 1-\eta_1\eta_2\eta_3 \\ 1-\eta_1 \end{pmatrix}\right) + \hat{K}\left(\xi \begin{pmatrix} 1-\eta_1\eta_2\eta_3 \\ 1-\eta_1 \\ 1 \\ 1-\eta_1+\eta_1\eta_2 \end{pmatrix}\right) \\ & + \hat{K}\left(\xi \begin{pmatrix} 1 \\ \eta_1(1-\eta_2+\eta_2\eta_3) \\ 1-\eta_1\eta_2 \\ \eta_1(1-\eta_2) \end{pmatrix}\right) + \hat{K}\left(\xi \begin{pmatrix} 1-\eta_1\eta_2 \\ \eta_1(1-\eta_2) \\ 1 \\ \eta_1(1-\eta_2+\eta_2\eta_3) \end{pmatrix}\right) \\ & + \hat{K}\left(\xi \begin{pmatrix} 1-\eta_1\eta_2\eta_3 \\ \eta_1(1-\eta_2\eta_3) \\ 1 \\ \eta_1(1-\eta_2) \end{pmatrix}\right) + \hat{K}\left(\xi \begin{pmatrix} 1 \\ \eta_1(1-\eta_2) \\ 1-\eta_1\eta_2\eta_3 \\ \eta_1(1-\eta_2\eta_3) \end{pmatrix}\right) \Bigg\} d\eta_1 d\eta_2 d\eta_3 d\xi. \end{aligned} \tag{4.2}$$

Example 4.1

a. Kernel function k_1.

The kernel function k_1 appears only in the context of the double layer potential where we used piecewise constant ansatz functions. Hence, we have $u(x) - u(y) \equiv 0$ on $\tau = t$ and the integral (4.1) vanishes in the case of identical panels.

b. Kernel function k_2.

In the case of the single layer potential the kernel k_2 is multiplied by the factor $\{u(y) - u(x)\}$ and the integral (4.1) vanishes in the case of identical panels as in (a).

For the hypersingular operator, we distinguish between two cases.

i. If the curved surface is approximated by plane triangles the kernel k_2 is zero on $\tau \times \tau$ since n_y is orthogonal to $y - x$.

ii. For τ, being a curved triangle, we obtain

$$g_\tau \left(\overrightarrow{\operatorname{curl} u}\right) \circ \Psi_\tau = J_\tau \hat{\nabla}^\perp \hat{u} \tag{4.3}$$

with $\hat{u} = u \circ \Psi_\tau$, $J_\tau := \left[\hat{\partial}_1 \Psi_\tau, \hat{\partial}_2 \Psi_\tau\right]$, and $\hat{\nabla}^\perp := \left(\hat{\partial}_2, -\hat{\partial}_1\right)$. Hence,

$$\hat{K}(\hat{x}, \hat{y}) = -\left\langle J_\tau(\hat{x}) \left(\hat{\nabla}^\perp \hat{v}\right)(\hat{x}), J_\tau(\hat{y}) \left(\hat{\nabla}^\perp \hat{u}\right)(\hat{y}) \right\rangle \frac{\left\langle \hat{n}_{\hat{y}}, \Psi_\tau(\hat{y}) - \Psi_\tau(\hat{x}) \right\rangle^2}{4\pi \left\| \Psi_\tau(\hat{y}) - \Psi_\tau(\hat{x}) \right\|^3}, \tag{4.4}$$

where $\hat{n}_{\hat{y}} := \left(\hat{\partial}_1 \Psi_\tau \times \hat{\partial}_2 \Psi_\tau\right) / \left\| \hat{\partial}_1 \Psi_\tau \times \hat{\partial}_2 \Psi_\tau \right\|$.

4.1.1.2 Common edge We assume that τ, t have a common edge. Without loss of generality we may assume that $\Psi_\tau(\xi, 0) = \Psi_t(\xi, 0)$ holds for $\xi \in [0, 1]$ (in an implementation this property has to be ensured by locally adapting the mappings Ψ_τ and Ψ_t).

The following coordinate transforms are taken from [7], [15]. We have

$$\begin{aligned} I_{\tau\times t} = \int_{(0,1)^4} & \xi^3 \eta_1^2 \hat{K} \begin{pmatrix} \xi \\ \xi\eta_1\eta_3 \\ \xi(1-\eta_1\eta_2) \\ \xi\eta_1(1-\eta_2) \end{pmatrix} + \xi^3 \eta_1^2 \eta_2 \left\{ \hat{K} \begin{pmatrix} \xi \\ \xi\eta_1 \\ \xi(1-\eta_1\eta_2\eta_3) \\ \xi\eta_1\eta_2(1-\eta_3) \end{pmatrix} \right. \\ & \left. + \hat{K} \begin{pmatrix} \xi(1-\eta_1\eta_2) \\ \xi\eta_1(1-\eta_2) \\ \xi \\ \xi\eta_1\eta_2\eta_3 \end{pmatrix} + \hat{K} \begin{pmatrix} \xi(1-\eta_1\eta_2\eta_3) \\ \xi\eta_1\eta_2(1-\eta_3) \\ \xi \\ \xi\eta_1 \end{pmatrix} + \hat{K} \begin{pmatrix} \xi(1-\eta_1\eta_2\eta_3) \\ \xi\eta_1(1-\eta_2\eta_3) \\ \xi \\ \xi\eta_1\eta_2 \end{pmatrix} \right\} d\eta d\xi. \end{aligned} \tag{4.5}$$

4.1.1.3 Common Point We assume that τ, t have a common point. Without loss of generality we may assume that $\Psi_\tau(0,0) = \Psi_t(0,0)$ holds.

The following coordinate transforms are taken from [7], [15]. We have

$$I_{\tau\times t} = \int_{(0,1)^4} \xi^3\eta_2 \left\{ \hat{K}\left(\xi, \xi\eta_1, \xi\eta_2, \xi\eta_2\eta_3\right) + \hat{K}\left(\xi\eta_2, \xi\eta_2\eta_3, \xi, \xi\eta_1\right)\right\} d\eta d\xi. \quad (4.6)$$

4.1.1.4 Positive distance In the case of positive distance $\operatorname{dist}(\tau, t) > 0$, the integrand is regular. The transformation to the reference element and application of Duffy coordinates results in

$$I_{\tau\times t} = \int_{(0,1)^4} \xi_1\xi_2 \hat{K}\left(\xi_1, \xi_1\eta_1, \xi_2, \xi_2\eta_2\right) d\eta d\xi. \quad (4.7)$$

4.1.2 Numerical quadrature

In [16], [12], [7], [18], [15], it was shown that all integrands in (4.2), (4.5), (4.6), and (4.7) are analytic.

As numerical quadrature scheme, tensorised Gauß-Legendre formulae (scaled to the interval $(0,1)$) can be employed and will converge exponentially. We denote by $Q^{(m)}_{\tau\times t}$ the quadrature approximation to the general integral $I_{\tau\times t}$ with m Gaußpoints per coordinate direction. The required orders will be derived from the error analysis which will be presented in Part II of this paper. The numerical experiments in Section 5 indicate the *minimal* number of Gauß points such that the optimal convergence order is preserved.

4.1.3 Efficient implementation

In this subsection, we give some remarks concerning the efficient implementation of the quadrature methods and start with the double layer potential.

The integral in (3.1) can be written as the difference

$$\int_{\Gamma\times\Gamma} v(x)\left(u(y)\frac{\partial G_1}{\partial n_y}(x,y)\, ds_y\right) ds_x - \int_\Gamma v(x)u(x)\left(\int_\Gamma \frac{\partial G_1}{\partial n_y}(x,y)\, ds_y\right) ds_x. \quad (4.8)$$

Let $\mathbf{K} = (\mathbf{K}_{\tau t})_{\tau,t\in\mathcal{G}}$ denote the stiffness matrix of the double layer potential operator in the standard representation

$$\mathbf{K}_{\tau t} := \int_\tau \int_t \frac{\partial G_1}{\partial n_y}(x,y)\, ds_y\, ds_x \qquad \text{for all } \tau, t \in \mathcal{G}.$$

Note that the quadrature approximation $\tilde{\mathbf{K}}$ is computed via our low order quadrature scheme which would be too inaccurate when being used within the standard representation.

The quadrature approximation $\tilde{\mathbf{D}}$ for the difference of integral operators in (4.8) may be expressed as $\tilde{\mathbf{D}} = \tilde{\mathbf{K}} - \mathrm{diag}(\tilde{\mathbf{K}}\mathbf{1})$, where the non-trivial entries of the diagonal matrix $\mathrm{diag}(\tilde{\mathbf{K}}\mathbf{1})$ are given by the row sums of $\tilde{\mathbf{K}}$. For the implementation, this means that we compute a low-order quadrature approximation of the standard double layer potential, determine the row sums, and then modify the diagonal entries in order to guarantee that the spherical angle condition $\tilde{\mathbf{D}}\mathbf{1} = 0$ holds.

For the single layer operator, the remarks for the double layer potential apply verbatim to the second term in (3.3). Note that the spherical angle ρ can be computed analytically for piecewise flat surfaces (see, e.g., [11]).

4.2 Quadrature techniques for the kernel k_3

Our goal is to approximate the bilinear form

$$a_3(u,v) = \int_\Gamma \int_\Gamma v(x)u(y)\Delta_{\Gamma,xy}G_2(x,y)\,ds_y\,ds_x, \tag{4.9}$$

where

$$G_2(x,y) = \frac{\|x-y\|}{4\pi} \quad \text{and} \quad \Delta_{\Gamma,xy} = -\langle \nabla_x, \nabla_y^t \rangle.$$

The quadrature for the kernel k_3 is based on polynomial approximations of G_2 and the application of the operator $\Delta_{\Gamma,xy}$ to these polynomials. Therefore, it is essential that the the approximation is globally Lipschitz-continuous. The latter property is obtained as follows: Let $(x^\iota)_{\iota\in\mathcal{N}}$ denote the vertices of the triangles in $\mathcal{G}$ and $(\chi^\iota)_{\iota\in\mathcal{N}}$ the corresponding continuous, linear nodal basis (i.e., the well-known "hat functions"). These globally Lipschitz-continuous functions form a partition of unity that can be used to combine local interpolants.

4.2.1 Multidimensional Chebyshev interpolation

The polynomial approximation of G_2 is based on polynomial interpolation. Let $\mathcal{I}_m$ be an m-th order interpolation operator on $[-1,1]$, e.g., the Chebyshev interpolation operator, and let $(x_i)_{i=0}^m$ be the corresponding interpolation points and $(\mathcal{L}_i)_{i=0}^m$ the corresponding Lagrange polynomials. For each interval $J = [a,b]$, the transformed interpolation operator is defined by

$$\mathcal{I}_m^J[u] := \sum_{i=0}^m u(x_i^J)\mathcal{L}_i^J,$$

where

$$x_i^J := \frac{b+a}{2} + \frac{b-a}{2} x_i, \qquad \mathcal{L}_i^J(x) := \prod_{j=0, j\neq i}^{m} \frac{x - x_j^J}{x_i^J - x_j^J}$$

are the properly scaled interpolation points and Lagrange polynomials.

In order to construct a multi-dimensional interpolation operator, we fix a minimal axis-parallel box $B^\iota = J_1^\iota \times J_2^\iota \times J_3^\iota \subseteq \mathbb{R}^3$ satisfying $\operatorname{supp} \chi^\iota \subseteq B^\iota$. The tensor-product interpolation operator is then defined by

$$\mathcal{I}_m^\iota := \mathcal{I}_m^{J_1^\iota} \otimes \mathcal{I}_m^{J_2^\iota} \otimes \mathcal{I}_m^{J_3^\iota}$$

and can be evaluated by

$$\mathcal{I}_m^\iota[u] := \sum_{\nu \in \mathfrak{K}} u(x_\nu^\iota) \mathcal{L}_\nu^\iota$$

for $x_\nu^\iota := (x_{\nu_1}^{J_1^\iota}, x_{\nu_2}^{J_2^\iota}, x_{\nu_3}^{J_3^\iota})$, $\mathcal{L}_\nu^\iota(x) := \mathcal{L}_{\nu_1}^{J_1^\iota}(x_1) \mathcal{L}_{\nu_2}^{J_2^\iota}(x_2) \mathcal{L}_{\nu_3}^{J_3^\iota}(x_3)$ and $\mathfrak{K} := \{0, \ldots, m\}^3$.

4.2.2 Globally continuous approximation

For the product set $\Gamma \times \Gamma$, we simply use tensorised version of the hat functions and interpolation operators: $(\chi^\kappa \otimes \chi^\iota)_{\iota,\kappa \in \mathcal{N}}$ and $(\mathcal{I}_m^\kappa \otimes \mathcal{I}_m^\iota)_{\iota,\kappa \in \mathcal{N}}$. Using these, we introduce the global approximation

$$\tilde{G}_2 := \sum_{\iota \in \mathcal{N}} \sum_{\kappa \in \mathcal{N}} \chi^\kappa \otimes \chi^\iota \; \mathcal{I}_m^\kappa \otimes \mathcal{I}_m^\iota[G_2] \tag{4.10}$$

of G_2. Note that the Lipschitz continuity of the functions χ^ι, χ^κ is inherited by $\tilde{G}_2$.

Next, we apply the differential operator $\Delta_{\Gamma,xy} = -\langle \nabla_{\Gamma,x} \nabla_{\Gamma,y}^t \rangle$ to the approximation $\tilde{G}_2$:

$$\begin{aligned}
\Delta_{\Gamma,xy} \tilde{G}_2 &= \sum_{\iota \in \mathcal{N}} \sum_{\kappa \in \mathcal{N}} \sum_{\nu \in \mathfrak{K}} \sum_{\mu \in \mathfrak{K}} G_2(x_\nu^\kappa, x_\mu^\iota) \Delta_{\Gamma,xy} (\chi^\kappa \otimes \chi^\iota \; \mathcal{L}_\nu^\kappa \otimes \mathcal{L}_\mu^\iota) \\
&= - \sum_{\iota,\kappa \in \mathcal{N}} \sum_{\nu,\mu \in \mathfrak{K}} G_2(x_\nu^\kappa, x_\mu^\iota) \langle \nabla_\Gamma (\chi^\kappa \mathcal{L}_\nu^\kappa), \nabla_\Gamma^t (\chi^\iota \mathcal{L}_\mu^\iota) \rangle.
\end{aligned}$$

The approximation of $a_3(\cdot,\cdot)$ is defined by replacing G_2 by $\tilde{G}_2$

$$\begin{aligned}
\tilde{a}_3(u,v) &:= \int_\Gamma \int_\Gamma v(x) u(y) (\Delta_{\Gamma,xy} \tilde{G}_2)(x,y) \, dy \, dx \\
&= \sum_{\iota,\kappa \in \mathcal{N}} \sum_{\nu,\mu \in \mathfrak{K}} G_2(x_\nu^\kappa, x_\mu^\iota) \left\langle \mathcal{V}_\nu^\kappa(v), \mathcal{W}_\mu^\iota(u) \right\rangle,
\end{aligned} \tag{4.11}$$

where the functionals $\mathcal{V}_\nu^\iota(u)$, $\mathcal{W}_\nu^\iota(u) \in \mathbb{R}^3$ are defined, for all $\iota \in \mathcal{N}$ and all

$\nu \in \mathfrak{K}$, by

$$\mathcal{V}_\nu^\iota(u) := \int_\Gamma \nabla_\Gamma(\chi^\iota \mathcal{L}_\nu^\iota)(x)u(x)\,dx, \qquad \mathcal{W}_\nu^\iota(u) := \int_\Gamma \nabla_\Gamma^t(\chi^\iota \mathcal{L}_\nu^\iota)(x)u(x)\,dx.$$

Using this representation, we can approximate matrix entries $A_{ij}^3 = a_3(b_i, b_j)$ by

$$\tilde{A}_{i,j}^3 := \tilde{a}_3(b_i, b_j) = \sum_{\iota,\kappa\in\mathcal{N}} \sum_{\nu,\mu\in\mathfrak{K}} G_2(x_\nu^\kappa, x_\mu^\iota) \left\langle \mathcal{V}_\nu^\kappa(b_j), \mathcal{W}_\mu^\iota(b_i) \right\rangle. \tag{4.12}$$

4.2.3 Efficient implementation

To obtain an efficient implementation, we first compute the quantities

$$\begin{aligned} V_{\nu,i}^\iota &:= \sum_{\tau\subset\operatorname{supp} b_i\cap\operatorname{supp}\chi^\iota} \int_\tau \nabla_\Gamma(\chi^\iota \mathcal{L}_\nu^\iota)(x)b_i(x)\,dx = \mathcal{V}_\nu^\iota(b_i), \\ W_{\nu,i}^\iota &:= \sum_{\tau\subset\operatorname{supp} b_i\cap\operatorname{supp}\chi^\iota} \int_\tau \nabla_\Gamma^t(\chi^\iota \mathcal{L}_\nu^\iota)(x)b_i(x)\,dx = \mathcal{W}_\nu^\iota(b_i). \end{aligned}$$

For flat panels, these quantities can be computed analytically. In our applications, the interpolation order $m = O(1)$ is bounded independent of the number of unknowns. Thus, a moderate order of quadrature will give us sufficiently accurate values for these auxiliary values, and (4.12) can be written in the form

$$\tilde{A}_{i,j}^3 = \sum_{\iota,\kappa\in\mathcal{N}} \sum_{\nu,\mu\in\mathfrak{K}} G_2(x_\nu^\kappa, x_\mu^\iota) \left\langle V_{\nu,j}^\kappa, W_{\mu,i}^\iota \right\rangle.$$

The algorithmic formulation is as follows.

```
for ι, κ ∈ 𝒩 do
  for ν, μ ∈ 𝔎 do begin
    s := G_2(x_ν^κ, x_μ^ι);
    for i ∈ {i : supp(b_i) ∩ supp(χ^ι) ≠ ∅} do
      for j ∈ {j : supp(b_j) ∩ supp(χ^κ) ≠ ∅} do
        Ã^3_{i,j} := Ã^3_{i,j} + s V^κ_{ν,j} W^ι_{μ,i}
  end
```

In this algorithm, the number of kernel evaluations is $(\#\mathcal{N})^2(\#\mathfrak{K})^2$, and this is optimal for our approximation scheme. We emphasize that the interpolation order m has not to be increased logarithmically with the number of unknowns but is a small constant $m \sim 2$.

5 Numerical experiments

We have performed numerical experiments for all three types of boundary integral equations related to Laplace's equation. The ansatz and test spaces are chosen as described in Section 2.

5.1 Double layer potential

For our numerical tests, we have chosen the unit cube $\Omega = (0,1)^3$ and solve the exterior Neumann problem

$$-\Delta u = 0 \text{ in } \mathbb{R}^3 \setminus \overline{\Omega}, \qquad \frac{\partial u}{\partial n} = g_N \text{ on } \partial\Omega \tag{5.1}$$

with $g_N := \partial/\partial n \|x - x_0\|^{-1}$ and $x_0 = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)^\intercal$. The exact solution is $u(x) = \|x - x_0\|^{-1}$. Green's representation formula leads to the boundary integral equation

$$\left(\frac{1}{2}I + K\right) u = V\frac{\partial u}{\partial n} \tag{5.2}$$

almost everywhere on Γ. The operator V on the right-hand side was discretised by the classical formulation.

In Table 5.1, we have listed the relative errors $u - \tilde{u}_\ell$ in the $L^2(\Gamma)$-norm, where $\tilde{u}_\ell$ is the Galerkin solution by using our quadrature methods. The mesh size is halved with each refinement level ℓ and the corresponding boundary element space is denoted by $\mathcal{S}_\ell$. We denote by *DLPalt* the alternative representation, and the pair (m_1, m_2) contains the number m_1 of Gauß points in each space direction for the singular resp. m_2 for the regular case. The number of Gauß points was chosen such that the optimal rate of convergence is preserved over the tested range of refinements. Note, that the discretisation of the classical representation fails to converge with the expected linear rate of convergence if the quadrature order is chosen constant independent of the refinement level. The table gives the relative L^2-errors on level ℓ

$$e_\ell = \frac{\|u - \tilde{u}_\ell\|_{L^2(\Gamma)}}{\|u\|_{L^2(\Gamma)}}$$

and the corresponding contraction rates $\rho_\ell := e_{\ell-1}/e_\ell$.

We see that we can replace the nearfield integrals by zero and approximate the farfield integrals by one-point quadrature while keeping the optimal linear convergence order for piecewise constant approximations of the double layer potential.

Level	n	$DLPalt(0,1)$	ρ_l
1	12	0.30704	—
2	48	0.14851	2.06
3	192	0.09517	1.56
4	768	0.04883	1.94
5	3072	0.02499	1.95
6	12288	0.01135	2.20

Table 1
Relative L^2-errors for the Galerkin solution with constant order numerical quadrature for the DLP.

5.2 Single layer potential

We have solved the boundary integral equation (5.2) on the boundary of the unit cube for the unknown Neumann data when the Dirichlet data of u is given.

In Table 5.2 the triple (m_1, m_2, m_3) contains the number m_1 of quadrature points (per space dimension) for the singular integrals and the kernel k_2, m_2 denotes the number of quadrature points for the regular integrals and m_3 denotes the order of interpolation which is used for the approximation of the generator function G_2.

Level	n	$SLPalt(2,4,2)$	ρ_l	$SLPalt(2,4,1)$	ρ_l
1	12	0.27908	—	0.38572	—
2	48	0.43442	0.64	3.75139	0.10
3	192	0.33033	1.32	0.17385	21.58
4	768	0.13756	2.40	0.24419	0.71
5	3072	0.05031	2.73	0.15471	1.58
6	12288				

Table 2
Relative errors in the energy norm of the Galerkin solution with constant order quadrature and interpolation for the SLP.

We see that the minimal order of interpolation should be chosen equal to 2, while the order 1 leads to reduced convergence rates.

5.2.1 Hypersingular operator

For the hypersingular operator, we employed Green's formula to transfer the exterior Neumann problem onto the boundary Γ to obtain the boundary integral equation

$$Wu = \left(\frac{1}{2}I + K'\right)\frac{\partial u}{\partial n}. \tag{5.3}$$

Here, K' denotes the adjoint double layer potential. According to our smoothness assumption for the domain Ω, we choose Γ to be the boundary of the unit sphere. The solution is chosen to be $u(x) = x_1^2 + 2x_2^2 - 3x_3^2$ and the right-hand side accordingly.

In Table 3 we compare the relative errors with different parameter choices. The notation is the same as for the single layer potential. The optimal convergence rate of $O(h^{3/2})$ is related to the expected contraction rate $2^{3/2} \approx 2.828$ with increasing refinement level.

Level	#panels	n	$Walt(1,4,2)$	ρ_l	$Walt(1,4,1)$	ρ_l
1	8	6	1.02414	—	3.55884	—
2	32	18	2.98031	0.34	1.76776	2.01
3	128	66	0.704344	4.23	0.549386	3.22
4	512	258	0.180291	3.90	0.157188	3.49
5	2048	1026	0.053908	3.34	0.049123	3.20
6	8192	4098	0.018803	2.86	0.022313	2.20

Table 3
Relative errors in the energy norm of the Galerkin solution with constant order quadrature and interpolation for the hypersingular operator.

We see that the interpolation with polynomials of degree $m = 2$ is sufficient such that the Galerkin with quadrature converges with optimal order. For $m = 1$, the numerical results show that the converges order becomes suboptimal with increasing refinement level.

References

[1] S. Börm and S. Sauter. Classical BEM with linear complexity Technical report 15-2003, Universität Zürich.

[2] P. Ciarlet. *The finite element method for elliptic problems.* North-Holland, 1987.

[3] M. Costabel. Boundary integral operators on Lipschitz domains: Elementary results. *Siam, J. Math. Anal.*, 19:613–626, 1988.

[4] P. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press, 1984.

[5] R. DeVore and G. Lorentz. *Constructive Approximation*. Springer-Verlag, New York, 1993.

[6] J. Elschner. The Double Layer Potential Operator over Polyhedral Domains I: Solvability in Weighted Sobolev Spaces. *Appl. Anal.*, 45:117–134, 1992.

[7] S. Erichsen and S. Sauter. Efficient automatic quadrature in 3-d Galerkin BEM. *Comp. Meth. Appl. Mech. Eng.*, 157:215–224, 1998.

[8] W. Hackbusch. *Integral Equations*. ISNM. Birkhäuser, 1995.

[9] W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering *Numerische Mathematik*, 54:463–491, 1989

[10] W. Hackbusch and B.N. Khoromskij: *A sparse $\mathcal{H}$-matrix arithmetic. Part II: Application to multi-dimensional problems*. Computing **64** (2000), 21-47.

[11] F. T. Johnson. A General Panel Method for the Analysis and Design of Arbitrary Configurations in Incompressible Flows. NASA Report 3079, NASA Ames Research Center, USA, 1980.

[12] W. Hackbusch and S. Sauter. On the Efficient Use of the Galerkin Method to Solve Fredholm Integral Equations. *Applications of Mathematics*, 38(4-5):301–322, 1993.

[13] W. McLean. *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge, Univ. Press, 2000.

[14] V. Rokhlin. Rapid solutions of integral equations of classical potential theory. *Journal of Computational Physics*, 60(2):187–207, 1985.

[15] S. Sauter and C. Schwab. *Randelementmethoden*. Teubner, Leipzig, to appear in 2003.

[16] S. A. Sauter. *Über die effiziente Verwendung des Galerkinverfahrens zur Lösung Fredholmscher Integralgleichungen*. PhD thesis, Inst. f. Prakt. Math., Universität Kiel, 1992.

[17] S. Sauter. Variable order panel clustering. *Computing*, 64:223–261, 2000.

[18] T.v.Petersdorff and C. Schwab. Fully Discrete Multiscale Galerkin BEM. In W. Dahmen, P. Kurdila, and P. Oswald, editors, *Multiresolution Analysis and Partial Differential Equations*, pages 287–346, New York, 1997. Academic Press.