Journal of Computational and Applied Mathematics 487 (2026) 117713

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A stabilized local integral method using RBFs for Helmholtz problems arising from electrodynamics

L. Ponzellini Marinelli [a,b,c,*], L. Raviola [a]

[a] Facultad de Ciencias Exactas, Ingeniería y Agrimensura, Universidad Nacional de Rosario, Argentina
[b] Pontificia Universidad Católica Argentina, Rosario, Argentina
[c] Instituto Física Rosario, UNR-CONICET, Argentina

ARTICLE INFO

Keywords:
Meshless method
Radial basis functions
Local integral method
Helmholtz problems

ABSTRACT

In this paper we present the Stabilized Local Boundary Domain Integral Method (SLBDIM), which is a local integral boundary element technique with stable computation based on Radial Basis Function (RBF) approximations, applied to Helmholtz problems. We present numerical results for small shape parameters of the RBF that stabilize the errors. We also discuss accuracy, conditioning and comparisons with other methods for various case studies. The virtues of the method are demonstrated through its application to problems arising in wave chaos, acoustics, and dielectric microresonators. The SLBDIM is computationally efficient and well suited to geometries with arbitrarily shaped domains, including those with corners.

1. Introduction and motivation

The Helmholtz equation plays a key role in scientific disciplines such as physics, engineering and geophysics, and is particularly important in optical problems. For example, dielectric microresonators have attracted interest in recent decades due to their technological applications in microlasers, as well for their connection with quantum billiards and wave chaos [1]. This equation can be solved analytically by separation of variables in special geometries, but for generic domains numerical methods are generally required.

Many numerical techniques can be used to solve the Helmholtz equation, including the finite element method (FEM), the finite volume method, the boundary element method (BEM) [2], spectral methods [3], transform methods [4] and wavelet methods [5]. However, several of these require the construction of a specific mesh or refinement in order to efficiently solve certain numerical problems on non-trivial geometries.

In the BEM, the partial differential equations (PDEs) that describe the problem are transformed into a boundary integral equation, using Green's identities and applying the integral formulation to points distributed in the domain [6,7]. Many local integral methods are based on an integral formulation on small, highly overlapping stencils with local interpolations.

In recent decades, radial basis function (RBF) methods have become an extremely powerful tool for interpolation in sets of scattered nodes and for numerical approximation of PDEs in non-trivial geometries. Many modern books deal with the theory, implementation and application of these methods [8–10]. One advantage is that, when creating the distribution nodes, it is possible to achieve local refinement in critical areas depending on the specific problem [11].

* Corresponding author.
E-mail addresses: ponzellini@ifir-conicet.gov.ar (L. Ponzellini Marinelli), raviola@fceia.unr.edu.ar (L. Raviola).

https://doi.org/10.1016/j.cam.2026.117713
Received 23 September 2025; Received in revised form 30 March 2026
Available online 15 April 2026

Several articles deal with the numerical solution of Helmholtz problems using global RBF methods [12–15]. The application of a meshless method for the Helmholtz equation on arbitrary 2D domains in high-frecuency regimes has been studied in [16]. For localized methods, such as the radial basis function-finite difference (RBF-FD) method to solve the Helmholtz equation, we recommend the work [17]. Another local weak-based meshless method has been studied in [18] to determine the optimal radius of subdomains in the context of a local integral equation discretization.

Exponential convergence can be demonstrated using infinitely differentiable RBFs, such as Gaussian RBFs. However, a practical obstacle is the ill-conditioning of the interpolation matrix when the shape parameter ε tends to zero. While it is true that reducing this parameter improves the interpolation accuracy of the method considerably, the numerical conditioning of the problem worsens when it is solved using a direct numerical method. Stable algorithms, such as RBF-QR used in the present work, overcome this conditioning issue.

In this paper we present the Stabilized Localized Boundary-Domain Integral Method (SLBDIM) [19] in the context of Helmholtz-type equations. SLBDIM is a new stable integral local numerical method for the approximation of elliptic-type PDE solutions to Boundary Value Problems (BVP) in two dimensions using local interpolations with RBF for small values of $\varepsilon > 0$. This technique is a combination of meshless methods, local integral formulations and boundary elements in multi-domains. It is independent of a structured mesh and requires only an unstructured distribution of nodes in the domain Ω and on its boundary $\Gamma = \partial\Omega$, allowing it to handle complex geometries. For local interpolations, the Gaussian RBFs $\varphi(r) = e^{-(\varepsilon r)^2}$ are used when $\varepsilon \to 0$, ensuring stability in local interpolations.

The numerical results presented are for a small shape parameter which stabilizes the errors. Comparisons with other methods are also discussed in several cases. We demonstrate that the method is computationally efficient and suitable for non-trivial geometries. Specifically, we solve differential problems with Dirichlet-type boundary conditions over square and irregular domains with quasi-uniform point distributions.

2. The stabilized localized boundary domain integral method for Helmholtz equations

2.1. Problem description and local integral method

We consider the following boundary value problem (BVP) on an open, bounded and simply connected domain $\Omega \subset \mathbb{R}^2$,

$$(BVP)\quad \begin{cases} \mathcal{L}[u](\mathbf{x}) = f(\mathbf{x}), & \mathbf{x} \in \Omega, \quad (1a)\\ \mathcal{B}[u](\mathbf{x}) = g(\mathbf{x}), & \mathbf{x} \in \Gamma = \partial\Omega, \quad (1b) \end{cases}$$

where $\mathcal{L}[\ .\ ] = \Delta + \lambda$ is a Helmholtz-type elliptic differential operator, $\Delta = \frac{\partial}{\partial x^2} + \frac{\partial}{\partial y^2}$ is the Laplacian, $\lambda \in \mathbb{R}$ (if $\lambda = k^2 > 0$, k is the wavenumber) and $f(\mathbf{x})$ is a smooth source term. $\mathcal{B}[\ .\ ]$ is the boundary condition operator.

The boundary conditions (BC) are Dirichlet over Γ_1, Neumann over Γ_2 or mixed over $\Gamma = \Gamma_1 \cup \Gamma_2$ and $\Gamma_1 \cap \Gamma_2 = \varnothing$,

$$u \begin{cases} (\mathbf{x}) = g_1(\mathbf{x}), & \mathbf{x} \in \Gamma_1, \quad (2a)\\ \dfrac{\partial u(\mathbf{x})}{\partial n} = g_2(\mathbf{x}), & \mathbf{x} \in \Gamma_2, \quad (2b) \end{cases}$$

where g_1 and g_2 are known data and $\frac{\partial u(\mathbf{x})}{\partial n}$ is the outward normal derivative of the unknown field u. The PDE (1a) can be written as

$$\Delta u(\mathbf{x}) = f(\mathbf{x}) - \lambda u(\mathbf{x}) = b(\mathbf{x}, u(\mathbf{x})), \quad (3)$$

where $u(\mathbf{x})$ is the potential at point $\mathbf{x} \in \Omega$ and b depends on the spatial variable and the unknown field.

Applying Green's second identity for u satisfying (3), we obtain

$$u(\boldsymbol{\xi}) = \int_\Omega u^*(\mathbf{x}, \boldsymbol{\xi}) b \, d\Omega_{\mathbf{x}} - \oint_\Gamma \left[u^*(\mathbf{x}, \boldsymbol{\xi}) \frac{\partial u(\mathbf{x})}{\partial n} - u(\mathbf{x}) \frac{u^*(\mathbf{x}, \boldsymbol{\xi})}{\partial n} \right] d\Gamma_{\mathbf{x}}, \quad (4)$$

where $\boldsymbol{\xi} \in \Omega$ is an interior point and $u^*(\mathbf{x}, \boldsymbol{\xi})$ is the fundamental Laplacian solution.

From Eq. (4) we have a formula for the integral representation of the PDE over a subregion Ω_i with boundary Γ_i. The collocation at the interior point $\boldsymbol{\xi}$ is obtained as before from the fundamental solution and Green's second identity,

$$u(\boldsymbol{\xi}) = \int_{\Gamma_i} q^*(\mathbf{x}, \boldsymbol{\xi}) u(\mathbf{x}) d\Gamma_{\mathbf{x}} - \int_{\Gamma_i} u^*(\mathbf{x}, \boldsymbol{\xi}) q(\mathbf{x}) d\Gamma_{\mathbf{x}} + \int_{\Omega_i} u^*(\mathbf{x}, \boldsymbol{\xi}) b \, d\Omega_{\mathbf{x}}, \quad (5)$$

where $q = \frac{\partial u}{\partial n}$ and $q^* = \frac{\partial u^*}{\partial n}$ are the the normal derivatives of the unknown field and the fundamental solution respectively.

Using the well-known Green-Dirichlet function $G(\mathbf{x}, \boldsymbol{\xi})$ and its normal derivative $Q(\mathbf{x}, \boldsymbol{\xi})$ in Eq. (5), see reference [20], we obtain a new integral formulation of the form

$$u(\boldsymbol{\xi}) = \int_{\Gamma_i} Q(\mathbf{x}, \boldsymbol{\xi}) u(\mathbf{x}) \, d\Gamma_{\mathbf{x}} + \int_{\Omega_i} G(\mathbf{x}, \boldsymbol{\xi}) b \, d\Omega_{\mathbf{x}}, \quad (6)$$

since the integral over Γ_i with G in (5) vanishes because its value is zero.

Furthermore, if we express the non-homogeneous term b of the form

$$b(\mathbf{x}, u(\mathbf{x})) = f(\mathbf{x}) - \lambda u(\mathbf{x}), \quad (7)$$

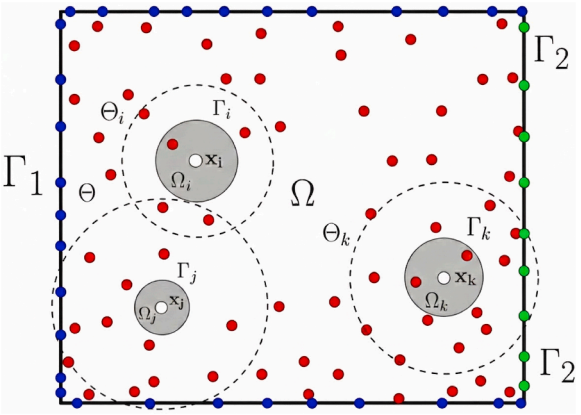

Fig. 1. Schematic representation of local stencils Θ_i, Θ_j, Θ_k, integration subregions Ω_i, Ω_j, Ω_k, and integration countours Γ_i, Γ_j, Γ_k, for SLBDIM with Dirichlet BC (blue nodes) and Neumann BC (green nodes).

then the integral representation (6) in each subregion Ω_i can be written as

$$u(\boldsymbol{\xi}) = \int_{\Gamma_i} Q(\mathbf{x}, \boldsymbol{\xi}) u(\mathbf{x})\, d\Gamma_{\mathbf{x}} + \int_{\Omega_i} G(\mathbf{x}, \boldsymbol{\xi}) f(\mathbf{x})\, d\Omega_{\mathbf{x}} - \int_{\Omega_i} \lambda G(\mathbf{x}, \boldsymbol{\xi}) u(\mathbf{x})\, d\Omega_{\mathbf{x}}, \tag{8}$$

where the source f is data.

The collocation technique is performed only on interior points of the domain to obtain one integral depending on Γ_i and two integrals depending on Ω_i. In particular, when the source f is zero on Ω, the Eq. (8) depends on a integral domain and a line integral.

In the SLBDIM formulation, the integration contour, denoted by the symbol Γ_i, is the boundary of the local integration subregion, denoted by the symbol Ω_i. Each subregion is embedded within an interpolation stencil, named Θ_i, which exhibits significant mutual overlap. The stencil consists of a fixed number of nodes closest to the center nodes. Fig. 1 schematically illustrates the distribution of the stencils Θ_i, Θ_j, Θ_k (i.e. the stencil for each subregion) and the integration subregions Ω_i, Ω_j, Ω_k, respectively. The overlap between the two stencils Θ_i and Θ_j, which share two red nodes, is clearly visible in this figure. It is worth noting that this work does not employ ghost nodes or extrapolation techniques outside the domain. Instead, all nodes forming a stencil are interior points strictly within the physical boundary. These are shown as blue nodes for Dirichlet BC and green nodes for Neumann BC, while red nodes are interior nodes.

In order to calculate boundary and domain integrals, the subregions of integration, Ω_i, must first be defined. In this article, we use the Green-Dirichlet function, which is defined on a circle with a radius that is inversely proportional to the number of internal points of the domain Ω, meaning that these points are completely inside Ω as shown in Fig. 1. Once the discretized SLBDIM equation has been defined, the boundary and domain integrals are calculated numerically using Gauss-Legendre quadrature.

When the interpolation stencil has nodes on the boundary of the global domain $\Gamma = \Gamma_1 \cup \Gamma_2$, we consider interpolation points on the boundary of the stencil. Different cases of stencils can happend: the stencil is totally internal to the domain; the stencils have contact with Γ with Dirichlet BC; the stencils have contact with Γ with Neumann BC, and finally the stencils that have contact with Γ with both Dirichlet and Neumann BC. In all cases, BCs imposed on the boundary value problem under study.

2.2. Local interpolations with RBFs

For local interpolations of the integral method in the Eq. (8), we consider a RBF $\varphi : \mathbb{R}^d \to \mathbb{R}$ such that

$$\varphi(\mathbf{x}) = \phi(r), \qquad r = \|\mathbf{x} - \mathbf{x}_j\|, \tag{9}$$

where $\phi : [0, \infty) \to \mathbb{R}$, $\|.\|$ is the Euclidean norm on $\mathbb{R}^d$ that depends on the distance to a center $\mathbf{x}_j \in \mathbb{R}^d$. If it also depends on the shape parameter $\varepsilon > 0$, then it is often written as $\varphi_j^{\varepsilon}(\mathbf{x}) = \phi(r, \varepsilon)$.

In the Localized Boundary-Domain Integral Method (LBDIM), the field u is locally interpolated with the RBF $\{\varphi_j\}_{j=1}^n$ with center of the stencil $\Theta_{\mathbf{x}} = \{\mathbf{x}_j\}_{j=1}^n$,

$$u(\mathbf{x}) \approx \sum_{j=1}^{n} \alpha_j \varphi_j(\mathbf{x}), \tag{10}$$

where the interpolation matrix $\mathbf{A}$ has coefficients

$$\mathbf{A}_{jk} = \varphi_k(\mathbf{x}_j) = \phi(\|\mathbf{x}_j - \mathbf{x}_k\|), \quad j, k = 1, \ldots, n. \tag{11}$$

When considering data nodes on the boundary of the global domain, the interpolant is of the Hermite type (see [8]). This is the case for the stencil Θ_j with Dirichlet BC and for the stencil Θ_k with Neumann BC from Fig. 1, where the RBF interpolant used (10) is

Hermite-based. In that case, the unknown field is approximated as

$$u(\mathbf{x}) \approx \sum_{j=1}^{n_i} \alpha_j \phi\big(\|\mathbf{x}-\mathbf{x}_j\|\big) + \sum_{j=n_i+1}^{n_i+n_b} \alpha_j \mathcal{B}_\xi \phi(\|\mathbf{x}-\xi\|)|_{\xi=\mathbf{x}_j}, \tag{12}$$

where n_i is the number of interior nodes in the stencil and n_b is the number of nodes on the boundary. The operator $\mathcal{B}_\xi$ is the boundary operator with respect to $\boldsymbol{\xi}$, which is the identity operator in Dirichlet BCs and a differential operator in Neumann BCs.

The term b of (7), which defines the Helmholtz equation, is also interpolated with the RBF $\{\chi_j\}_{j=1}^m$ with the center of the stencil $\Theta_\mathbf{y} = \{\mathbf{y}_j\}_{j=1}^m$,

$$b \approx f(\mathbf{x}) - \lambda \sum_{j=1}^{m} \beta_j \chi_j(\mathbf{x}), \tag{13}$$

where the interpolation matrix $\widetilde{\mathbf{A}}$ has coefficients

$$\widetilde{\mathbf{A}}_{jk} = \chi_k(\mathbf{y}_j) = \chi(\|\mathbf{y}_j - \mathbf{y}_k\|), j,k = 1,\dots,m. \tag{14}$$

The RBFs are ultimately of the same type and have the same centers. Taking the same RBF bases with the same centers yields the results of $\{\varphi_j\}_{j=1}^n$ and $\{\chi_j\}_{j=1}^m$ for $m = n$, although these could differ depending on the application problem or numerical experience.

The local integral formulation of Eq. (8) has the form

$$u(\boldsymbol{\xi}) \approx \sum_{j=1}^{n} \alpha_j \left\{ \int_{\Gamma_i} Q(\mathbf{x},\boldsymbol{\xi})\varphi_j(\mathbf{x})\, d\Gamma_\mathbf{x} \right\} + \int_{\Omega_i} G(\mathbf{x},\boldsymbol{\xi}) f(\mathbf{x})\, d\Omega_\mathbf{x} - \sum_{j=1}^{m} \beta_j \left\{ \int_{\Omega_i} \lambda G(\mathbf{x},\boldsymbol{\xi})\chi_j(\mathbf{x}) d\Omega_\mathbf{x} \right\}, \tag{15}$$

where α_j and β_j are coefficients to be determined.

2.3. Discretization of the SLBDIM

If $\Theta = \{\mathbf{x}_1,\dots,\mathbf{x}_N\}$ is the discretization of the domain Ω and $\boldsymbol{\xi} = \mathbf{x}_i \in \Theta$ is the collocation point, the discretized formula of the unknown field is

$$u_i = u(\mathbf{x}_i) = \sum_{j=1}^{n} \alpha_j h_{ij} - \sum_{j=1}^{m} \beta_j g_{ij} + f_i, \tag{16}$$

where α_j and β_j are the coefficients from the Eqs. (10) and (13). The coefficients h_{ij}, g_{ij} and f_i are of the form

$$h\begin{cases} {}_{ij} = \displaystyle\int_{\Gamma_i} Q(\mathbf{x},\mathbf{x}_i)\varphi_j(\mathbf{x}) d\Gamma_\mathbf{x}, & \text{(17a)} \\ g_{ij} = \displaystyle\int_{\Omega_i} \lambda G(\mathbf{x},\mathbf{x}_i)\chi_j(\mathbf{x}) d\Omega_\mathbf{x}, & \text{(17b)} \\ f_i = \displaystyle\int_{\Omega_i} G(\mathbf{x},\mathbf{x}_i) f(\mathbf{x}) d\Omega_\mathbf{x}, & \text{(17c)} \end{cases}$$

where the domain and boundary integrals are calculated numerically by Gauss-Legendre quadratures (see [6]).

Defining the column vectors $\boldsymbol{\alpha} = [\alpha_1,\dots,\alpha_n]^T$ and $\boldsymbol{\beta} = [\beta_1,\dots,\beta_m]^T$ as interpolation coefficients, the discretized form (16) of u can be expressed as

$$u_i = \mathbf{h}_i^T\boldsymbol{\alpha} + \mathbf{g}_i^T\boldsymbol{\beta} + f_i, \tag{18}$$

where the column vectors $\mathbf{h}_i = [h_{i1},\dots,h_{in}]^T$ and $\mathbf{g}_i = [g_{i1},\dots,g_{im}]^T$ are the influence coefficients and $f_i \in \mathbb{R}$ is the data.

By interpolation with the RBF basis $\{\varphi_j\}_{j=1}^n$, the vector $\boldsymbol{\alpha}$ is obtained from the local system $\mathbf{A}_i\boldsymbol{\alpha} = \mathbf{d}_i$ as

$$\boldsymbol{\alpha} = \mathbf{A}_i^{-1}\mathbf{d}_i, \tag{19}$$

where the components of the interpolation matrix are given by (11) and the vector $\mathbf{d}_i$ can be described in terms of the unknown nodal values of u and the known boundary condition values of the vector $g(u_b)$ (see Eq. (1b)).

By interpolation with the RBF basis $\{\chi_j\}_{j=1}^m$, the vector $\boldsymbol{\beta}$ is obtained from the local system $\widetilde{\mathbf{A}}_i\boldsymbol{\beta} = \mathbf{b}_i$ as

$$\boldsymbol{\beta} = \widetilde{\mathbf{A}}_i^{-1}\mathbf{b}_i, \tag{20}$$

where the components of matrix $(\widetilde{\mathbf{A}}_i)_{jk} = \chi_k(\mathbf{y}_j)$ for $j,k = 1,\dots,m$ and the vector $\mathbf{b}_i$ can be described as vector $\mathbf{d}_i$ in terms of the unknown nodal values of u and $g(u_b)$ evaluated at $\mathbf{y}_j$.

We define the matrix $\mathbf{A}_{\mathbf{b}_i} \in \mathbb{R}^{m\times n}$ as the computation matrix of the vector $\mathbf{b}_i$ with known coefficients,

$$(\mathbf{A}_{\mathbf{b}_i})_{jk} = b\big(\varphi_k(\mathbf{y}_j)\big), \quad j = 1,\dots,m, \quad k = 1,\dots,n, \tag{21}$$

where b is the function from Eq. (7). Using the Eqs. (21) and (19) in (20) we get

$$\boldsymbol{\beta} = \widetilde{\mathbf{A}}_i^{-1}\mathbf{b}_i = \widetilde{\mathbf{A}}_i^{-1}\Big(\mathbf{A}_{\mathbf{b}_i}\boldsymbol{\alpha}\Big) = (\widetilde{\mathbf{A}}_i^{-1}\mathbf{A}_{\mathbf{b}_i})\boldsymbol{\alpha} = \widetilde{\mathbf{A}}_i^{-1}\mathbf{A}_{\mathbf{b}_i}\big(\mathbf{A}_i^{-1}\mathbf{d}_i\big), \tag{22}$$

which depends on the known interpolation matrices and the boundary conditions of the problem are incorporated in vector $\mathbf{d}_i$.

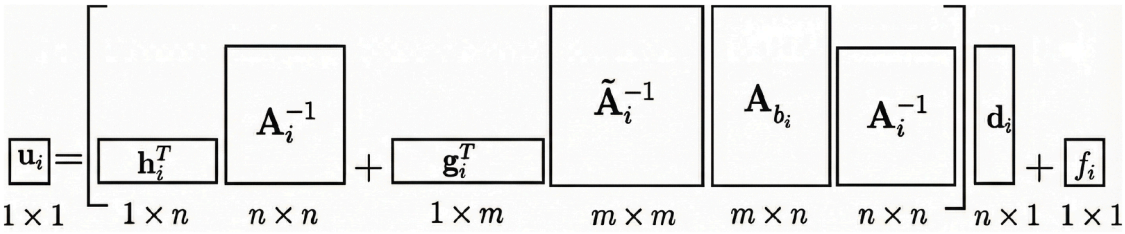

Fig. 2. Matrix diagram in the SLBDIM of the Eq. (23).

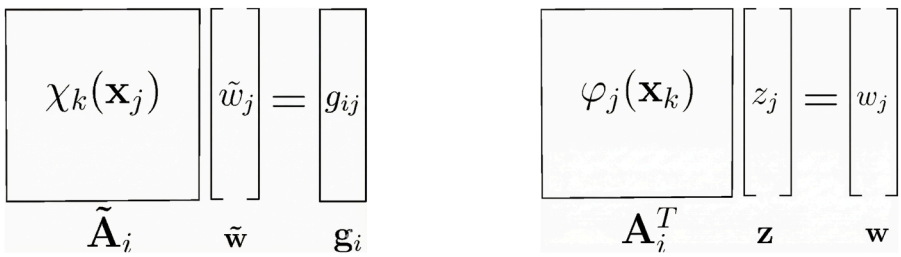

Fig. 3. Step 1 and step 3 of the algorithmic procedure in the numerical resolution of local SELs with RBF basis $\{\chi_j\}_{j=1}^m$ and $\{\phi_j\}_{j=1}^n$, respectively.

2.4. Implementation of the SLBDIM

2.4.1. Assembled matrix of the SLBDIM

Substituting (19) and (22) into the discretized form (18), we obtain the discretized matrix form for u_i in terms of $\mathbf{d}_i$

$$u_i = \left(\mathbf{h}_i^T \mathbf{A}_i^{-1} + \mathbf{g}_i^T \widetilde{\mathbf{A}}_i^{-1} \mathbf{A}_{\mathbf{b}_i} \mathbf{A}_i^{-1} \right) \mathbf{d}_i + f_i, \quad i = 1, \ldots, N, \tag{23}$$

which is calculated at each collocation node of each stencil to form the assembled matrix. A matrix diagram is shown in Fig. 2.

2.4.2. Algorithmic procedure for avoiding the calculation of the matrix inverse in SLBDIM

To maintain numerical efficiency and stability, the explicit numerical computation of the inverse matrices $\mathbf{A}_i^{-1}$ and $\widetilde{\mathbf{A}}_i^{-1}$ is avoided. Instead, the small local dense systems are solved utilizing a direct method, such as LU decomposition suitable for these systems. We rewrite expression (23) as

$$u_i = \mathbf{z}^T \mathbf{d}_i + f_i, \tag{24}$$

where

$$\mathbf{z}^T = \mathbf{h}_i^T \mathbf{A}_i^{-1} + \mathbf{g}_i^T \widetilde{\mathbf{A}}_i^{-1} \mathbf{A}_{\mathbf{b}_i} \mathbf{A}_i^{-1}. \tag{25}$$

So the algorithmic procedure for solving Eq. (23) while avoiding the computational cost of calculating the inverses is as follows:

Step 1. Solve $\widetilde{\mathbf{A}}_i \widetilde{\mathbf{w}} = \mathbf{g}_i$ (since $\widetilde{\mathbf{A}}_i$ is symmetric).
Step 2. Calculate $\mathbf{w} = \mathbf{h}_i + \mathbf{A}_{b_i}^T \widetilde{\mathbf{w}}$.
Step 3. Solve $\mathbf{A}_i^T \mathbf{z} = \mathbf{w}$.
Step 4. Develop $u_i = \mathbf{z}^T \mathbf{d}_i + f_i$.

The Fig. 3 shows a diagram of the efficient algorithmic procedure used in the numerical solution of local linear systems with Gaussian RBFs in the SLBDIM. These equations are assembled into a sparse system and solved numerically using iterative projection methods.

3. Stabilization of the SLBDIM

3.1. Stability with Gaussian RBFs

There are several types of RBFs $\phi(r)$ depending on a shape parameter $\varepsilon > 0$, as shown in Table 1. In this work, we focus on the use of strictly positive definite Gaussian RBFs that depend on ε.

Even if the invertibility of the interpolation matrix is guaranteed for GA, MQ, IMQ, IQ, stability still is a research topic when interpolating with RBF. When the shape parameter ε tends to zero, the interpolation error decreases until it becomes unstable for low values of ε. This is because the interpolation matrix (26) becomes extremely ill-conditioned, the expansion coefficients α_j become

Table 1
Typical RBFs depending on the shape parameter ε.

Smooth $\phi(r,\varepsilon)$	Name of RBF	Abbreviation
$e^{-(\varepsilon r)^2}$	Gaussian	GA
$\sqrt{1+(\varepsilon r)^2}$	Multiquadric	MQ
$\left(1+(\varepsilon r)^2\right)^{\beta}, \beta \in \mathbb{R}\backslash\mathbb{N}_0$	Generalized Multiquadric	GMQ
$1/\sqrt{1+(\varepsilon r)^2}$	Inverse Multiquadric	IMQ

large in magnitude, and numerical oscillation causes cancellation errors. The linear systems that arise from local interpolations with RBF are solved using direct methods in [21,22].

Some strategies to solve the stability problem include algorithms for finding an optimal shape parameter [23] or an algorithm for determining an optimal interval for the shape parameter [24]. These schemes involve an additional calculation to determine these optimal values for approximating a function and for approximating the solution of PDEs. In these works, the optimal value of the shape parameter is suggested by optimizing an error function or by finding a suitable interval for choosing a good shape parameter.

The RBF interpolation matrix for a given RBF ϕ is given by

$$\mathbf{A}(\varepsilon) = \begin{bmatrix} \phi(\|\mathbf{x}_1-\mathbf{x}_1\|,\varepsilon) & \phi(\|\mathbf{x}_1-\mathbf{x}_2\|,\varepsilon) & \dots & \phi(\|\mathbf{x}_1-\mathbf{x}_n\|,\varepsilon) \\ \phi(\|\mathbf{x}_2-\mathbf{x}_1\|,\varepsilon) & \phi(\|\mathbf{x}_2-\mathbf{x}_2\|,\varepsilon) & \dots & \phi(\|\mathbf{x}_2-\mathbf{x}_n\|,\varepsilon) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|\mathbf{x}_n-\mathbf{x}_1\|,\varepsilon) & \phi(\|\mathbf{x}_n-\mathbf{x}_2\|,\varepsilon) & \dots & \phi(\|\mathbf{x}_n-\mathbf{x}_n\|,\varepsilon) \end{bmatrix}. \tag{26}$$

When ε is small, the RBFs become almost linearly dependent ('flat'), forming an inadequate function basis and generating ill-conditioned interpolation matrices $\mathbf{A}(\varepsilon)$ in an appropriate interpolation space. To avoid this issue, numerical techniques have been developed in [25,26] that stabilize the solutions of linear systems where the RBFs forming the matrix of the system take arbitrarily small shape parameters. More recent methods are summarized in [27]. The RBF-QR method, which was developed for global interpolation of scattered nodes using Gaussian RBFs, is numerically stable for near-zero parameters, including the $\varepsilon \to 0$ limit. The RBF-QR algorithm transforms the basis of functions, denoted by $\{\phi_j\}$, into a new basis, denoted by $\{\psi_j\}$, using combinations of polynomial powers, Chebyshev polynomials and trigonometric functions. This novel approach was adopted by Ponzellini Marinelli, Caruso and Portapila in [19] for local integral methods for general elliptic PDEs.

3.2. Stability of the local integral method

As mentioned in section before, with the new basis $\{\psi_j\}$, the discretized equation of this new formulation now takes the form

$$u_i = \sum_{k=1}^{n} \gamma_k \, l_{ik} + \sum_{k=1}^{n} \lambda_k \, \widetilde{l}_{ik} + \tilde{f}_i, \tag{27}$$

where γ_k and λ_k are the coefficients from the Eqs. (10) and (13) in the new basis. The coefficients l_{ik}, $\widetilde{l}_{ik}$ and $\tilde{f}_i$ are

$$l \begin{cases} {}_{ij} = \displaystyle\int_{\Gamma_i} Q(\mathbf{x},\mathbf{x}_i)\psi_j(\mathbf{x})d\Gamma_{\mathbf{x}}, & \text{(28a)} \\ \widetilde{l}_{ij} = \displaystyle\int_{\Omega_i} \lambda G(\mathbf{x},\mathbf{x}_i)\psi_j(\mathbf{x})d\Omega_{\mathbf{x}}, & \text{(28b)} \\ \widetilde{f}_i = \displaystyle\int_{\Omega_i} G(\mathbf{x},\mathbf{x}_i) f(\mathbf{x})d\Omega_{\mathbf{x}}. & \text{(28c)} \end{cases}$$

The new interpolation matrix form for u of (23) at each node is given by

$$u_i = \left(\boldsymbol{l}_i^T \boldsymbol{B}_i{}^{-1} + \widetilde{\boldsymbol{l}}_i^T \widetilde{\boldsymbol{B}}_i{}^{-1} \boldsymbol{B}_{\mathbb{b}_i} \boldsymbol{B}_i{}^{-1}\right)\boldsymbol{d}_i + \tilde{f}_i, \tag{29}$$

where $\boldsymbol{l}_i = [\dots, l_{ik}, \dots]^T$ and $\widetilde{\boldsymbol{l}}_i = [\dots, \widetilde{l}_{ik}, \dots]^T$ are the column vectors. The new matrix diagram in the SLBDIM of the equation is similar to Fig. 2.

Here, the column vectors $\boldsymbol{l}_i$ and $\widetilde{\boldsymbol{l}}_i$ represent the discrete contributions to the system matrices. Specifically, $\boldsymbol{l}_i$ corresponds to the column vector obtained by integrating the fundamental solution with the new basis functions ψ_j , while $\widetilde{\boldsymbol{l}}_i$ corresponds to the intregral with the new basis and the coefficient λ from Helmholtz Eq. (1b). The distinct notation is used here to emphasize that these vectors are computed using the specific new local basis functions defined in the previous section.

The local interpolation matrix used for internal stencils is,

$$\boldsymbol{B}_{\psi}^{i} = \boldsymbol{V}\begin{bmatrix} \boldsymbol{I}_n \\ \widetilde{\boldsymbol{R}}^T \end{bmatrix}, \tag{30}$$

where the components of the matrix $\boldsymbol{V}$ are $V_{jk} = V_k(\mathbf{x}_j)$ for $j,k = 1,\dots,n$ and $\widetilde{\boldsymbol{R}}$ is the correction matrix in the RBF-QR method ([25] for details).

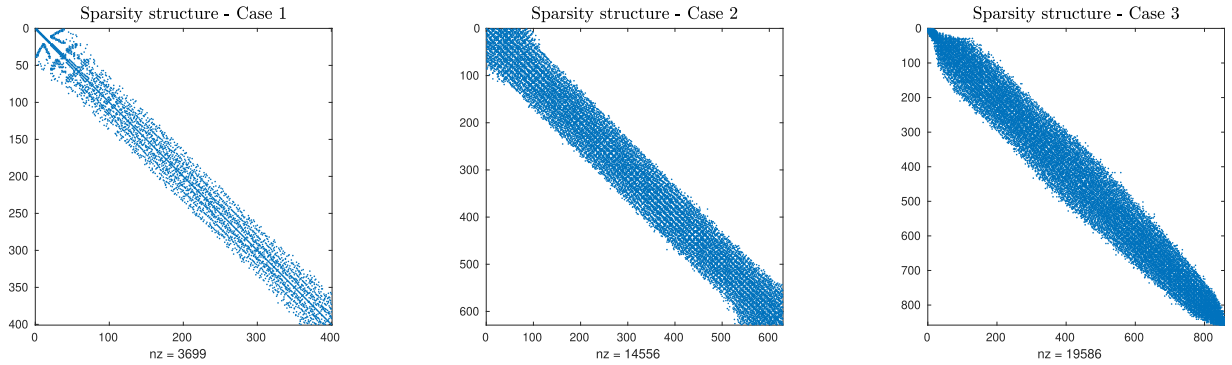

Fig. 4. Sparsity structure for the SLBDIM with interior nodes of $N_{int} = 400$ (left), 628 (center) and 857 (right), and a stencil size of $n = 25$.

For boundary stencils, the local interpolation matrix is $\boldsymbol{B}_i$ and consists of two matrix blocks,

$$\boldsymbol{B}_i = \begin{bmatrix} \boldsymbol{B}^i_{\psi} \\ \boldsymbol{B}^i_{\mathcal{B}\psi} \end{bmatrix}, \tag{31}$$

where the components of the first matrix block is $(\boldsymbol{B}^i_{\psi})_{jk} = \psi_k(\mathbf{x}_j)$, for $j = 1, \dots, n_{int}$ (interior nodes) and $k = 1, \dots, n$ (boundary nodes), and the components of the second matrix block is $(\boldsymbol{B}^i_{\mathcal{B}\psi})_{jk} = \mathcal{B}\left[\psi_k(\mathbf{x}_j)\right]$ for $j = n_{int} + 1, \dots, n$ and $k = 1, \dots, n$, where $\mathcal{B}[.]$ is the boundary operator applied to the new basis function ψ_k, which is the identify operator for Dirichlet BCs and a differential operator for Neumann BCs.

As the shape parameter ε in the Gaussian RBF $\phi(r, \varepsilon) = e^{-(\varepsilon r)^2}$ approaches zero (flat basis), the precision of the interpolation method increases but the local interpolation matrix becomes ill-conditioned. The new basis ψ_k that expands the same space of the Gaussian RBFs produces a better local interpolation basis for numerical solutions. The RBF-QR method is then applied to stabilise the direct algorithm.

Also, to avoid calculating the inverse matrices ${\boldsymbol{B}_i}^{-1}$ and ${\widetilde{\boldsymbol{B}}_i}^{-1}$ of Eq. (29) when $\varepsilon \to 0$ we follow an algorithmic procedure presented in Section 2.4.2. Specifically, Steps 1 and 3 in Fig. 3 are used to solve the system without forming the explicit inverse of these interpolation matrices in the new basis.

Incorporating these techniques into the local integral method stabilizes the numerical error of the approximation of the Helmholtz-type equations. The SLBDIM has been presented and tested in [19] for Poisson problems, convection-diffusion equations and general elliptic PDEs. Another strategy of stability technique for local integral methods using RBF interpolation functions was presented in Ponzellini Marinelli [28].

3.3. Numerical resolution of the assembled linear system

Rearranging Eq. (29) for each interior node yields an algebraic equation belonging to a system known as the assembled linear system:

$$\mathbf{M}\mathbf{u} = \mathbf{c}, \tag{32}$$

where $\mathbf{M} \in \mathbb{R}^{N \times N}$ is the coefficient matrix, $\mathbf{c} \in \mathbb{R}^N$ is the vector on the right-hand side; and $\mathbf{u} \in \mathbb{R}^N$ is the vector of unknowns for reconstructing the approximation of the solution $u(x)$ over the entire domain Ω. Since the stencils are strongly overlapping and their size is much smaller than that of the matrix $n << N$, each equation has a small number of unknowns. Consequently, the matrix $\mathbf{M}$ is sparse but the vector $\mathbf{c}$ is not.

It is relevant to note that we use both local and global strategies for solving our linear systems. First, for each stencil, we solve two small, dense local systems directly to obtain the weights. These weights are then assembled into a global sparse matrix. Finally, we obtain the assembled linear system and solve it iteratively. The numerical solution to this system is performed using Krylov-type iterative methods to reduce the computational cost. In particular, the generalized minimal residual method (GMRES), see [29], is a standard variant of the Krylov subspace methods that is stable and low cost for large unsymmetric linear systems generated by the SLBDIM. One advantage of GMRES is that it computes an approximate solution with a minimum residual norm. In [30], a preconditioned iterative version of GMRES method for the two-dimensional Helmholtz equation was shown to significantly reduce memory requirements and arithmetic complexity. Furthermore, this relative improvement increases as the size of the problem grows.

Fig. 4 shows the typical sparsity structures of the assembled linear algebraic systems generated by the SLBDIM. The Dirichlet or Neumann BCs are incorporated at the right-hand side of the system. For the experimental cases in this paper, the matrices have 2.3%, 3.7% and 2.7% nonzeros at present, but as N increases, the achieved sparsity pattern is 0.93% for $N_{int} = 2560$. A preconditioner is not applied in these cases since the problem is well conditioned. In fact, the condition number of the corresponding assembled matrices in the figure are 2.07×10^1, 1.55×10^4 and 2.44×10^4, respectively.

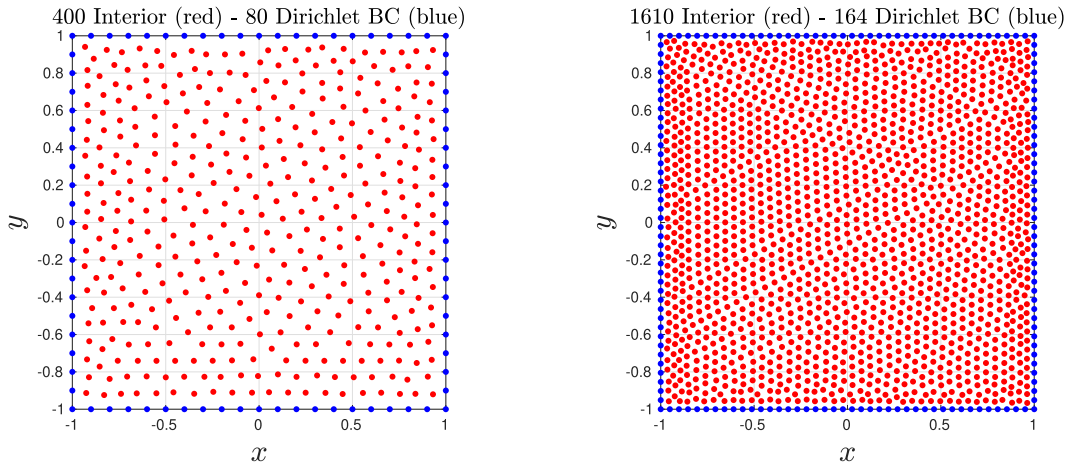

Fig. 5. Quasi-uniform 2D node distribution for $N_{int} = 400, 1610$ interior points and $N_{bou} = 80, 164$ boundary points with Dirichlet BC, respectively.

4. Numerical examples on several domains

In this section, we report four numerical experiments to demonstrate the accuracy and efficiency of the proposed numerical scheme for solving two-dimensional Helmholtz-type equations. The algorithm implementations and the numerical experiments were performed using MATLAB version R2017a software on a PC with 7.5 GB of RAM and a 2.70 GHz 7th generation Intel Core i7-7500U processor.

The reported errors are the standard error L_2

$$L_2\text{-Error} \quad = \quad \sqrt{\frac{\sum_{i=1}^{N} \left(u_i^{exac} - u_i^{approx}\right)^2}{\sum_{i=1}^{N} \left(u_i^{exac}\right)^2}}, \tag{33}$$

the standard error L_∞

$$L_\infty\text{-Error} \quad = \quad \max_{i=1,\ldots,N} \{|u_i^{exac} - u_i^{approx}|\}, \tag{34}$$

and the root mean square error

$$\text{RMSE} \quad = \quad \sqrt{\frac{\sum_{i=1}^{N} \left(u_i^{exac} - u_i^{approx}\right)^2}{N}}. \tag{35}$$

The condition number (CN) of a square matrix $\mathbf{A}(\varepsilon)$ is defined as $\kappa(\mathbf{A}(\varepsilon)) = \|\mathbf{A}(\varepsilon)\| \|\mathbf{A}^{-1}(\varepsilon)\|$. If $\kappa(\mathbf{A}(\varepsilon)) \approx 1$, then the matrix $\mathbf{A}$ is well-conditioned. As $\kappa(\mathbf{A}(\varepsilon))$ increases, the matrix becomes ill-conditioned. To analyze the stability of the SLBDIM, we evaluate the CN of the local collocation matrix (30) by using the Matlab command `cond`.

4.1. Case 1

This Helmholtz-type PDE is given over the rectangular domain $\Omega = [-1, 1]^2$ and the boundary domain $\Gamma = \partial\Omega$

$$\begin{cases} \Delta u - k^2 u = f, & (x, y) \in \Omega, \\ u = g, & (x, y) \in \Gamma, \end{cases} \tag{36}$$

where the oscillating source function is

$$f(x, y) = 2\cos(x^2 + y) - (4x^2 + 1 + k^2)\sin(x^2 + y), \tag{37}$$

and the parameter $k = 9$. The BCs of this BVP are of Dirichlet type and the analytical solution is $u(x, y) = \sin(x^2 + y)$. In this case we will use the local integral method presented in its original form with Gaussian RBF kernels $\phi(r) = e^{-(\varepsilon r)^2}$, called LBDIM, and in its stabilized form, called SLBDIM.

There are several ways to discretize the Ω domain with distributions of nodes. In our case, we use the quasi-uniform distribution generation algorithm developed in [11] for 2D. These distributions were generated using a fast-forward method, which creates a set of nodes from a density function starting at the boundary of the domain and moving towards its interior. In Fig. 5, we show two discretization points for problem (36) using Gaussian RBFs with $N_{int} = 400$ internal nodes and $N_{bou} = 80$ boundary points, and $N_{int} = 1610$ internal nodes and $N_{bou} = 164$, respectively. Note that the generation algorithm can construct points uniformly whitin the interior domain Ω and on the boundary Γ.

We compare the L_2-Error of the LBDIM and the SLBDIM using the Gaussian RBFs for the local interpolations with $\varepsilon \in [1, 10]$. Fig. 6 shows that, as ε decreases, the accuracy increases; however, the LBDIM becomes unstable and the convergence is interrupted

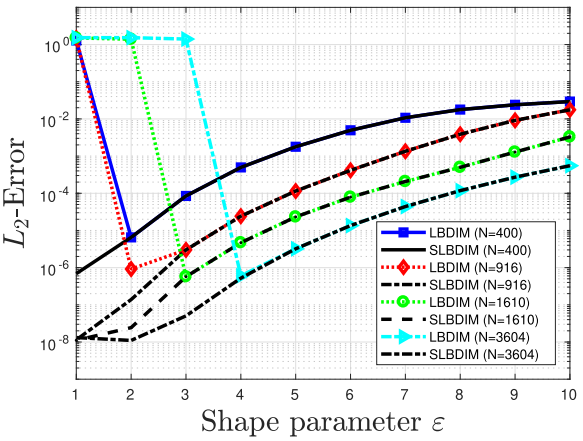

Fig. 6. Comparison of the L_2-Error between LBDIM and SLBDIM versus the shape parameter ε.

for all cases with quasi-uniform nodes with $N = 400, 916, 1610, 3604$. Once again, we observe that the errors decrease as the number of nodes on the domain and boundary increases. This plot shows that local interpolation leads to a loss of accuracy for small shape parameters. However, the best performance is achieved by using the SLBDIM to solve this Helmholtz-type equation with known analytical solutions. The error for $N = 916, 1610, 3604$ is of order 10^{-8}. In this numerical experiment, the stencil size is fixed at $n = 50$.

In Fig. 7 we show the isolines of the $\log_{10}(L_2\text{-Error})$ for $\varepsilon \in [1, 10]$ and stencil sizes $n = 10{:}10{:}100$. As n increases, the size of the linear systems increase and the conditioning of the interpolation matrices deteriorates. To understand the importance of the local stability technique, both plots in this figure must be examined simultaneously. The yellow region in the top left shows the region of numerical instability due to poor conditioning, while the dark blue region on the right shows how order of 10^{-8} is stably achieved. As N increases from 916 to 3604, this numerical behaviour remains similar when reading the figure row-wise.

In [17], the same Helmholtz-type PDE is treated with mixed type BC. This paper shows that, for $N = 900$ nodes, the L_2-Error is of order 10^{-5} using the RBF–Finite Difference (RBF–FD) technique with a Gaussian hybrid kernel of type $\phi(r) = \alpha e^{-(\varepsilon r)^2} + \beta r^3$, where α and β are the weights controlling the contribution of the Gaussian and cubic kernels, respectively.

In Fig. 8, the CN isolines $\log_{10}(\kappa(\boldsymbol{A}_i))$ are shown. The range of the shape parameter is $[1, 10]$ and those for different stencil sizes are $n = 10{:}10{:}100$. As n increases, the conditioning of the local interpolation matrices also increases. The yellow area in the top left shows the region where the CN is over 10^{20}. In the isolines of the plots in the right-hand column, the dark blue region shows better conditioning up to 10^{10}. These ten orders of magnitude are significant when using linear solvers numerically. As N increases from 916 to 3604, we can observe that the conditioning behaviour remains similar when reading the figures row-wise.

Fig. 7 shows that the error plots suggest smaller values of ε for better accuracy, whereas the condition isolines plots in this figure suggest larger values of ε for better stability.

4.2. Case 2

Consider the following two-dimensional Helmholtz equation

$$\begin{cases} \Delta u + k^2 u = f, & \Omega = [0,1] \times [0,1], \\ u = g, & \Gamma = \partial\Omega, \end{cases} \tag{38}$$

where $k^2 = 2$ and the source is $f(x, y) = 2x - 4y$. The analytical solution is

$$u(x, y) = \sin(\sqrt{3}x)\sinh(y) + \cos(\sqrt{2}y) + x - 2y, \tag{39}$$

and the function g, which is defined over the boundary Γ, is chosen to be the same, resulting in a Dirichlet boundary condition.

We use quasi-uniform nodes within the domain and stencils of size $n = 25$, counting the collocation center, as shown in Fig. 9. We also use the Matlab built-in function `knnsearch` to find the nearest neighbours when calculating the stencils which is a k-dimensional tree algorithm. This reduces the bandwidth of the sparse assembled matrix, as shown in Fig. 4.

In Table 2 we show the accuracy of the SLBDIM for the fixed shape parameter $\varepsilon = 1$. The number of quasi-uniform interior points of the domain, N, varies from 121 to 628. It can be seen that the orders of magnitude decrease from 10^{-6} to 10^{-8} starting from 441 nodes. Our study also points out that CNs for local collocation matrices (30) generated by SLBDIM are lower than the ones obtained by direct methods for low values of ε. The CNs are of order 10^4 for the stencil size $n = 25$.

In Table 3 we show the accuracy of the SLBDIM for a range of low values, $\varepsilon \in \{10^0, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. Convergence of the method is observed for low values of the shape parameter, achieving an RMSE of order 10^{-8} at 225 nodes. The ε shown corresponds to the lowest error achieved in this range. In this case, the CN varies from 10^4 to 10^5 for the stencil size $n = 25$.

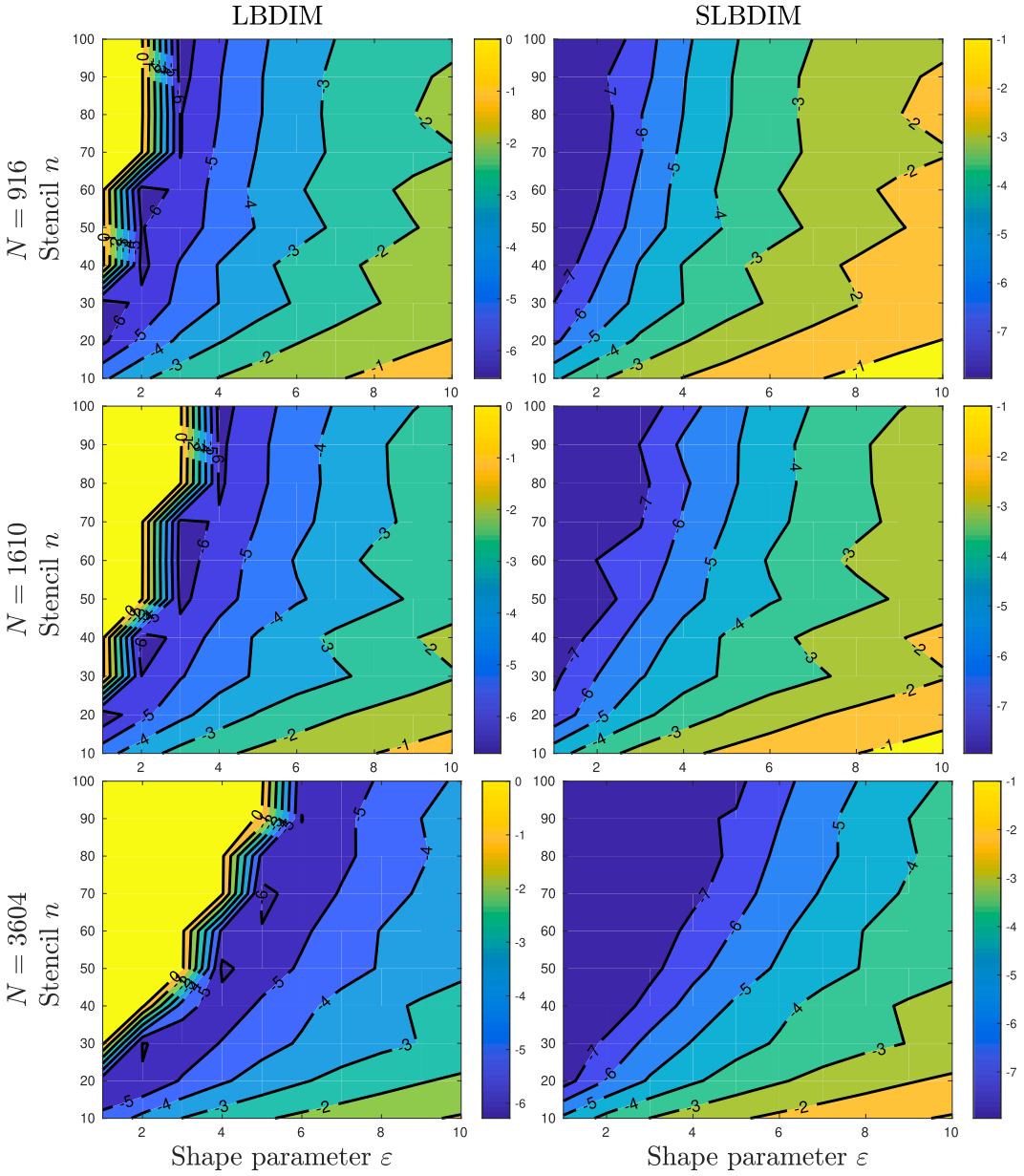

Fig. 7. Accuracy isolines ($\log_{10}(L_2$-Error)) with $N_{int} = 916, 1610, 3604$ interior points varying the shape parameter ϵ and the stencil size n.

Table 2
L_2-Error and $RMSE$ of the SLBDIM for shape parameter $\epsilon = 1$.

N		SLBDIM			
N_{int}	N_{bou}	ϵ	L_2-Error	$RMSE$	CN
121	44	1.0	1.30e-06	1.20e-06	1.87e+04
225	60	1.0	6.25e-07	5.86e-07	2.15e+04
361	76	1.0	4.16e-07	3.93e-07	2.35e+04
441	88	1.0	8.29e-08	7.86e-08	4.48e+04
530	92	1.0	5.59e-08	5.29e-08	7.57e+04
628	100	1.0	4.58e-08	4.38e-08	3.00e+04

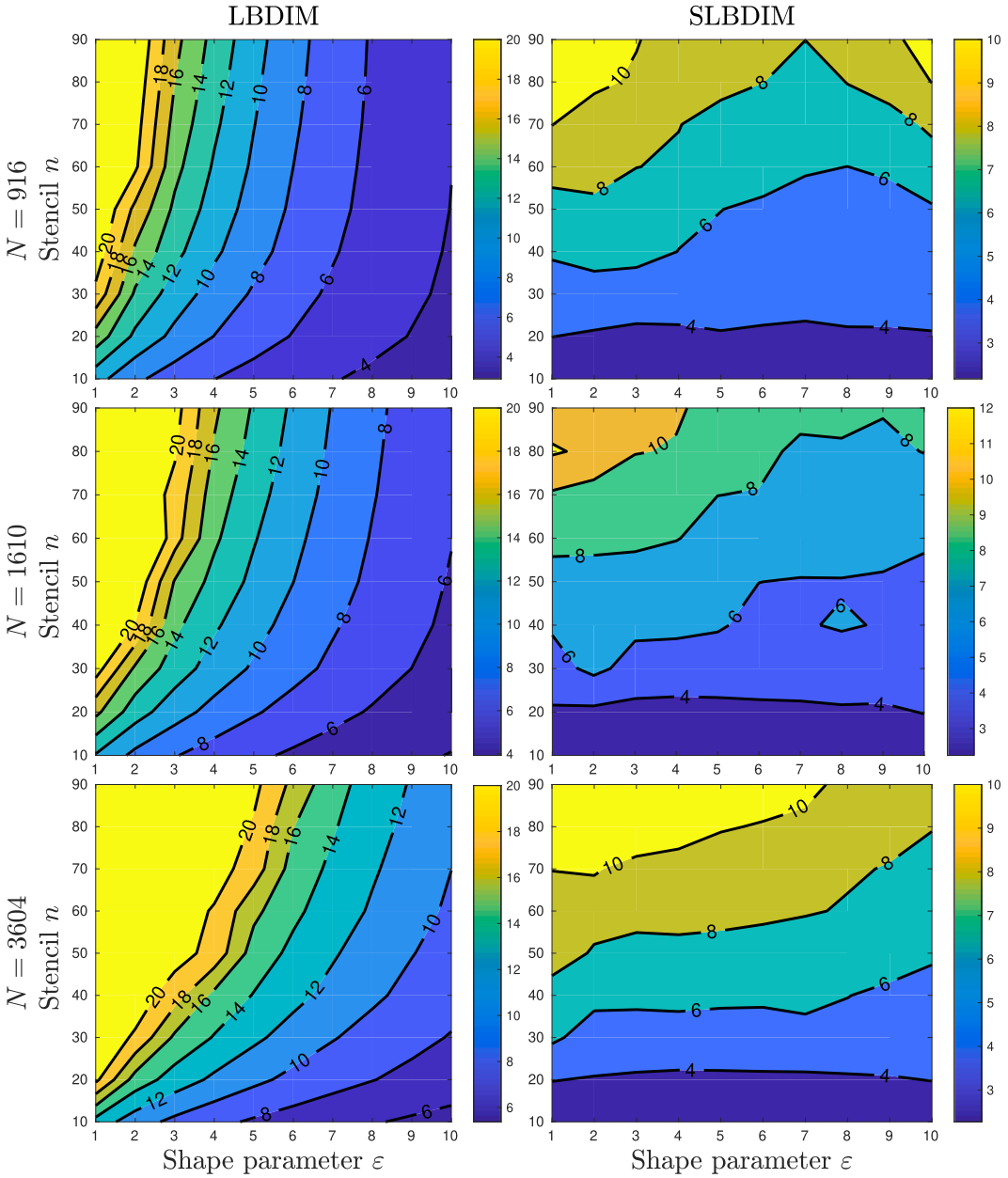

Fig. 8. Condition number isolines ($\log_{10}(\kappa(\boldsymbol{A}_i))$ with $N_{int}=916,1610,3604$ interior points varying the shape parameter ϵ and the stencil size n.

Table 3
L_2-Error and $RMSE$ for $\epsilon \in \{10^{-1}, \dots, 10^{-5}\}$.

N		SLBDIM			
N_{int}	N_{bou}	low ϵ	L_2-Error	$RMSE$	CN
121	44	0.1	2.31e-07	2.14e-07	4.60e+04
225	60	0.1	5.43e-08	5.08e-08	1.53e+05
361	76	0.01	3.58e-08	3.38e-08	2.89e+04
441	88	0.1	3.57e-08	3.39e-08	6.46e+04
530	92	0.00001	3.80e-08	3.60e-08	9.57e+04
628	100	0.00001	3.85e-08	3.69e-08	5.34e+04

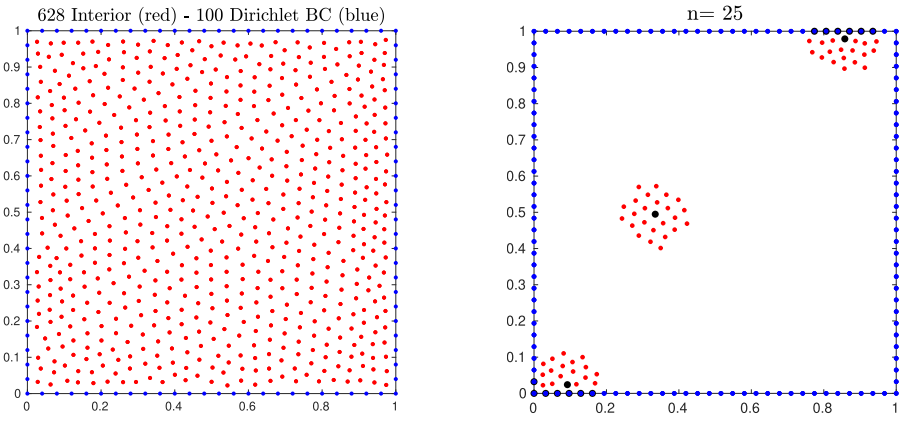

Fig. 9. Quasi-uniform node distribution with $N = 628$ interior nodes (left). Stencil node sets with $n = 25$ (right).

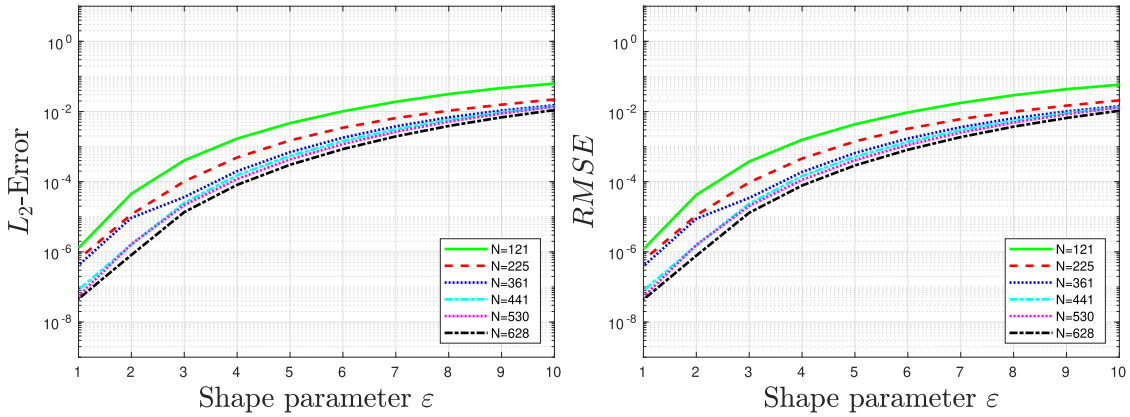

Fig. 10. Comparison of the L_2-Error and $RMSE$ for the SLBDIM versus the shape parameter ε.

In [15], this differential problem with mixed BC over the same domain is investigated. The authors use Multiquadric RBF kernels $\varphi(r,\varepsilon) = \sqrt{1+(\varepsilon r)^2}$ and a novel RBF with placement points in the range $N \in [50, 350]$. The results obtained in this reference indicate errors of order 10^{-5} for the shape parameter $\varepsilon \in [0, 4]$.

Fig. 10 shows that, as ε decreases for the local interpolation matrices, the accuracy increases for all cases with quasi-uniform nodes. Here, we consider $N_{int} = 121, 225, 361, 441, 530, 628$ internal nodes and $N_{bou} = 44, 60, 76, 88, 92, 100$ boundary nodes, respectively. The L_2-Error and $RMSE$ show the same behavior. For the values $N_{int} = 441, 530, 628$, the errors are of order 10^{-8} when using a stencil size of $n = 25$.

As mentioned in Section 3.3, the numerical solution to the assembled system is found using the GMRES iterative method. The sparsity structure shown in Fig. 4 and the CN of the assembled matrix for SLBDIM result in a low numerical cost. In this case, no preconditioner is applied since the CN is 1.55×10^4 for the largest case $N_{int} = 628$.

Increasing the wavenumber

The Helmholtz problems presented in (38) are challenging because the matrices become indefinite when the sign is positive before k^2, and the solutions can be of a high frequency, making them hard to resolve and difficult to determine the correct phase. Various types of boundary condition are used to handle transmitted and reflected waves correctly. We include some higher-frequency tests for $k^2 \in [5, 50]$, as shown in Fig. 11. We modified the source shown in Eq. (39) due to the analytical solution given for $k^2 = 2$ in that case. The new source is given by

$$f_{new}(x, y) = (k^2 - 2)\sin(\sqrt{3}x)\sinh(y) + (k^2 - 2)\cos(\sqrt{2}y) + k^2(x - 2y). \tag{40}$$

The numerical results demonstrate the robustness of the SLBDIM approach. As can be seen in Fig. 11, the L_2-Error and the $RMSE$ remain stable at approximately 10^{-8} for a wide range of frequencies, from $k^2 = 5$ to $k^2 = 35$. The L_∞-Error, while roughly an order of magnitude higher, displays the same stability across this frequency range, validating the method's local accuracy. At higher wavenumbers, from $k^2 = 40$ to $k^2 = 50$, we observe the expected increase in error due to the pollution effect inherent in numerical schemes for the Helmholtz equation. Nevertheless, the method remains reasonably accurate and does not become unstable.

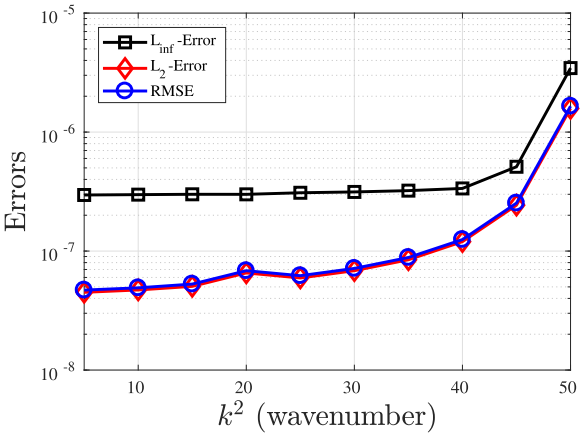

Fig. 11. Comparison of L_∞-Error, L_2-Error and $RMSE$ for the SLBDIM versus the wavenumber k^2.

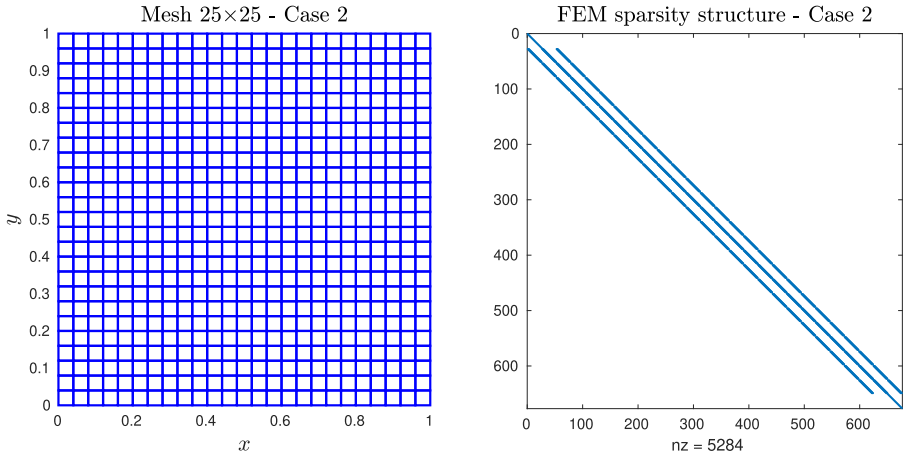

Fig. 12. FEM mesh uniform node distribution $N = 625$ interior nodes (left). Sparsity structure FEM matrix (right).

Sparsity and convergence rates comparison with FEM

While the stabilization of the SLBDIM demonstrates robustness, a comparative performance analysis is included. Specifically, we benchmark our meshless method against the standard finite element method (FEM) with linear basis functions. To assess the efficiency of the SLBDIM relative to the traditional FEM, we examine the trade-off between the sparsity of the global matrix and the error convergence rates.

Fig. 12 shows a FEM grid for $N_{elem} = 26 \times 26 = 676$ and the FEM sparsity structure of the global matrix for problem (38). In this experimental case, the matrix has 5284 non-zero entries and a density of 1.35%, while the assembled SLBDIM matrix in Fig. 4 has 3.7% non-zero entries for $N_{int} = 628$ quasiuniform interior nodes. Both matrices are poorly conditioned: the CNs of the FEM global matrix is $\kappa_{FEM} = 2.14 \times 10^3$ and $\kappa_{SLBDIM} = 1.55 \times 10^4$ for the SLBDIM. The GMRES iterative method was used for both schemes with $\texttt{tol} = 1 \times 10^{-12}$.

In Table 4 we present the convergence study for the numerical of the Helmholtz problem (38) using FEM with $N_{elem} = n \times n$ square elements and N_{int} interior nodes. As the mesh refinement increases from 11×11 to 25×25 nodes, a monotonic decrease is observed in the $RMSE$, L_2-Error and L_∞-Error. To quantify the convergence rate, we analyze the reduction of the error relative to the mesh size h. Comparing the coarsest mesh with the finest mesh in the table, the experimental order of convergence is estimated to be approximately 1.8 for the L_2-norm. This behavior approaches the theoretical quadratic convergence rate ($O(h^2)$) expected for linear basis functions in the L_2 norm.

A direct comparison between Table 2 (SLBDIM) and Table 4 (FEM) reveals a difference in accuracy between the two methods. For the FEM discretization $N_{int} = 121$, the standard FEM yields an L_2-error of 1.66×10^{-4} while the SLBDIM achieves an error of 1.30×10^{-6} for the same number of interior nodes. This represents an improvement of two orders of magnitude in favor of the stabilized meshless method. This behaviour becomes more pronounced as the node density increases. At the finest resolution $N_{int} = 625$, the FEM error reduces to 3.77×10^{-5}, whereas the SLBDIM reaches 4.58×10^{-8}. While the FEM exhibits the expected algebraic convergence rate ($O(h^2)$), the SLBDIM demonstrates a much faster convergence, outperforming the FEM by nearly three orders of magnitude.

Table 4
Errors for different meshes with numbers of interior points (N_{int}).

Mesh		FEM		
N_{elem}	N_{int}	$RMSE$	L_2-Error	L_∞-Error
10×10	121	1.65e-04	1.66e-04	3.28e-04
16×16	225	9.46e-05	9.63e-05	1.85e-04
20×19	361	6.12e-05	6.27e-05	1.19e-04
22×22	441	5.08e-05	5.22e-05	9.80e-05
24×24	529	4.28e-05	4.41e-05	8.23e-05
26×26	625	3.66e-05	3.77e-05	7.01e-05

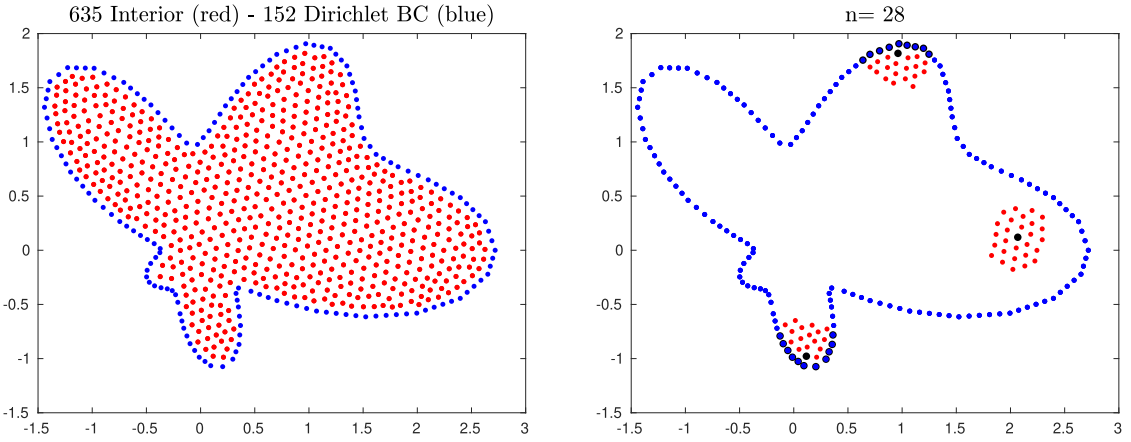

Fig. 13. Quasi-uniform node distribution in the amoeba domain with $N = 635$ interior nodes (left). Stencil node sets with $n = 28$ (right).

4.3. Case 3

In the third case, we will consider the following Helmholtz problem:

$$\begin{cases} \Delta u - k^2 u = f, & (x,y) \in \Omega, \\ u = g, & (x,y) \in \Gamma = \partial\Omega, \end{cases} \tag{41}$$

where the domain is defined by $\Omega = \{(r,\theta) : r \le \rho(\theta), 0 \le \theta \le 2\pi\}$, being $\rho(\theta) = e^{\sin(\theta)}\sin^2(2\theta) + e^{\cos(\theta)}\cos^2(2\theta)$. This example was considered in [15,16,31] for the same amoeba-like domain. The analytical solution for this case is given by $u(x,y) = xe^y - ye^{-x}$ as in [16]. Fig. 13 shows the quasi-uniform node distribution with $N_{int} = 635$ internal nodes and stencils of size $n = 28$ formed with interior or boundary nodes.

Although polyharmonic splines (PHS) with multivariate polynomials offer algebraic rates without the need to tune the shape parameter [32,33], in this work we focus specifically on achieving stable spectral convergence for Gaussian kernels using the RBF-QR stabilization technique as a solution for applications. A comparison between PHS + poly and SLBDIM was presented in [19], where the SLBDIM showed better performance for Poisson problems.

In Table 5 we show the L_∞-Error and the RMSE of the approximation u using the SLBDIM with stencils of size $n = 28$ for different numbers of boundary nodes N_{bou} and interior nodes N_{int}. As expected, the accuracy increases as the shape parameter ε of the Gaussian RBF diminishes from 1.0 to 0.1. For $\varepsilon = 1$ we obtained L_∞-Errors of order 10^{-3} and $RMSE$ of order 10^{-4} for the smallest node set, with $N_{int} = 179$ and $N_{bou} = 95$. As we increase the total number of nodes in the domain and on the boundary up to $N_{int} = 1133$ and $N_{bou} = 190$, respectively, the L_∞-Error and the $RMSE$ decrease to an order of 10^{-6}. For $\varepsilon = 0.1$, we achieve L_∞-Error and $RMSE$ of orders 10^{-5} and 10^{-6}, respectively, for the coarser discretization, and orders 10^{-7} and 10^{-8} for the finer one.

The L_∞-Error and the RMSE versus the number of interior nodes for stencil sizes $n = 21, 28, 36, 78, 153$ and $\varepsilon = 1.3$, are plotted in Fig. 14. In this figure it is observed that the L_∞-Error and the RMSE decrease in a similar way. The best results are obtained for stencil size $n = 153$ with orders of magnitude 10^{-7} and 10^{-8}, respectively. The behavior of convergence is similar to the results obtained for the RMSE versus the number of the center points under different number of collocation points in [16], but more stable.

In [15], a new RBF was proposed to solve a Helmholtz problem over the same amoeba-shape domain with different source. Convergence is shown for the relative average error against the number of collocation points for the best shape parameter in range $[0.1, 2.5]$. For the new RBF with $N \in [350, 400]$ collocation points the authors achieved an accuracy of order 10^{-5}. For $N_{int} = 357$, our results are shown in Table 6 showing better accuracy.

Table 5
L_∞-Error and $RMSE$ by the SLBDIM with stencil size $n = 28$ and different N_{int} and N_{bou}.

N		SLBDIM ($\varepsilon = 1$)		SLBDIM ($\varepsilon = 0.1$)	
N_{int}	N_{bou}	L_∞-Error	$RMSE$	L_∞-Error	$RMSE$
179	95	1.37e-03	2.08e-04	1.26e-05	1.58e-06
278	107	2.52e-04	9.09e-05	1.26e-06	3.57e-07
357	118	1.16e-04	2.26e-05	1.93e-06	2.03e-07
452	128	8.14e-05	1.96e-05	6.24e-07	1.13e-07
635	152	5.13e-05	1.34e-05	8.25e-07	1.12e-07
857	170	1.65e-05	6.87e-06	8.20e-07	1.27e-07
1133	190	7.35e-06	2.88e-06	2.73e-07	8.56e-08

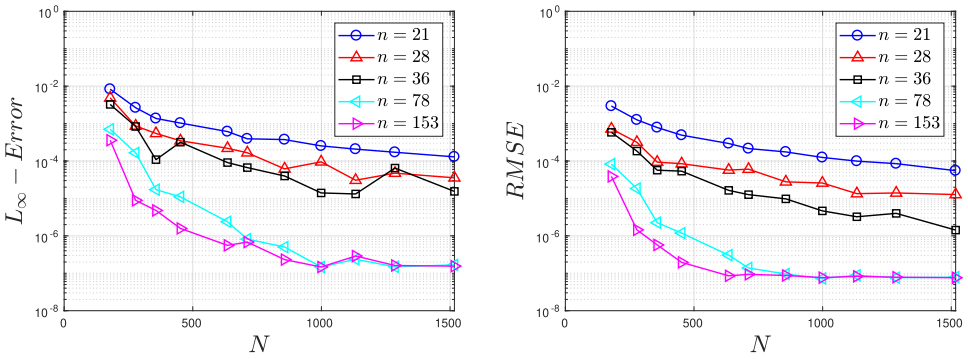

Fig. 14. The L_∞-Error (left) and the RMSE (right) versus the number of interior nodes N_{int} with different stencil sizes, $n = 21, 28, 36, 78, 153$.

Table 6
The L_∞2-Error and RMSE obtained by the SLBDIM with $n = 21, 28, 36, 78$ for optimal ε.

SLBDIM		Errors	
n	ε	L_∞-Error	$RMSE$
21	0.3	1.28e-05	8.12e-06
28	0.3	1.40e-06	3.09e-07
36	0.2	2.30e-07	7.93e-08
78	0.4	1.57e-07	7.47e-08

Convergence rate analysis

Fig. 15 displays the convergence rate for $L_\infty - Error$ and RMSE. For stencil sizes $n = 21, 28, 36$, we observe monotonic fourth-, fifth-, and sixth-order convergence, $m \approx 4, 5, 6$, respectively, across the entire refinement range. Given that in the $\varepsilon \to 0$ the Gaussian RBF approaches a polynomial, the observed rates are consistent with the theoretical convergence order. However, for finer node distributions, $h \lesssim 0.035$, the larger stencils, $n = 28, 36$, exhibit non-monotonic convergence, likely due to error saturation.

Computational cost

To quantify the computational cost, Fig. 16 shows the total CPU time and the GMRES solver times, respectively, as function of N_{int}, for stencil sizes $n \in \{21, 28, 36, 78, 153\}$. These plots reveal that for $n = 21, 28, 36$, the total computational cost remains similar. Increasing the stencil size from 21 to 36 incurs a negligible overhead in both total execution time and solver time. Choosing a larger stencil size to improve accuracy does not result in a significant performance reduction. In contrast, the largest stencil sizes $n = 78, 153$, exhibit an increase in computational cost. A comparison of the scales on the two plots shows that the GMRES time constitutes a small fraction of the total CPU time. The main computational cost of SLBDIM lies in stabilizing the solution of the linear systems for local stencils and computing weights rather than in solving the sparse linear system iteratively.

The results reported by Cavoretto and Rossi [31] using Kansa's method on amoeba-like domains show computational cost and accuracy. In that work, an adaptive refinement scheme beginning with $N = 261$ initial interior nodes and ending with $N = 1156, 1299$ achieved a RMSEs of 2.56×10^{-4} and 4.92×10^{-4}, respectively, for a similar problem. The SLBDIM can reach an RMSE of 10^{-8}, albeit at a higher computational cost, as shown in Fig. 16. However, the order 10^{-4} can be reached at a low computational cost using the direct formulation presented as the LBDIM in Case 1, as shown in Table 7. For $N = 1286$, it is observed that the RMSE is of order 10^{-4} with total CPU times between 5 and 6 s. Furthermore, with $N = 278$ interior nodes, the total time is approximately one second. This is lower than the time reported in the aforementioned paper. Additionally, the CN for the local interpolation matrices is of orders

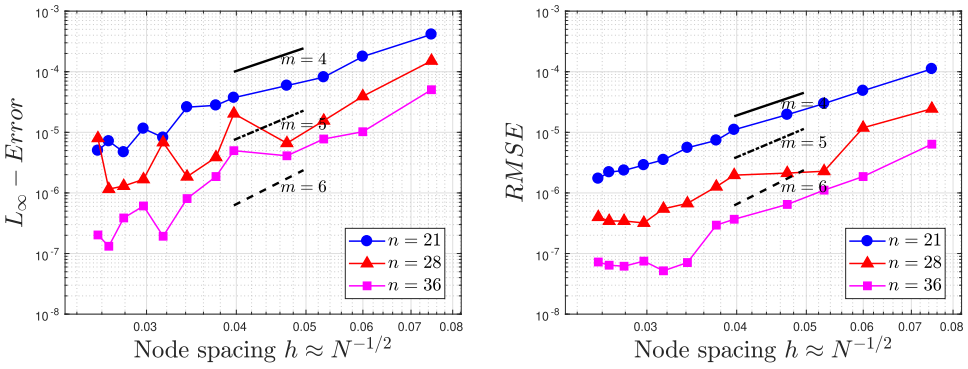

Fig. 15. Convergence orders for SLBDIM.

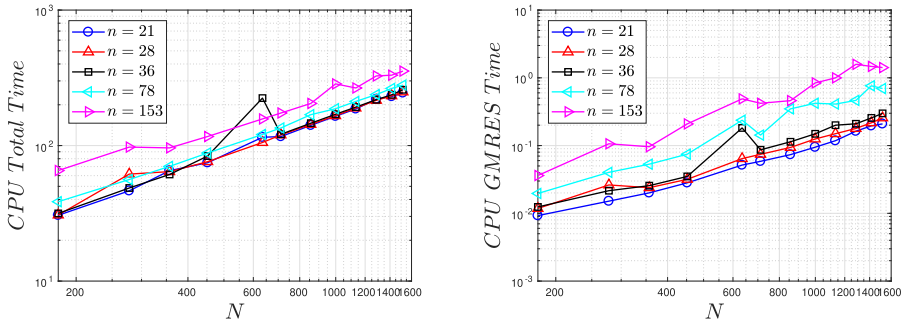

Fig. 16. CPU total time (left) and CPU GMRES time (right) versus number of interior nodes N_{int} for different stencil sizes $n = 21, 28, 36, 78, 153$.

Table 7
Results obtained for Case 3 via LBDIM with number of interior discretization points $N = 278, 1286$ and stencil sizes $n = 21, 28, 36$.

N_{int}	n	ε	$RMSE$	CN	CN GMRES	Time (s)
1286	21	1.4	1.26e-04	1.70e+11	1.37e+02	5.51
	28	1.9	1.29e-04	1.11e+11	1.60e+02	6.19
	36	2.2	1.01e-04	3.78e+11	1.54e+02	6.58
278	21	0.8	1.37e-04	2.64e+10	8.50e+01	1.00
	28	1.1	1.42e-04	1.10e+10	9.80e+01	1.31
	36	1.2	1.11e-04	5.63e+10	1.13e+02	1.36

10^{10} and 10^{11}, whereas the CN of the assembled matrix is of orders 10^1 and 10^2. Both are many orders of magnitude lower than the CN for the method reported in [31].

4.4. Case 4

Helmholtz equation with a high-frequency source

In this final numerical experiment, we consider a Helmholtz equation with Dirichlet boundary conditions. The equation and BCs are defined over the circular domain $\Omega = \{(x, y) \in \mathbb{R}^2 / x^2 + y^2 \leq 1\}$ show in Fig. 17

$$\begin{cases} \Delta u + k^2 u = f, & (x, y) \in \Omega, \\ u = g, & (x, y) \in \Gamma = \partial\Omega. \end{cases} \tag{42}$$

The source term is a non-trivial, highly oscillatory right-hand side, $f(x, y) = (k^2 - 200)\sin[10(x + y)]$, and the exact solution is $u(x, y) = \sin(10(x + y))$. The wave number considered is $k = 5$.

This numerical example is solved for a domain discretizations based on the repel algorithm mentioned in [32], corresponding to $h = 0.05$ and $N_{int} = 1185$ interior nodes and $N_b = 125$ boundary nodes, respectively. The left subplot in Fig. 17 shows the nodes distribution for $h = 0.05$ and the right subplot shows the oscillatory behavior of the analytical solution of Eq. (42) over the unit disk.

Fig. 18 illustrates the convergence of the numerical solution in terms of the L_∞-Error as a function of the stencil size n. The influence of the stencil size and the shape parameter ε on the global accuracy is analyzed. The log-log plot reveals an algebraic convergence behavior, indicated by the approximately linear slope of the error curves. We observe that the convergence rate is

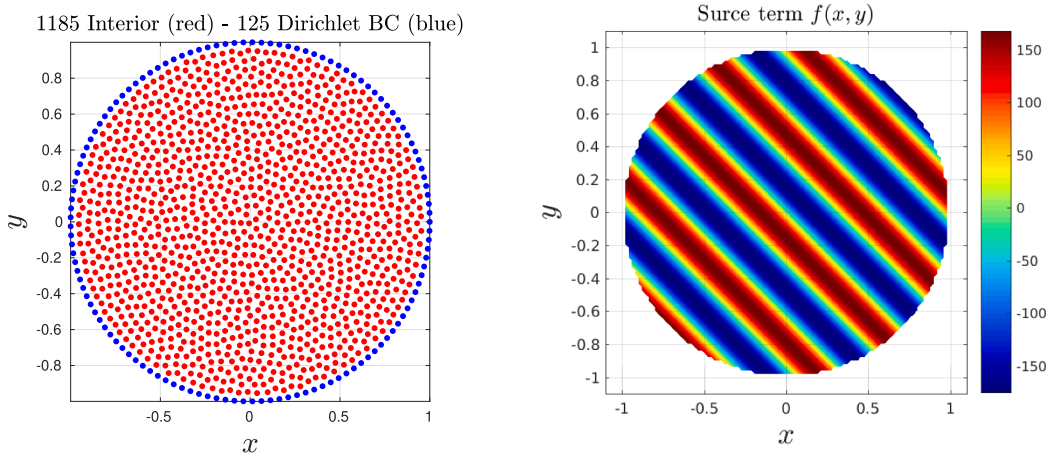

Fig. 17. Repel node distribution (left) and exact solution (right).

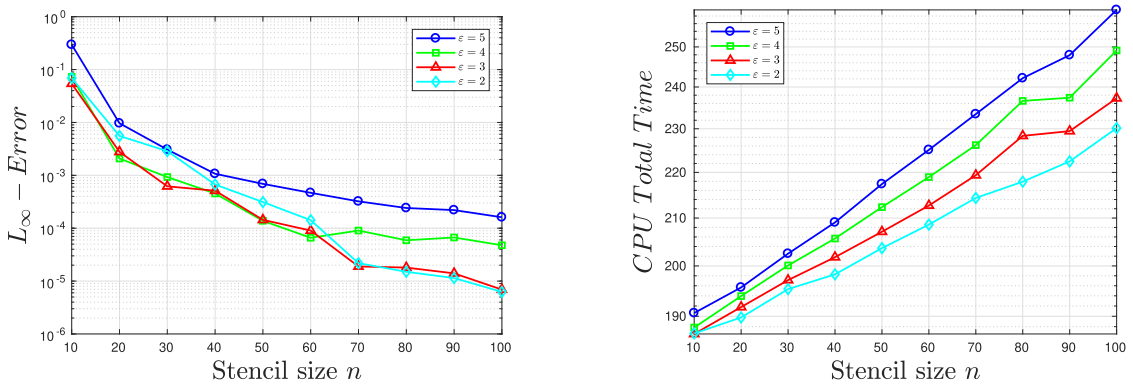

Fig. 18. L_∞-Error for the Helmholtz equation over circular domain.

sensitive to the choice of ε. The configuration with $\varepsilon = 5$ exhibits the shallowest slope, indicating a lower order of convergence. Conversely, lowering the shape parameter to $\varepsilon = 2, 3$ steepens the convergence slope, allowing the method to reach an accuracy of $\mathcal{O}(10^{-6})$ when n approaches 100. The computational efficiency of the method is analyzed in Fig. 18 (right). This plot reports the total CPU time required for the simulation as the stencil size varies from $n = 10$ to $n = 100$. As expected, the computational cost increases almost linearly with the size of the local stencil. As it is observed, the simulation with $\varepsilon = 5$ is the most computationally expensive, whereas $\varepsilon = 2$ consistently yields the lowest execution times across the entire range of n achieving an error of order 10^{-6}.

5. Summary

In this paper, we introduce a new stabilized local integral method for solving two-dimensional Helmholtz problems, and discuss their numerical solutions using quasi-uniform node distributions over different domains and geometries with corners. Numerical results for Helmholtz-type equations have been obtained using the SLBDIM, which involves local interpolations with Gaussian RBFs. This is a meshless method that can be adapted to problems involving complex geometries. Its effectiveness has been demonstrated through numerical results, as shown in four cases, and compared with other results in the literature and other mesh-dependent methods like the FEM. Case 1 demonstrates the benefit of using the SLBDIM to identify stability regions of convergence of L_2-Error, achieving an order of magnitude of 10^{-8} as the shape parameter ε approaches zero. Case 2 examines a range of low shape parameter values that produce low numerical errors. The method is also tested with increasing wavenumber demonstrating its robustness. Case 3 discusses the convergence rate of the method over amoeba-like domain, showing monotonic fourth-, fifth-, and sixth-order convergence, for stencil sizes $n = 21, 28, 36$, respectively. Case 4 analyzes a problem with a high-frequency source over a circular domain achieving an L_∞-Error of order 10^{-6}. When applied to Helmholtz equations, this new method enables us to obtain accurate results in a stable manner and address geometries from arbitrary domains for which analytical solutions do not exist. This opens up the possibility of solving wave chaos and dielectric microresonators in real-world physical problems.

CRediT authorship contribution statement

L. Ponzellini Marinelli: Conceptualization, Methodology, Software, Formal analysis, Investigation, Resources, Writing – original draft; **L. Raviola:** Conceptualization, Formal analysis, Investigation, Writing – review & editing.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Ponzellini Marinelli, Luciano reports financial support was provided by National University of Rosario. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors sincerely thank the two anonymous referees for their valuable comments and suggestions, which enabled to significantly improve the quality of this paper. Moreover, this work was supported by the Research Project 80020220600057UR at the National University of Rosario. The authors would also like to thank Martín Armoa and the Rosario Institute of Physics, National Scientific and Technical Research Council, Rosario, Argentina.

References

[1] H. Cao, J. Wiersig, Dielectric microcavities: model systems for wave chaos and non-Hermitian physics, Rev. Mod. Phys. 87 (2015) 61–111.
[2] C. Loeffler, W. Mansur, H.M. Barcelos, B.A. Solving, Helmholtz problems with the boundary element method using direct radial basis function interpolation, Eng. Anal. Bound. Elem. 61 (2015) 218–225.
[3] L. Trefethen, Spectral Methods in Matlab, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.
[4] C.-I. Davis, B. Fornberg, A spectrally accurate numerical implementation of the Fokas transform method for Helmholtz-type PDEs, Complex Var. Elliptic Equ. 59 (2014) 564–577.
[5] S. Shiralashetti, M. Kantli, A. Deshi, A new wavelet multigrid method for the numerical solution of elliptic type differential equations, Alex. Eng. J. 57 (2018) 203–209.
[6] C. Brebbia, D. Dominguez, Boundary Elements. An Introductory Course. 2nd Ed., WIT Press, Computational Mechanics Publications, Southampton and Boston, 1998.
[7] P. Partridge, C. Brebbia, L. Wrobel, The Dual Reciprocity Boundary Element Method, Computational Mechanics Publications co-published with Elsevier Applied Science, Southampton Boston, 1992.
[8] G. Fasshauer, Meshfree Approximation Methods with MATLAB, World Scientific Publishing Co., Hackensack, NJ, USA, 2007.
[9] G. Fasshauer, M. McCourt, Kernel-based Approximation Methods using MATLAB, World Scientific Publishing Co., Hackensack, NJ, USA,2015.
[10] B. Fornberg, N. Flyer, A Primer on Radial Basis Functions with Applications to the Geosciences, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2015.
[11] B. Fornberg, N. Flyer, Fast generation of 2-D node distributions for mesh-free PDE discretizations, Comput. Math. Appl. 69 (2015) 531–544.
[12] H. Barcelos, C. Loeffler, L. Lara, The direct interpolation boundary element method and the domain superposition technique applied to piecewise Helmholtz's problems with internal heterogeneity, Eng. Anal. Bound. Elem. 131 (2021) 41–50.
[13] F. Gholampour, E. Hesameddini, A. Taleei, A global RBF-QR collocation technique for solving two-dimensional elliptic problems involving arbitrary interface, Eng. Comput. 37 (2021) 3793–3811.
[14] E. Larsson, U. Sundin, A investigation of global radial basis function collocation methods applied to Helmholtz problems, Dolomites Res. Notes Approx. 13 (2020) 65–85.
[15] J. Lin, W. Chen, K. Sze, A new radial basis function for Helmholtz problems, Eng. Anal. Bound. Elem. 36 (2012) 1923–1930.
[16] Y. Hong, J. Lin, W. Chen, A typical backward substitution method for the simulation of Helmholtz problems in arbitrary 2D domains, Eng. Anal. Bound. Elem. 93 (2018) 167–176.
[17] P. Mishra, G. Fasshauer, M. Sen, L. Ling, A stabilized radial basis-finite difference (RBF-FD) method with hybrid kernels, Comput. Math. Appl. 77 (2019) 2354–2368.
[18] H. Hosseinzadeh, A. Shirzadi, On optimal radius of sub-domains in meshless LBIE method, Math. Comput. Simul. 213 (2023) 145–160.
[19] L. Ponzellini Marinelli, N. Caruso, M. Portapila, A stable computation on local boundary-domain integral method for elliptic PDEs, Math. Comput. Simul. 180 (2021) 379–400.
[20] M. Greenberg, Applications of Green's Functions in Science and Engineering, Dover Publications, Mineola, New York, 2015.
[21] E. Larsson, B. Fornberg, A numerical study of some radial rasis function based solution methods for elliptic PDEs, Comput. Math. Appl. 46 (2003) 891–902.
[22] R. Schaback, Small errors imply large evaluation instabilities, Adv. Comput. Math. 49 (2023) 1–27.
[23] G. Fasshauer, J. Zhang, On choosing "optimal" shape parameters for RBF approximation, Numer. Algorithms 45 (2007) 345–368.
[24] J. Biazar, M. Hosami, An interval for the shape parameter in radial basis function approximation, Appl. Math. Comput. 315 (2017) 131–149.
[25] B. Fornberg, E. Larsson, N. Flyer, Stable computations with Gaussian radial basis functions, SIAM J. Sci. Comput. 33 (2011) 869–892.
[26] E. Larsson, E. Lehto, A. Heryudono, B. Fornberg, Stable computation of differentiation matrices and scattered node stencils on Gaussian radial basis functions, SIAM J. Sci. Comput. 35 (2013) A2096–A2119.
[27] G. Wright, B. Fornberg, Stable computations with flat radial basis functions using vector-valued rational approximations, J. Comput. Phys. 331 (2017) 137–156.
[28] L. Ponzellini Marinelli, Stabilizing radial basis functions techniques for a local boundary integral method, Rev. Unión Mat. Argent. 64 (2023) 375–396.
[29] Y. Saad, Iterative Methods for Sparse Linear Systems, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.
[30] E. Larsson, A domain decomposition method for the Helmholtz equation in a multilayer domain, SIAM J. Sci. Comput. 20 (1999) 1713–1731.

[31] R. Cavoretto, A. De Rossi, An adaptive LOOCV-based refinement scheme for RBF collocation methods over irregular domains, Appl. Math. Lett. 103 (2020) 106178.
[32] V. Bayona, N. Flyer, B. Fornberg, G. Barnett, On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs, J. Comput. Phys. 332 (2017) 257–273.
[33] M. Nielsen, B. Fornberg, High-order numerical method for solving elliptic partial differential equations on unfitted node sets, Comput. Math. Appl. 193 (2025) 103–116.