Applied Mathematical Modelling 113 (2023) 241–261

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

A meshfree point collocation method for elliptic interface problems

Heinrich Kraus[a,b,*], Jörg Kuhnert[a], Andreas Meister[b], Pratik Suchde[a,c]

[a] Fraunhofer ITWM, Fraunhofer-Platz 1, Kaiserslautern 67663, Germany
[b] Universität Kassel, Heinrich-Plett-Straße 40, Kassel 34132, Germany
[c] University of Luxembourg, 2 Av. de l'Universite, Esch-sur-Alzette 4365, Luxembourg

ARTICLE INFO

Article history:
Received 12 April 2022
Revised 28 July 2022
Accepted 1 August 2022
Available online 4 August 2022

Keywords:
Diffusion operator
Discontinuous coefficients
Meshfree
Collocation
Generalized finite difference method

ABSTRACT

We present a meshfree generalized finite difference method for solving Poisson's equation with a diffusion coefficient that contains jump discontinuities up to several orders of magnitude. To discretize the diffusion operator, we formulate a strong form method that uses a smearing of the discontinuity; and a conservative formulation based on locally computed Voronoi cells. Additionally, we propose a novel conservative formulation for enforcing Neumann boundary conditions that is compatible with the conservative formulation of the diffusion operator. Finally, we introduce a way to switch from the strong form to the conservative formulation to obtain a locally conservative and positivity preserving scheme. The presented numerical methods are benchmarked against four test cases of varying complexity and jump magnitude on point clouds with nodes that are not aligned to the discontinuity. Our results show that the new hybrid method that switches between the two formulations produces better results than the classical generalized finite difference approach for high jumps in diffusivity.

© 2022 Elsevier Inc. All rights reserved.

1. Introduction

The present work lays the foundation for simulating phase change processes with a meshfree generalized finite difference method (GFDM) [1,2]. GFDMs have been proven to be broadly applicable for different applications, from solving elliptic problems [3], transient heat conduction problems [4], shallow water equations [5] to Navier–Stokes equations in industrial applications [6]. Our eventual goal, beyond the scope of the present work, is to use a meshfree GFDM in a Lagrangian framework [7] for phase change processes in a one-fluid model, which describes a multiphase flow in a monolithic approach without explicitly distinguishing between the different phases or tracking the interfaces between them [8].

There exist numerous approaches for simulating phase change processes. For problems with a known and usually fixed location of the interface, a common approach is a domain decomposition where the phases are treated as separate subdomains with interface boundary conditions connecting them [9,10]. Since we do not know the position of the interface in practical applications, computing a domain decomposition based on physical properties such as the temperature is required. However, this is computationally heavy and unfeasible in Lagrangian methods where a domain decomposition would need to be computed once or even multiple times per time step. Level-set methods are another common approach for simulating phase change processes [11]. For these types of methods, a level-set function serves as an identifier to determine to which phase a point in the computational domain belongs, and is treated as an additional variable in the system of PDEs. Level-set

* Corresponding author.
E-mail address: heinrich.kraus@itwm.fraunhofer.de (H. Kraus).

https://doi.org/10.1016/j.apm.2022.08.002
0307-904X/© 2022 Elsevier Inc. All rights reserved.

methods are also used for smeared-out or diffuse interfaces, but treating the level-set function as an additional variable also means that an initial level-set function needs to be set at the beginning of the simulation together with appropriate boundary conditions. However, it is often desirable for the interface between the two phases to emerge naturally without the necessity to specify the time and location of its emergence.

The monolithic one-fluid approach described earlier leads to jumps in material coefficients potentially of several orders of magnitude. To illustrate this, let us consider the case of freezing water. Near the freezing temperature, the viscosity of liquid water is $\eta \approx 1 \times 10^{-3} Pas$ whereas the viscosity of ice cannot be determined since it exhibits no flow. To be able to simulate the freezing process, we might use the viscosity of pitch as a replacement for the viscosity of ice. Pitch is a material that appears to be solid but in reality, it is viscoelastic with a viscosity of $\eta \approx 2 \times 10^{8} Pas$ [12]. If we were to model the viscosity of ice to be of a similar magnitude, we would obtain a jump of 1×10^{11} in the phase change region. Viscosity and other material coefficients, such as thermal conductivity and density, appear as a diffusivity in diffusion operators of the form $\nabla \cdot (\eta \nabla u)$. This motivated us to study Poisson's equation in divergence form with a discontinuous diffusion parameter in more detail. For such elliptic interface problems, there exist several ways of discretizing the diffusion operator with jumping diffusion coefficients. One common approach is the use of enrichment which has been used in meshfree methods [13,14] and finite element methods [15,16] alike. The goal of enrichment is to extend the basis of the function space, usually by discontinuous functions, and to achieve more accurate results that way. In present literature that deals with meshfree methods, the location of the interface is usually known [9] and the considered jumps are only of a few orders of magnitude [17–19].

In our formulation of the GFDM, we approximate strong form differential operators based on local neighborhoods. However, it fails to ensure flux conservation in its core formulation which can lead to numerical instabilities. Several approaches have been pursued to eradicate this issue but only offer partial solutions. Suchde et al. [20] provide a weaker notion of flux conservation. Other approaches have been presented [19,21] to establishing differential operators globally that violate the local nature of differential operators and thus are too expensive to be used for time-varying point clouds.

In this paper, we introduce a method that does not depend on any a priori knowledge of the interface. It strives to identify the interface based on algebraic properties from the strong formulation of the discrete diffusion operator and switches over to a conservative method that is based on locally computed control volumes. In the finite volume method, such control volumes are based on a globally defined mesh. In contrast, the control volumes here can be computed locally [20].

This paper is organized as follows. In Section 2, we formulate Poisson's equation in divergence form and give a brief introduction to the generalized finite difference method. After that, we establish properties that our numerical methods have to fulfill; and then develop the strong form weighted-least-squares and the conservative control volume-based discretization of the diffusion operator. Finally, we introduce a way of combining these approaches into a hybrid formulation. These methods are benchmarked in Section 3 against four test cases with increasing complexity. We conclude the paper with a brief discussion of the methods and numerical results in Section 4 and a conclusion in Section 5.

2. Method formulation

We consider Poisson's equation in divergence form

$$-\nabla \cdot (\eta \nabla u) = f \tag{1}$$

on a domain $\Omega \subset \mathbb{R}^d$ with a diffusivity $\eta = \eta(\boldsymbol{x}) > 0$. Additionally, we set Dirichlet and Neumann boundary conditions

$$u|_{\partial\Omega_D} = g_D, \tag{2a}$$

$$\left.\frac{\partial u}{\partial \boldsymbol{n}}\right|_{\partial\Omega_N} = g_N \tag{2b}$$

with a disjoint partition of the boundary $\partial\Omega$ into $\partial\Omega_D$ and $\partial\Omega_N$. We also use the abbreviatory notation

$$Lu := \nabla \cdot (\eta \nabla u). \tag{3}$$

Since we lay the focus on diffusion coefficients with jumps on a $d-1$-dimensional manifold $\Gamma \subset \Omega$, we are looking for solutions $u \in H^1(\Omega)$ [22] that satisfy the jump conditions

$$[\![u]\!]_\Gamma = 0, \tag{4a}$$

$$[\![\eta \nabla u \cdot \boldsymbol{n}]\!]_\Gamma = 0. \tag{4b}$$

The vector $\boldsymbol{n}$ denotes the normal vector to Γ and the notation $[\![\cdot]\!]_\Gamma$ describes the jump of a function on the set Γ

$$[\![f]\!]_\Gamma : \Gamma \to \mathbb{R}, \quad \boldsymbol{x} \mapsto \lim_{\boldsymbol{y}\to\boldsymbol{x}^+} f(\boldsymbol{y}) - \lim_{\boldsymbol{y}\to\boldsymbol{x}^-} f(\boldsymbol{y})$$

where the one-sided limit from each side of the set Γ is represented by $\boldsymbol{y} \to \boldsymbol{x}^{\pm}$.

2.1. Generalized finite difference method

In our formulation of the GFDM, we use a finite set of points

$$\Omega_h = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\} \subset \overline{\Omega} = \Omega \cup \partial\Omega$$

together with an interaction radius $h : \Omega \to \mathbb{R}$ for spatially discretizing the domain Ω. For an arbitrary function $u : \Omega \to \mathbb{R}$ we use the abbreviatory notation $u_i = u(\boldsymbol{x}_i)$ and $\boldsymbol{u} = (u_1, \dots, u_N)^T$. The interaction radius is also sometimes called smoothing length and induces a distance function $d : \Omega \times \Omega \to [0, \infty)$ that measures the relative distance between points in the point cloud. Usually, a scaled Euclidean distance is used, for example

$$d_1(\boldsymbol{x}_j, \boldsymbol{x}_i) = \frac{\left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|}{h_i}$$

or

$$d_2(\boldsymbol{x}_j, \boldsymbol{x}_i) = \frac{2\left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|}{h_j + h_i}.$$

The distance function enables us to define neighborhoods

$$B_i = \{\boldsymbol{x}_j \in \Omega_h | d(\boldsymbol{x}_j, \boldsymbol{x}_i) \leq 1\} \tag{5}$$

and their respective index sets

$$S_i = \{j | \boldsymbol{x}_j \in B_i\} \tag{6}$$

for each point $\boldsymbol{x}_i$. Neighborhoods are called symmetric if

$$j \in S_i \Rightarrow i \in S_j$$

is satisfied for each point. Hence, by using d_1 we obtain radial but not necessarily symmetric neighborhoods while d_2 leads to symmetric but not necessarily radial neighborhoods. Neighborhoods are used to establish a notion of connectivity between points, based on which numerical computations are executed.

In the strong form GFDM, a linear differential operator D^* is discretized at a point $\boldsymbol{x}_i$ by using a numerical representation D_i^* that is given by coefficients c_{ij}^* such that

$$D^* u(\boldsymbol{x}_i) \approx D_i^* u = \sum_{j \in S_i} c_{ij}^* u_j.$$

The asterisk is a placeholder for any differential operator, such as the Laplacian Δ or the partial derivative ∂_{x_k} with respect to variable x_k. To compute the coefficients c_{ij}^*, we impose exact reproducibility of monomial test functions

$$D^*(\boldsymbol{x} - \boldsymbol{x}_i)^{\boldsymbol{\alpha}} = \sum_{j \in S_i} c_{ij}^* (\boldsymbol{x}_j - \boldsymbol{x}_i)^{\boldsymbol{\alpha}} \tag{7}$$

for all multi-indices $\boldsymbol{\alpha} \in \mathbb{N}_0^d$ with $|\boldsymbol{\alpha}| \leq K$. Here, the usual notations $\boldsymbol{x}^{\boldsymbol{\alpha}} = \prod_{n=1}^d x_n^{\alpha_n}$ and $|\boldsymbol{\alpha}| = \sum_{n=1}^d \alpha_n$ are used. This approach generally leads to fewer conditions than there are points in the neighborhood resulting in an under-determined linear system. To be able to solve the linear system, we add a minimization constraint on the coefficients

$$\min \sum_{j \in S_i} \left(\frac{c_{ij}^*}{w_{ij}} \right)^2, \tag{8}$$

with weights

$$w_{ij} = \exp\left(-2 \frac{\left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|}{h_i + h_j} \right). \tag{9}$$

Other types of weight functions have been discussed in the literature, see for example Jacquemin et al. [23]. They show that the approximation error also depends on the weight function. However, in our case, the main source of error is not the choice of weight function but the discontinuity of the diffusion coefficient. We reformulate Eqs. (7) and (8) into an optimization problem in matrix form

$$\min \left\| \boldsymbol{W}_i^{-1} \boldsymbol{c}_i^* \right\| \tag{10a}$$

$$\text{s.t.} \quad \boldsymbol{K}_i \boldsymbol{c}_i^* = \boldsymbol{b}_i^* \tag{10b}$$

where the solution is given by

$$\boldsymbol{c}_i^* = \boldsymbol{W}_i^2 \boldsymbol{K}_i^T (\boldsymbol{K}_i \boldsymbol{W}_i^2 \boldsymbol{K}^T)^{-1} \boldsymbol{b}_i^*. \tag{10c}$$

Here the weights w_{ij} are written in the $|S_i| \times |S_i|$ diagonal matrix $\boldsymbol{W}_i$. Each row of matrix $\boldsymbol{K}_i$ represents one test function applied to the points in the neighborhood and $\boldsymbol{b}_i^*$ are the exact evaluations of the differential operators to the respective monomial test function.

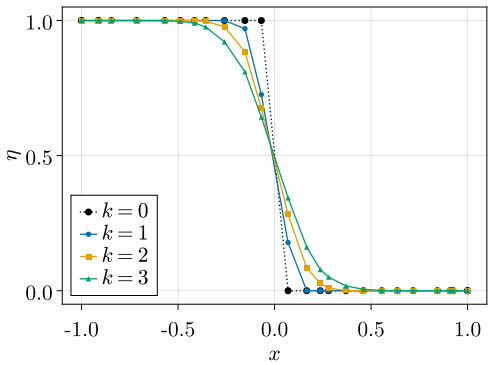


Fig. 1. Qualitative smoothing comparison of a piecewise constant function and k smoothing cycles.

2.2. Discretization of the diffusion operator

For the diffusion operator (3), we seek coefficients γ_{ij} such that we obtain an approximation of the form

$$Lu(\boldsymbol{x}_i) \approx L_i u = \sum_{j \in S_i} \gamma_{ij} u_j. \tag{11}$$

The coefficients γ_{ij} form a matrix $\boldsymbol{G}$ to obtain the discrete Poisson's equation

$$\boldsymbol{G}\boldsymbol{u} = \boldsymbol{f} \tag{12}$$

where the boundary conditions are included in $\boldsymbol{G}$ and $\boldsymbol{f}$. The maximum principle

$$-Lu \geq 0 \Rightarrow u \geq 0 \tag{13}$$

holds even for low regularity of the diffusivity η and source term f [22], and often the numerical schemes satisfy it discretely [24]. However, GFDM in its current strong form approach has no means to guarantee the discrete maximum principle

$$\boldsymbol{G}\boldsymbol{u} \geq \boldsymbol{0} \Rightarrow \boldsymbol{u} = \boldsymbol{G}^{-1}\boldsymbol{f} \geq \boldsymbol{0} \tag{14}$$

without solving a global linear system [25].

To guarantee the maximum principle discretely while maintaining the local nature of GFDM, one way is to enforce the sign condition

$$\gamma_{ii} < 0, \tag{15a}$$

and the diagonal dominance criterion

$$\sum_{\substack{j \in S_i \\ j \neq i}} \left|\gamma_{ij}\right| \leq |\gamma_{ii}| \tag{15b}$$

upon the coefficients in Eq. (11). The conditions in Eq. (15) lead to an M-matrix and thus inverse-positive matrix $\boldsymbol{G}$ that guarantees the discrete maximum principle (14), see Seibold [26].

2.2.1. Classical operator

For the strong form GFDM, we assume that the diffusivity parameter η is sufficiently smooth. To be able to use the strong form for discontinuous diffusivities, we smooth the diffusivity η by using weights

$$s_{ij} = \exp\left(-3\frac{\left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|^2}{h_i^2}\right).$$

in a convex combination of function values over the neighborhood

$$\tilde{\eta}_i = \frac{\sum_{j \in S_i} s_{ij}\eta_j}{\sum_{j \in S_i} s_{ij}}.$$

The smoothing process can be applied multiple times, leading to an incrementally increasing smearing of the jump. Figure 1 shows a comparison of a piecewise constant function with a different number of smoothing cycles and in Fig. 2, a discontinuous function with its smoothed equivalent is depicted.

For a smoothed diffusivity $\tilde{\eta}$, we apply the product rule on the diffusion operator (3) to obtain

$$Lu \approx \nabla \cdot (\tilde{\eta}\nabla u) = \nabla\tilde{\eta} \cdot \nabla u + \tilde{\eta}\Delta u. \tag{16}$$

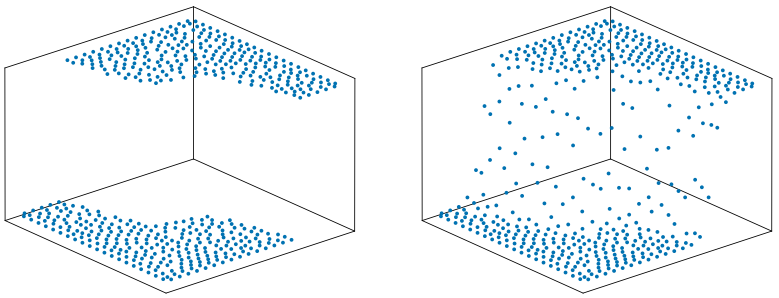


Fig. 2. Comparison of discontinuous function (left) and its smoothed counterpart (right) with 2 smoothing cycles.

Alternatively, it is sometimes advantageous to use the scaled diffusivity parameter $\mu = \log \eta$, especially when there are large jumps in η. Scaling the diffusivity that way yields, after potential smoothing steps, the diffusion operator

$$Lu = \nabla \cdot (\exp(\mu)\nabla u) \approx \nabla \cdot (\exp(\tilde{\mu})\nabla u) = \exp(\tilde{\mu})(\nabla \tilde{\mu} \cdot \nabla u + \Delta u). \tag{17}$$

By using monomial test functions of degree 1 or higher, the gradient of either $\tilde{\eta}$ or $\tilde{\mu}$ has to be computed numerically. The numerical computation of a smoothed and thus distorted diffusivity can be a source of inaccuracy for this method.

The coefficients γ_{ij} are then computed by solving the optimization problem (10). Since the optimization problem does not impose diagonal dominance, this can lead to non-diagonally dominant rows and, in some cases, instabilities in the numerical solution [25]. To reduce numerical instabilities, a correction technique has been developed where a correction vector $\boldsymbol{\xi}_i$ is computed that lies in the null space of the test function space

$$\sum_{j \in S_i} \xi_{ij} (\boldsymbol{x}_j - \boldsymbol{x}_i)^{\boldsymbol{\alpha}} = 0$$

for all considered monomials. To obtain a non-trivial zero operator, an additional constraint $\xi_{ii} = 1$ is added and optimization problem (10) is solved. With the correction vector $\boldsymbol{\xi}_i$, we are able to define a new operator

$$\boldsymbol{a}_i = \boldsymbol{\gamma}_i + \alpha \boldsymbol{\xi}_i, \tag{18}$$

that satisfies the same monomial reproducibility properties as $\boldsymbol{\gamma}_i$ for each $\alpha \in \mathbb{R}$. In the next step, we minimize the function

$$\phi(\alpha) = \sum_{j \in S_i} \frac{(\gamma_{ij} + \alpha \xi_{ij})^2}{(\gamma_{ii} + \alpha \xi_{ii})^2}$$

that attains its global minimum at

$$\alpha_{\min} = \frac{\gamma_{ii} \langle \boldsymbol{\gamma}_i, \boldsymbol{\xi}_i \rangle - \xi_{ii} \langle \boldsymbol{\gamma}_i, \boldsymbol{\gamma}_i \rangle}{\xi_{ii} \langle \boldsymbol{\gamma}_i, \boldsymbol{\xi}_i \rangle - \gamma_{ii} \langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_i \rangle}.$$

This correction technique does not guarantee diagonally dominant operators and there are numerous examples of neighborhoods where this technique does not provide diagonally dominant rows, some of which are shown in Appendix A. Nevertheless, it has been shown that the stability of the discrete Laplace operator increases by using this technique [25].

2.2.2. Conservative operator

For the conservative approach, we omit optimization problem (10) and use Voronoi cells

$$\Omega_i = \{\boldsymbol{x} \in \Omega \,|\, \|\boldsymbol{x} - \boldsymbol{x}_i\| < \left\|\boldsymbol{x} - \boldsymbol{x}_j\right\| \text{ for each } \boldsymbol{x}_j \in \Omega_h \setminus \{\boldsymbol{x}_i\}\} \tag{19}$$

instead. For a Voronoi cell Ω_i, the area in 2D and volume in 3D is given by

$$|\Omega_i| = \frac{1}{2d} \sum_{j \in S_i} \left|\Gamma_{ij}\right| \|x_j - x_i\|$$

where the boundary of the cell is represented as $\partial\Omega_i = \cup_j \Gamma_{ij}$ with $\Gamma_{ij} = \overline{\Omega}_i \cap \overline{\Omega}_j$. The size of Γ_{ij}, a line segment in 2D and a polygon in 3D, is denoted by $\left|\Gamma_{ij}\right|$.

For an inner point $\boldsymbol{x}_i \in \Omega$, we use a cell average [27] to compute the discrete diffusion operator

$$|\Omega_i| L_i u = \int_{\Omega_i} \nabla \cdot (\eta \nabla u) \mathrm{d}\boldsymbol{x}.$$

Applying the divergence theorem and exploiting the composition of the boundary of Ω_i into the faces Γ_{ij} leads on to

$$|\Omega_i| L_i u = \oint_{\partial\Omega_i} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S = \sum_{j \in S_i} \int_{\Gamma_{ij}} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S.$$

This is a representation of the diffusion operator in terms of fluxes over the surfaces of the Voronoi cells. Note that for each point $\boldsymbol{x}_j \in B_i$ that is not a Voronoi neighbor of $\boldsymbol{x}_i$, we use $\left|\Gamma_{ij}\right| = 0$. Applying Gaussian quadrature to each face leads to

$$\int_{\Gamma_{ij}} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S \approx \left|\Gamma_{ij}\right| \eta_{ij} \frac{u_j - u_i}{d_{ij}}$$

where $\eta_{ij} \approx \eta(\boldsymbol{x}_{ij})$ is an approximation of η at the midpoint

$$\boldsymbol{x}_{ij} = \frac{\boldsymbol{x}_i + \boldsymbol{x}_j}{2}$$

of the line segment

$$L_{ij} = \{\boldsymbol{x}_i + \theta(\boldsymbol{x}_j - \boldsymbol{x}_i) \mid \theta \in [0,1]\}$$

between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ and $d_{ij} = \left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|$ is the length of that line segment. Note that if the line segment L_{ij} and the surface Γ_{ij} intersect, then they intersect at $\boldsymbol{x}_{ij}$.

Finally, the coefficients γ_{ij} for the discrete diffusion operator read

$$\gamma_{ij} = \frac{\left|\Gamma_{ij}\right|}{|\Omega_i|} \frac{\eta_{ij}}{d_{ij}} \tag{20a}$$

where $i \neq j$ and for the diagonal entries

$$\gamma_{ii} = -\sum_{\substack{j \in S_i \\ j \neq i}} \frac{\left|\Gamma_{ij}\right|}{|\Omega_i|} \frac{\eta_{ij}}{d_{ij}}. \tag{20b}$$

These coefficients trivially fulfill diagonal dominance and the sign restriction from Eq. (15) such that they automatically lead to diagonally dominant rows and thus fulfill the discrete maximum principle (14).

In the case of meshfree GFDM, we have a fixed distribution of points $\boldsymbol{x}_i \in \Omega$ based on which we can calculate the Voronoi cells. The quality requirements imposed on point clouds are rather loose [28] in comparison to the strong restrictions on the meshes used for mesh-based methods [29]. Furthermore, the Voronoi cells used here are constructed locally from the neighborhoods B_i and can thus easily be used for moving point clouds as well.

Calculation of η_{ij}. To calculate the reconstructions η_{ij}, we use the harmonic mean

$$\eta_{ij} = \frac{2}{\frac{1}{\eta_i} + \frac{1}{\eta_j}}.$$

Alternatively, a discrete approximation operator can be computed with the optimization problem (10) that discretizes the identity mapping

$$I_{ij}\eta = \eta(\boldsymbol{x}_{ij}).$$

Similar operators are used for approximating function values at new points where the values were previously unknown. But for diffusion coefficients with large discontinuities, this can lead to oscillations and even negative reconstructions $\eta_{ij} < 0$.

Conservative Neumann boundary condition. Applying the cell average idea to boundary cells Ω_i that are associated with Neumann boundary points $\boldsymbol{x}_i \in \partial\Omega_N$, we obtain

$$\int_{\Omega_i} \nabla \cdot (\eta \nabla u) \mathrm{d}S = \int_{\Gamma_i} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S + \sum_{j \in S_i} \int_{\Gamma_{ij}} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S$$

where $\Gamma_i \subset \partial\Omega$ is a piece of the domain boundary associated with the boundary point $\boldsymbol{x}_i$. Using Eqs. (1) and (2b) finally leads to

$$-\int_{\Omega_i} f \mathrm{d}\boldsymbol{x} - \int_{\Gamma_i} \eta g_N \mathrm{d}S = \sum_{j \in S_i} \int_{\Gamma_{ij}} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S.$$

Dividing by $|\Omega_i|$ and using the coefficients in Eq. (20), we find

$$-f_i - \frac{|\Gamma_i|}{|\Omega_i|} \eta_i g_N(\boldsymbol{x}_i) = \sum_{j \in S_i} \gamma_{ij} u_j. \tag{21}$$

Discrete divergence theorem. For the divergence theorem

$$\int_{\Omega} \nabla \cdot (\eta \nabla u) \mathrm{d}\boldsymbol{x} = \oint_{\partial\Omega} \eta \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}S$$

to hold discretely, the coefficients γ_{ij} from Eq. (11) need to suffice to the column sum condition

$$\sum_{i=1}^{N} |\Omega_i| \gamma_{ij} = 0 \tag{22}$$

for interior points $\boldsymbol{x}_j \in \Omega$ [25]. The cell-based approach, given by Eq. (20) and (21), automatically satisfies Eq. (22) and fulfills the discrete divergence theorem on the entire domain. However, enforcing Eq. (22) in the row-based discrete operator calculation from Section (2.2.1), requires us to solve one huge optimization problem instead of solving many small optimization problems as in Eq. (10).

2.2.3. Hybrid

The computations of the Voronoi cells (19), even when done locally, can be costly and may not be necessary at all points. If the diffusion coefficient η is smooth in a certain region, it is appropriate to use the classical approach from Section 2.2.1 to obtain satisfactory results and only use the conservative method in regions where the diffusivity is discontinuous. To identify regions where η is discontinuous, we use the coefficients γ_{ij} obtained from the strong form method and define the diagonal dominance error

$$\sigma_i = \max\Big(\sum_{\substack{j\in S_i \\ j\neq i}} \frac{|\gamma_{ij}|}{|\gamma_{ii}|} - 1, \gamma_{ii}, 0\Big). \tag{23}$$

In the case that $\sigma_i \leq 0$, we use the coefficients γ_{ij} to discretize the diffusion operator, and for $\sigma_i > 0$ the coefficients are recomputed using the conservative formulation. This leads to a hybrid method that preserves the discrete maximum principle (14).

Better results can be achieved by using the conservative formulation not only for those points where $\sigma_i > 0$ but also for each neighboring point $j \in S_i$. This results in a conservative scheme for symmetric neighborhoods. Note that condition (23) checks only the diagonal dominance of a row. It can thus be triggered not only by high jumps in the diffusivity but also by local irregularities in the point cloud. Examples for such node configurations are presented in Appendix A.

Other jump identification techniques exist where the function is directly screened for discontinuities, see for example Allasia et al. [30], Gutzmer and Iske [31]. In their formulation, global properties of the examined function are used and thus these methods only work for one jump or multiple jumps of similar magnitudes.

3. Numerical results

In the preceding sections, we presented numerical methods for approximating the diffusion operator. With that, we introduced several parameters for computing the discrete differential operator. Parameters of special interest are

- number of smoothing cycles of the diffusivity,
- scaling of the diffusion coefficient

for the strong form method. For the hybrid method, we will take a closer look at

- the possibility of extending the conservative formulation to the neighbors of points where a lack of diagonal dominance was discovered,
- conservative Neumann boundaries.

In this section, we compare the different methods for select test cases in terms of convergence and susceptibility to jumps in the diffusivity. A performance comparison between the strong form method and the hybrid method is given in Appendix B.

On the rectangle $\Omega = [-1, 1] \times [-1, 1]$ we use two different types of point clouds, uniform and irregular point clouds, with increasing refinement levels $h_k = \frac{2^{-k}}{5}$ for $k = 0, \ldots, 5$. A comparison of the different point cloud types with similar refinements is shown in Figure 3. The irregular point clouds are generated with an advancing front method [32], using the software MESHFREE developed by Fraunhofer ITWM [33]. We set coefficients $r_{\min} = 0.25$ and $r_{\max} = 0.45$ such that the relative distance to the closest neighbor is bounded below by

$$r_{\min} \leq \min_{\substack{j\in S_i \\ j\neq i}} \frac{\|\boldsymbol{x}_j - \boldsymbol{x}_i\|}{h_i}$$

for each point $\boldsymbol{x}_i$. Additionally, we require each point in the domain $\boldsymbol{x} \in \Omega$ to fulfill the intersection condition

$$\overline{B(\boldsymbol{x}; r_{\max} h(\boldsymbol{x}))} \cap \Omega_h \neq \emptyset$$

that defines the maximum size of holes that appear in the point cloud. By $B(\boldsymbol{x}; r)$, we denote a ball around $\boldsymbol{x}$ with radius r with respect to the Euclidean metric. We used the irregular point clouds to test if and to what extent they influence the numerical results compared to uniform point clouds. In our experiments, we compared the numerical results from the different point cloud types and could observe a small difference in the discretization error. Since the convergence behavior was similar, we will only show results from irregular point clouds as seen in Fig. 3b and specifically state where a uniform point cloud was used. Because the point clouds are generated independently of the diffusivity, we are able to show that our method works well on point clouds that do not conform to the interface.

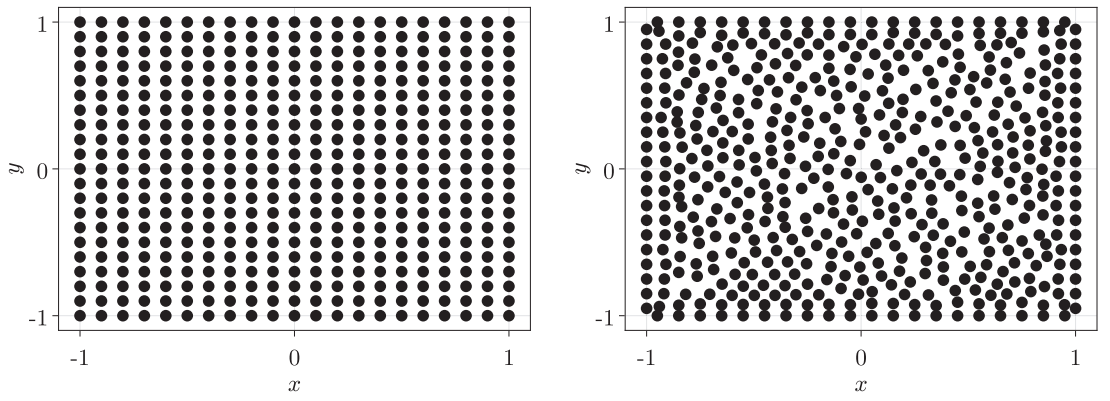


Fig. 3. Comparison of point cloud types.

To simplify notation, we decompose the boundary of Ω

$$\partial\Omega = \partial\Omega_L \cup \partial\Omega_R \cup \partial\Omega_T \cup \partial\Omega_B$$

into the left, right, top and bottom part respectively.

To analyze the convergence behavior of the presented methods, we compute the *relative* L^2 error of the numerical solution u_h to the analytical solution u

$$\|u - u_h\| = \frac{\|u - u_h\|_{L^2}}{\|u\|_{L^2}},$$

where the discrete L^2 norm for a function $f : \Omega \to \mathbb{R}$ is defined by

$$\|f\|_{L^2} = \left(\sum_{i=1}^{N} v_i |f(\boldsymbol{x}_i)|^2 \right)^{1/2}$$

and $v_i > 0$ are areas in 2D or volumes in 3D associated with each point $\boldsymbol{x}_i$.

3.1. Two-strip problem

For the two-strip problem, we solve the homogeneous Poisson's equation and define the diffusivity as a piecewise constant function

$$\eta(x, y) = \begin{cases} \eta_L, & x < 0, \\ \eta_R, & x > 0 \end{cases}$$

where η_L, $\eta_R > 0$. The Dirichlet and Neumann boundary conditions

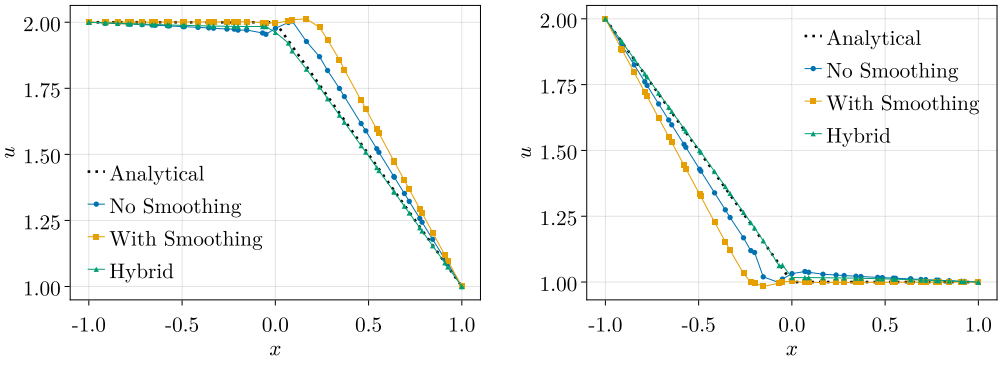
$$\begin{aligned} u|_{\partial\Omega_L} &= 2 \\ u|_{\partial\Omega_R} &= 1 \\ \left.\frac{\partial u}{\partial \boldsymbol{n}}\right|_{\partial\Omega_T} &= 0 \\ \left.\frac{\partial u}{\partial \boldsymbol{n}}\right|_{\partial\Omega_B} &= 0 \end{aligned}$$

lead to an analytical solution that is monotonously decreasing and piecewise linear in x-direction and constant in y-direction. We characterize a two-strip problem uniquely by the jump

$$\delta\eta = \frac{\eta_R}{\eta_L}.$$

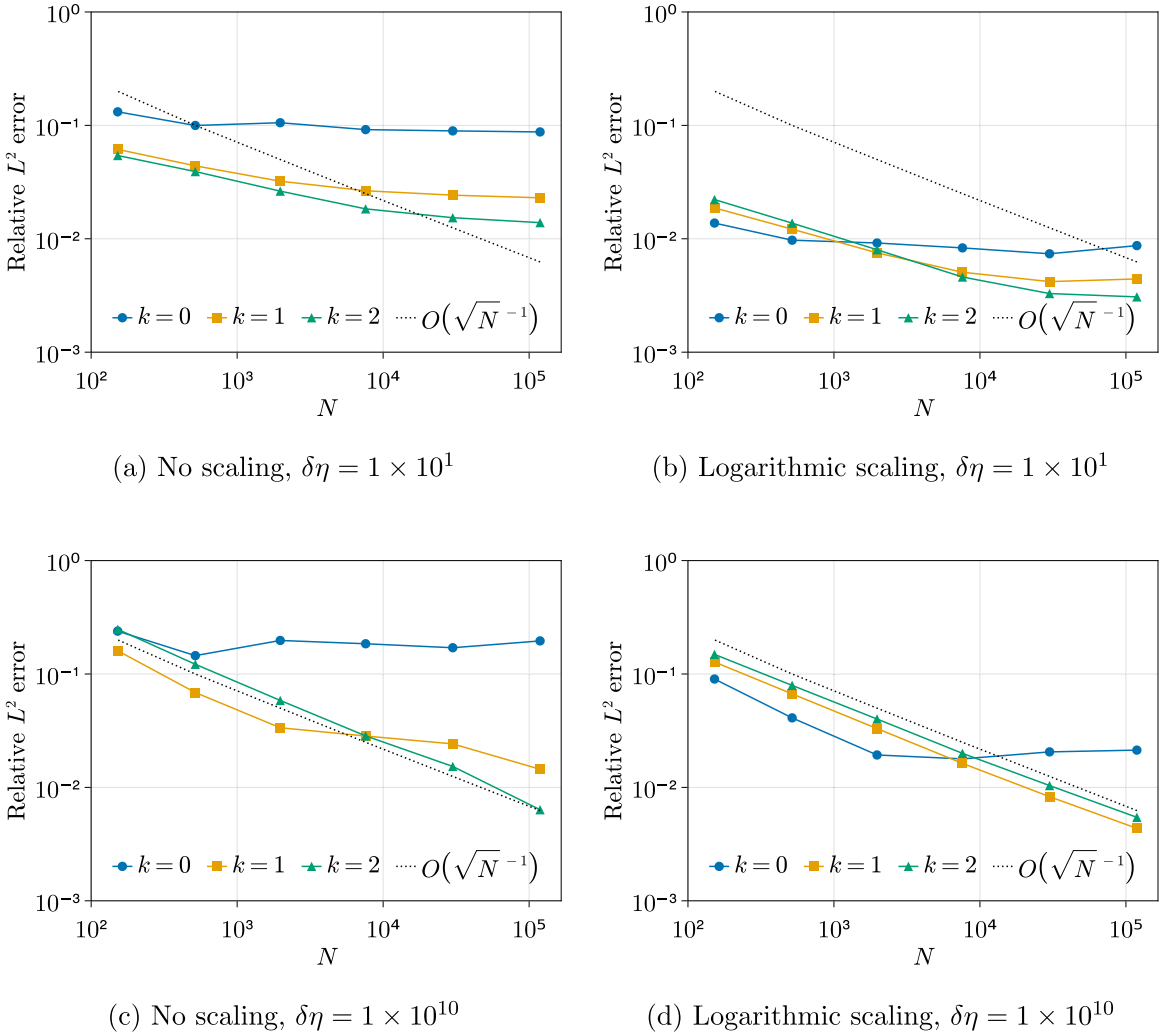
The solution profile of the analytical solution can be seen in Fig. 4 for a jump of $\delta\eta = 1 \times 10^{-8}$ and $\delta\eta = 1 \times 10^{8}$ respectively.

The purpose of this test case is to examine if smoothing and the logarithmic scaling of the diffusivity improve the numerical results. We produced results with zero, one, and two smoothing cycles. Figure 5 shows that the classical method without smoothing or scaling does not converge. Applying only the scaling, we observe that the method starts to converge for point clouds with a small point density but with an increasing number of points, the error stagnates. By smoothing



(a) $\delta\eta = 1 \times 10^{-8}$ (b) $\delta\eta = 1 \times 10^{8}$

Fig. 4. Solution profile at $y \approx 0$ to the two-strip test case with a jump of $\delta\eta$.



(a) No scaling, $\delta\eta = 1 \times 10^{1}$ (b) Logarithmic scaling, $\delta\eta = 1 \times 10^{1}$

(c) No scaling, $\delta\eta = 1 \times 10^{10}$ (d) Logarithmic scaling, $\delta\eta = 1 \times 10^{10}$

Fig. 5. Convergence plots of the classical strong form method for the two-strip test case with and without logarithmic scaling with different jumps $\delta\eta$ and number of smoothing cycles k.

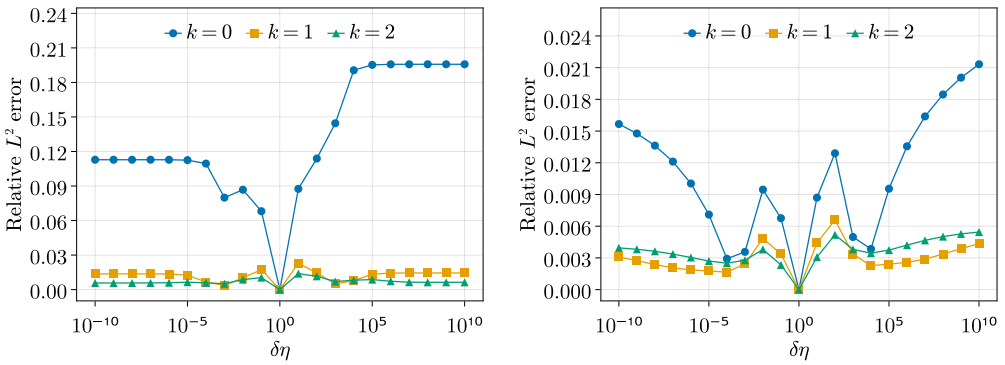


Fig. 6. L^2 errors for the two-strip test case depending on the jump $\delta\eta$.

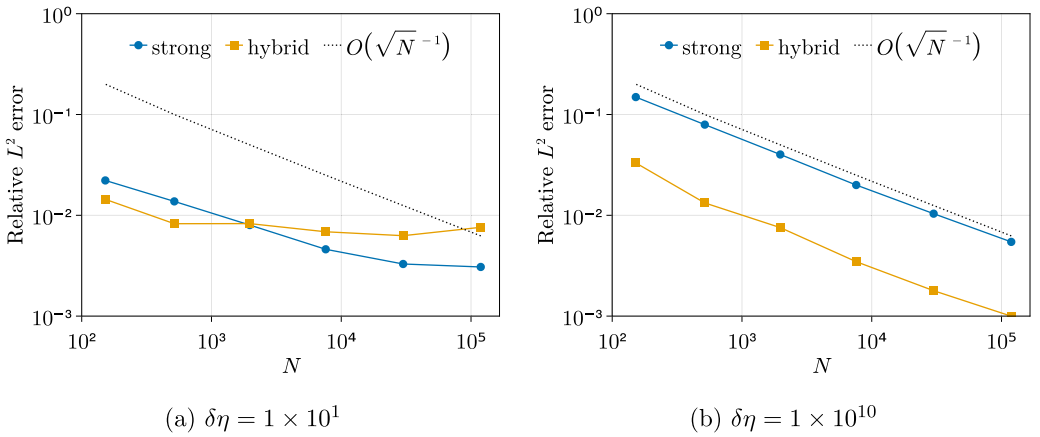
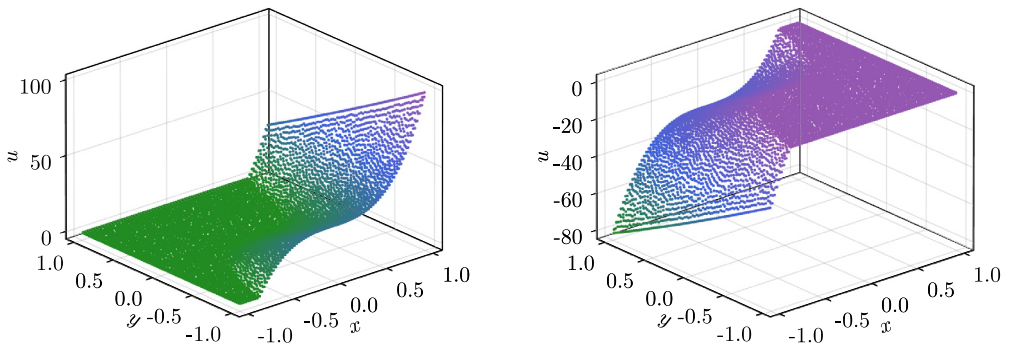


Fig. 7. L^2 errors of strong form method and hybrid method for the two-strip test case with different jump magnitudes.

the diffusivity, we observe numerous effects. First, the method has first-order convergence even without the scaling of the diffusivity. A reason for that is that by smoothing the diffusivity, the discontinuity is smeared out and the discrete gradient operates better. Secondly, if we additionally scale the diffusivity, the error is reduced for all methods. We also find that increasing the number of smoothing cycles does not necessarily lead to better results, see Fig. 5d, due to an increasing distortion of the model with an increasing number of smoothing cycles. Finally, we observe that for some point clouds, the non-smoothed diffusivity leads to better results than the smoothed diffusivity, as can be seen in Fig. 5b and d. The relative L^2 errors depending on the jump magnitude in Fig. 6 show that the methods with a smoothed diffusivity are less susceptible to high jump magnitudes. However, it also shows that the error can be reduced by a factor of 6 to 10 for the non-smoothed version and by a factor of approximately 4 for the smoothed versions by adding logarithmic scaling. The asymmetry of the error is due to the irregularity of the point cloud and because we compute the *relative* error. As seen in Fig. 4, the L^2 norm of the analytical solution is different for different jump magnitudes and thus the relative error differs.

With this information, we establish the “best” strong form method as the method that includes the scaling of the diffusivity and, depending on the magnitude of the jump, some smoothing of it. From the results, it can be doubted if the smoothing improves the numerical solution. But it stabilizes the convergence and additionally, the non-scaled version tends to have overshoots in the region of the interface. The smoothing causes the overshoot to smear out but not disappear, see Fig. 4. This behavior is characteristic of strong form methods and is also present in the smoothing method presented by Li and Ito [34]. In the following, we use the strong form method with a scaled diffusivity and 2 smoothing cycles for comparisons.

Next, we compare the strong form method with the hybrid method. We distinguish between the hybrid method where we switch to the conservative formulation only for those points where diagonal dominance could not be achieved by the strong form method; and the hybrid method where we switch the formulation for the entire neighborhood of such points. For now, let us only consider the first hybrid method; the second method will be introduced in later examples. The numerical solution of the hybrid method can be seen in Fig. 4. Figure 7 illustrates the difference between the strong form and the hybrid method for a small jump of only $\delta\eta = 1 \times 10^1$ (Fig. 7a) and a high jump of $\delta\eta = 1 \times 10^{10}$ (Fig. 7b). In Fig. 7a we can clearly see that the hybrid method does not show any convergence. The method gives almost identical results to



(a) $\delta\eta = 1 \times 10^{-8}$ (b) $\delta\eta = 1 \times 10^{8}$

Fig. 8. Analytical solution u of the curved interface problem with different jumps $\delta\eta$.

the scaled strong form method without smoothing as in Fig. 5b. This shows that for small jumps, the classical formulation already yields diagonally dominant rows, and only a few points that have distorted neighborhoods will use the conservative scheme. In fact, we obtain the same results if we compare both methods on uniform point clouds as seen in Fig. 3a because on uniform point clouds we observed diagonally dominant rows for each point. Increasing the magnitude of the jump, the error of the hybrid method reduces in comparison to the strong form method, see Fig. 7b. This shows that diagonal dominance depends on both, the quality of the point cloud and sufficient regularity of the diffusivity.

3.2. Curved interface

We now consider a test case with a curved interface [9]. The interface of the diffusion coefficient

$$\eta(x,y) = \begin{cases} \eta_L, & y > 2x^3, \\ \eta_R, & y < 2x^3, \end{cases}$$

is given by the curve

$$\Gamma = \{(x,y) \in \Omega \mid y = 2x^3\}.$$

For the inhomogeneous Poisson's equation, we define the source term

$$f(x,y) = -120x^4 + 24x(y-15) - 2$$

and set Dirichlet boundary conditions on the complete boundary such that the analytical solution

$$u(x,y) = \frac{(y-2x^3)^2 - 30(y-2x^3)}{\eta(x,y)}$$

is obtained. Unlike the previous test case, the solution in this test case depends not only on the jump $\delta\eta$ but also on the values η_L and η_R. In order to avoid high gradients for u, we use

$$\eta_R = 10^m$$

for a fixed $\eta_L = 1$ and

$$\eta_L = 10^m$$

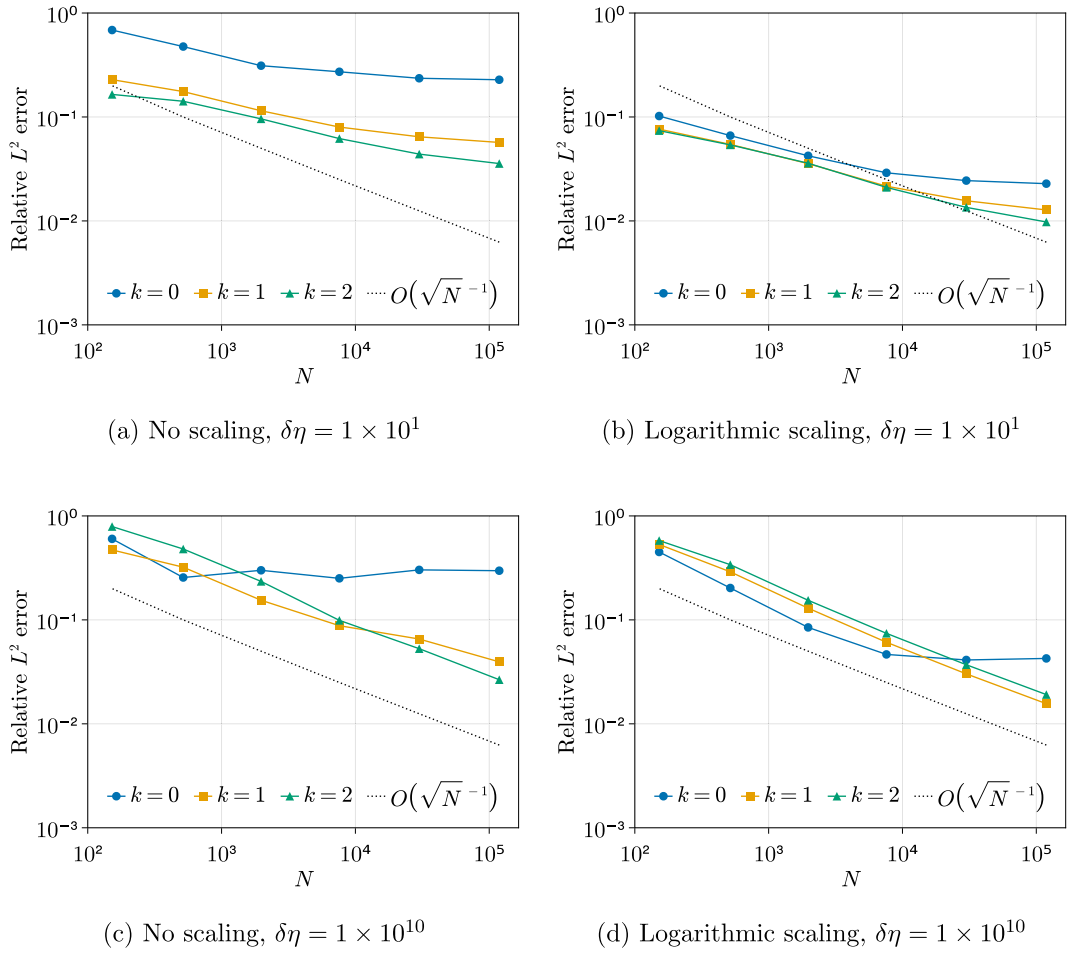
for a fixed $\eta_R = 1$ where $m = 0, \dots, 10$ and define the jump magnitude as in the two-strip test case. Examples for analytical solutions can be viewed in Fig. 8 for two different jump magnitudes. In these figures, we observe a weak discontinuity on the curve Γ.

For the curved interface test case, we qualitatively observe a similar impact of smoothing and scaling in comparison to the two-strip test case, see Figs. 9 and 10. Comparing the errors of both test cases yields that the curved interface test case has higher errors. Similarly, higher errors can be observed in the comparison of the hybrid and the strong form method in Fig. 11. In summary, it can be said that the curvature of the interface has no significant impact on the numerical methods and we draw the same conclusions as for the two-strip test case.

3.3. Interior interface

Unlike the above two test cases, in this test case, we consider a diffusivity that has a discontinuity along a curve that does not intersect the boundary of the domain. For this test case, we define the function

$$\phi(x,y) = \cos(\pi x/2)\cos(\pi y/2)$$



(a) No scaling, $\delta\eta = 1 \times 10^1$

(b) Logarithmic scaling, $\delta\eta = 1 \times 10^1$

(c) No scaling, $\delta\eta = 1 \times 10^{10}$

(d) Logarithmic scaling, $\delta\eta = 1 \times 10^{10}$

Fig. 9. Convergence plots of the classical strong form method for the curved interface test case with and without logarithmic scaling with different jumps $\delta\eta$ and number of smoothing cycles k.

and for the inhomogeneous Poisson's equation, we set the right-hand side

$$f(x,y) = -\Delta\phi(x,y) = \frac{\pi^2}{2}\phi(x,y).$$

Additionally, the Dirichlet boundary condition

$$u|_{\partial\Omega} = 0$$

is set. Given a height $H \in (0,1)$, we define the diffusivity

$$\eta(x,y) = \begin{cases} \eta_{\text{out}}, & \phi(x,y) < H, \\ \eta_{\text{in}}, & \phi(x,y) > H \end{cases}$$

with $\eta_{\text{out}} = 1$ and $\eta_{\text{in}} = 10^k$ for $k = 0, \ldots, 10$. Thus, the function

$$u = \frac{\phi - H}{\eta} + H$$

solves the inhomogeneous Poisson's equation with jump conditions (4). In our case $H = \frac{3}{4}$ is set which leads to a fully interior interface $\Gamma \subset \Omega$ that separates the domain into Ω_{in} and Ω_{out}. This test case has the additional challenge that the interior domain Ω_{in} is entirely contained within Ω and not directly connected to a Dirichlet boundary. This means that the jump conditions have to be propagated across the interface by the discrete diffusion operator. Since in the previous test cases small jumps of only one or two orders of magnitude could not be correctly identified by the diagonal dominance criterion, we focus on larger jumps of several orders of magnitude instead.

We distinguish between three methods. The strong form method is based on the monomial reproducibility enforced with optimization problem (10). The positivity preserving hybrid method is the hybrid approach where we switch to the

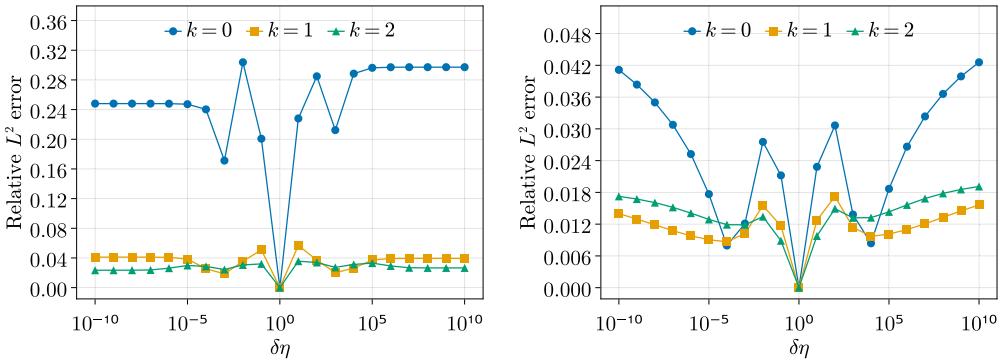


Fig. 10. L^2 errors for the curved interface test case depending on the jump $\delta\eta$.

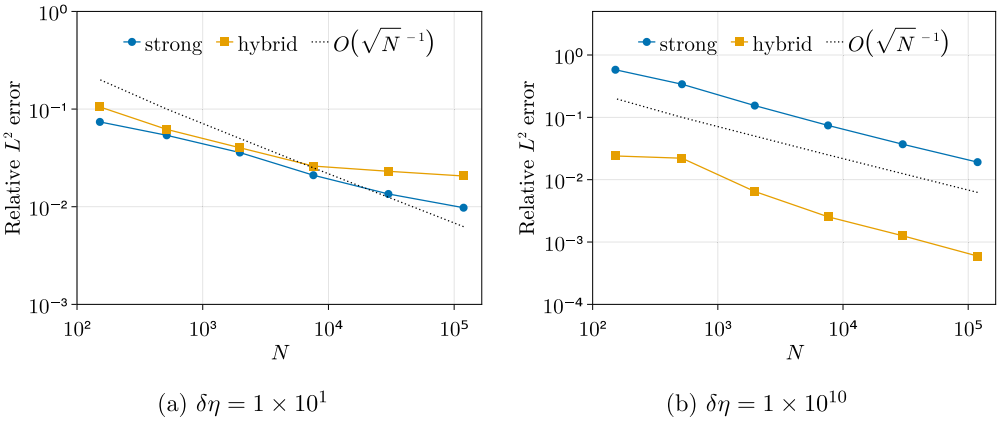


Fig. 11. L^2 errors of strong form method and hybrid method for the curved interface test case with different jump magnitudes.

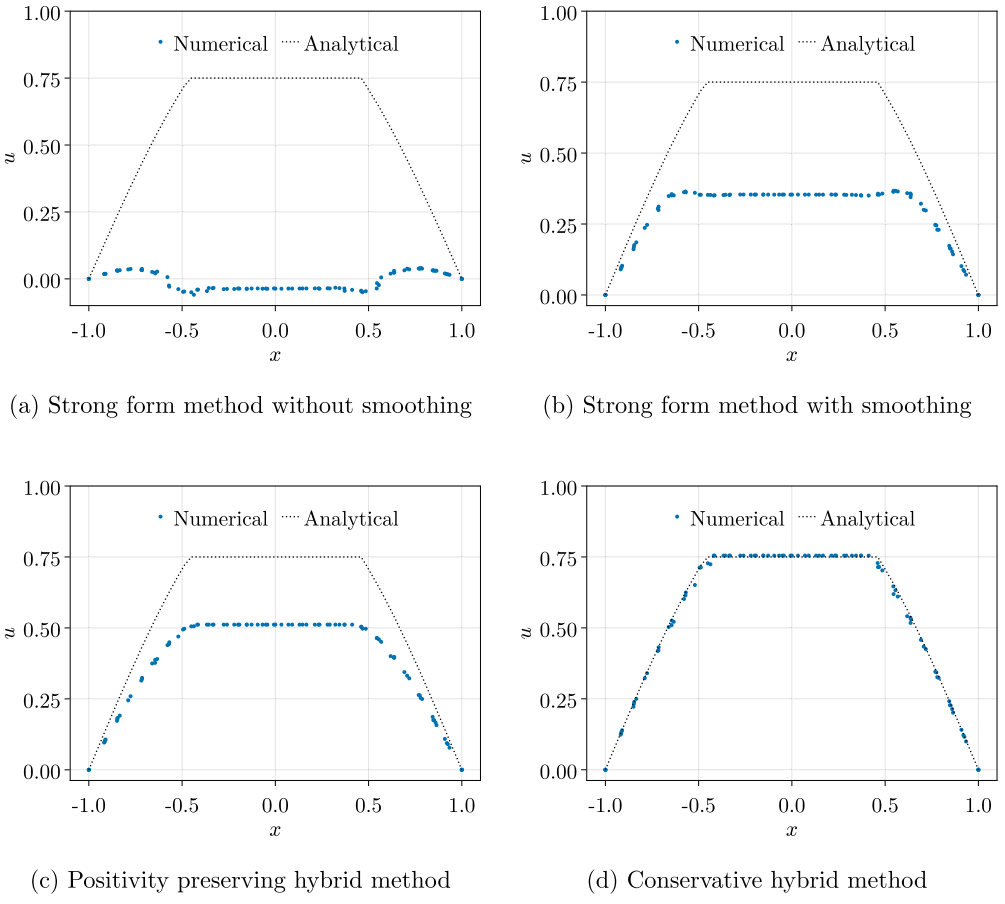
conservative formulation only for points where diagonal dominance could not be achieved by the strong form method. This results in an M-matrix in the linear system (12); hence the method is positivity preserving in the sense of (14). Finally, the conservative hybrid method switches to the conservative formulation in the entire neighborhoods of such points. This method is locally conservative additionally to positivity preserving due to the row sum condition (22).

Note that since $f \geq 0$ on Ω, according to the maximum principle (13) we expect the numerical solution to fulfill $u_i \geq 0$ for each $\boldsymbol{x}_i \in \Omega_h$. Figure 12a shows that the strong form method without smoothing fails to provide positive solutions. For the strong form method with smoothing, we observe the characteristic smeared-out overshoots that could already be seen for the two-strip problem, see Fig. 12b. Switching to the positivity preserving scheme, a positive numerical solution is guaranteed by the method. However, the flux jump condition from Eq. 4 appears to be violated by the numerical solution shown in Fig. 12c. This is caused by points $\boldsymbol{x}_i$ with, for example, $\eta_i = \eta_{\text{in}}$ that have a neighbor $j \in S_i$ across the interface with $\eta_j = \eta_{\text{out}}$ where a diagonally dominant strong form operator could be established and thus flux conservation is not necessarily satisfied. To enforce flux conservation for these points, we need to extend the conservative scheme to these points as well. This results in a good alignment of the numerical and the analytical solution in Fig. 12d. This behavior can be confirmed for different jump magnitudes in Fig. 13 where the positivity preserving method fails to provide first-order convergence. However, the conservative method has first-order convergence and this shows that the strong form scheme produces diagonally dominant discrete operators for some points that have their neighborhoods across the interface, as described earlier. Additionally, the conservative method is insensitive to the jump magnitude as shown in Fig. 14. In summary, the conservative scheme shows the best results for high jumps and thus should be the preferred method.

3.4. Three-strip problem

Similarly to the two-strip test case, we define the diffusivity with three positive values η_L, η_M and η_R

$$\eta(x, y) = \begin{cases} \eta_L, & x < -\frac{1}{3}, \\ \eta_M, & -\frac{1}{3} < x < \frac{1}{3}, \\ \eta_R, & x > \frac{1}{3} \end{cases}$$



(a) Strong form method without smoothing

(b) Strong form method with smoothing

(c) Positivity preserving hybrid method

(d) Conservative hybrid method

Fig. 12. Solution profile at $y \approx 0$ to the interior interface test case with a jump of $\delta\eta = 1 \times 10^6$.

that leads to three equally sized strips

$$\Omega_L = \left\{(x, y) \in \Omega \mid x < -\frac{1}{3}\right\},$$
$$\Omega_M = \left\{(x, y) \in \Omega \mid -\frac{1}{3} < x < \frac{1}{3}\right\},$$
$$\Omega_R = \left\{(x, y) \in \Omega \mid x > -\frac{1}{3}\right\}.$$

Dirichlet and Neumann boundary conditions are set exactly as for the two-strip test case such that the analytical solution is again monotonously decreasing and piecewise linear in x-direction. A three-strip problem can be characterized by the two jumps at the interfaces

$$\delta\eta_L = \frac{\eta_M}{\eta_L},$$
$$\delta\eta_R = \frac{\eta_R}{\eta_M}$$

and with $\eta_L = 1$ we have

$$\eta_M = \delta\eta_L,$$
$$\eta_R = \delta\eta_L\delta\eta_R.$$

In this test case, we added additional complexity to the problem by partitioning the domain into the three strips where the interior strip Ω_M is not directly connected to a Dirichlet boundary. The only boundary condition that appears for this subdomain is a Neumann boundary condition which introduces additional challenges to the numerical scheme.

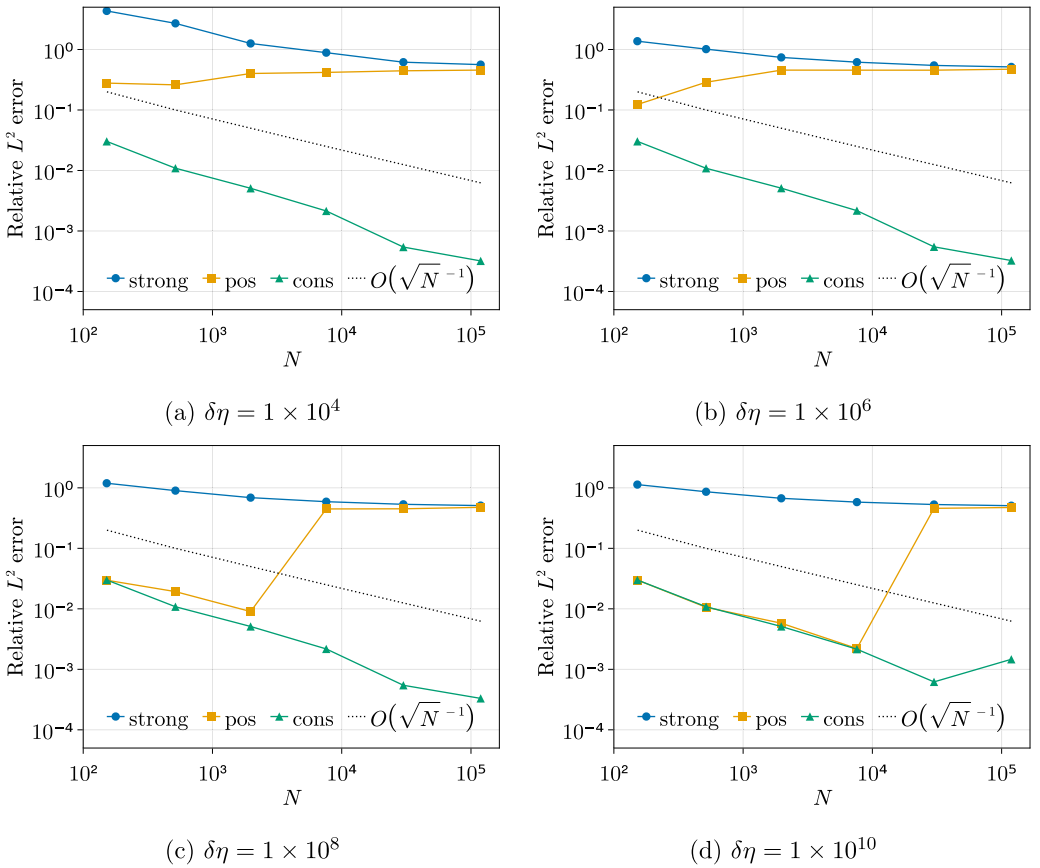


Fig. 13. Convergence plots of the classical strong form method, the positivity-preserving method (pos) and the conservative method (cons) for the interior interface test case with different jumps $\delta\eta$.

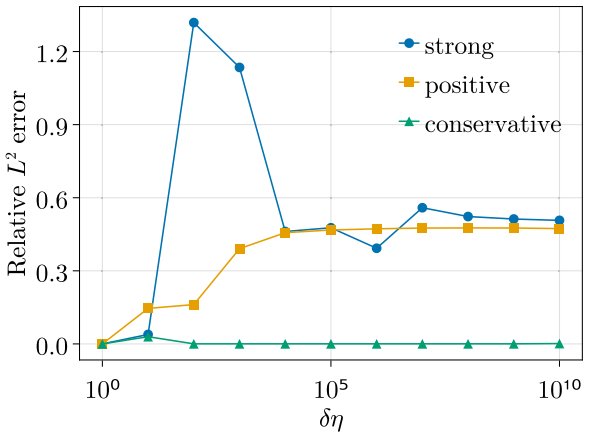


Fig. 14. L^2 errors for the interior interface test case depending on the jump $\delta\eta$.

1. We need the numerical scheme to propagate the jump conditions (4) from the surrounding domains Ω_L and Ω_R correctly.
2. Boundary points with a Neumann boundary condition that lie in the proximity of the interface have to propagate the correct jump conditions across the interface boundary.

In the previous experiment with the interior interface test case, we observed that the jump propagation across interfaces works best with the conservative method. However, problems arise from the Neumann boundary where false jump conditions are propagated. The goal of this test case is to examine if the conservative Neumann boundary conditions can be used to improve numerical results. We use conservative Neumann boundary conditions in the neighborhood of points where a

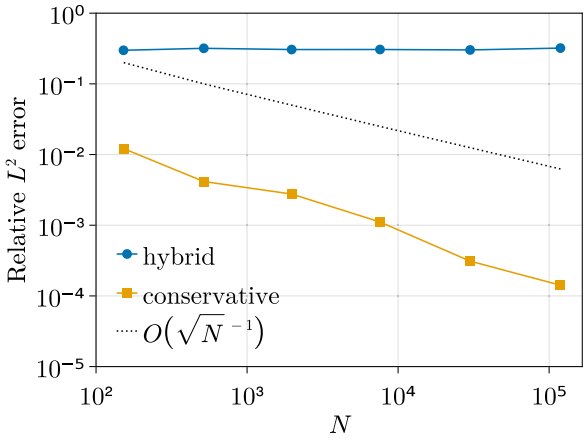
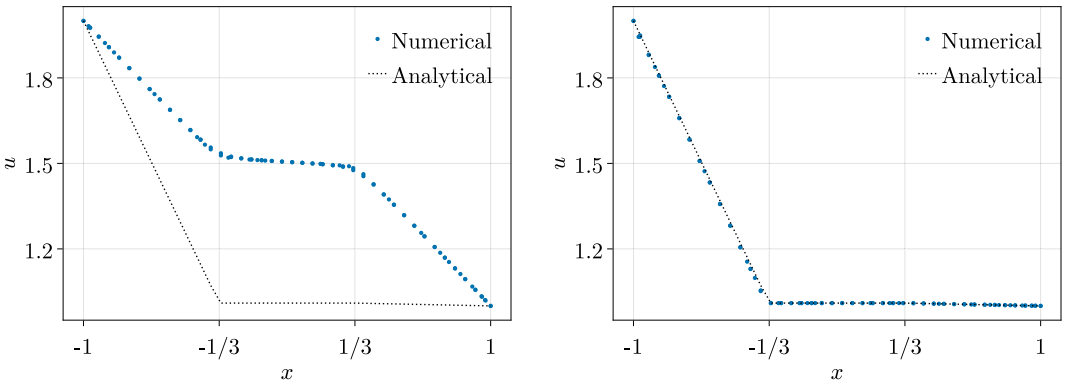


Fig. 15. L^2 errors for the three-strip test case of hybrid method with and without conservative Neumann boundary conditions for a jump $\delta\eta_L = 1 \times 10^6$ and $\delta\eta_R = 1 \times 10^{-4}$.



(a) Regular Neumann boundaries (b) Conservative Neumann boundaries

Fig. 16. Solution profiles comparison at $y \approx 0$ for the three-strip test case of hybrid method with and without conservative Neumann boundary conditions for a jump $\delta\eta_L = 1 \times 10^6$ and $\delta\eta_R = 1 \times 10^{-4}$.

switch to the conservative formulation is performed. We observe that the standard hybrid method does not converge for this test case as can be seen in Fig. 15. A view on Fig. 16a reveals that the analytical solution is not reached due to the lack of flux conservation [20]. For a numerical solution u_h let us define the error in flux by

$$\delta q = \eta \partial_x (u - u_h).$$

According to Eq. (4), the flux error δq has to be piecewise constant and continuous which leads to it being constant throughout the domain. In Fig. 17a we see that the flux error δq for the numerical solution without conservative Neumann boundary conditions is not constant. This observation can be made especially at the boundaries $\partial\Omega_T$ and $\partial\Omega_B$ which leads to the assumption that flux conservation has to be met at the boundary as well. Finally, switching to conservative Neumann boundary conditions, we see in Figs. 16b and 17b that the analytical solution matches the numerical solution visually and δq has lower oscillations. Figure 15 confirms that the method is first-order accurate with conservative Neumann boundary conditions.

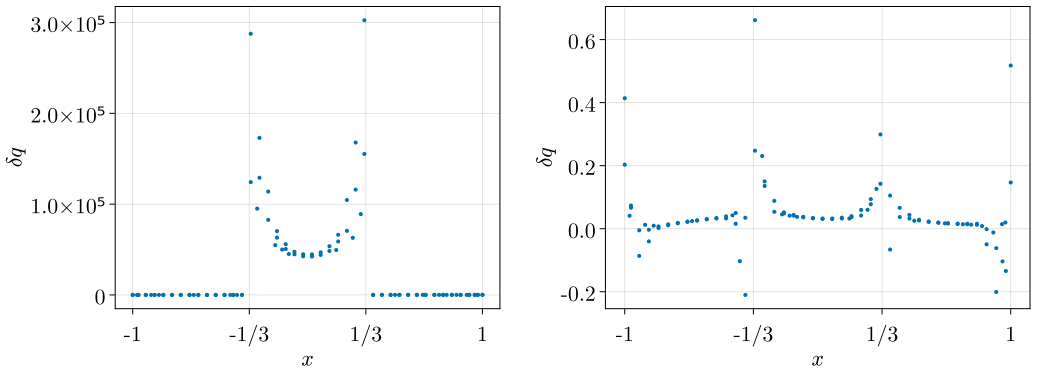
To better illustrate δq in Fig. 17, a domain decomposition based on η was executed to be able to calculate the gradient $\nabla u|_{\Omega_*}$ on each subdomain Ω_L, Ω_M and Ω_R without the influence of the neighboring subdomains. The numerical results were computed without a domain decomposition. Note that the outliers on Fig. 17b are due to the irregularity of the point cloud and the circumstance that we calculate the gradient near the interface which serves as a boundary to the subdomains. Computing δq on a uniform point cloud, we obtain Fig. 18 and thus the flux error at each point is given by

$$\delta q_i = c + \varepsilon_i$$

for a numerically computed constant $c \approx -1.834 \times 10^{-02}$ and an error

$$\max_{i=1,\dots,N} |\varepsilon_i| = O(1 \times 10^{-8})$$

which is negligible.



(a) Regular Neumann boundaries (b) Conservative Neumann boundaries

Fig. 17. Profiles of $\delta q = \eta \partial_x (u - u_h)$ at $y \approx 0$ for the three-strip test case of hybrid method with and without conservative Neumann boundary conditions for a jump $\delta\eta_L = 1 \times 10^6$ and $\delta\eta_R = 1 \times 10^{-4}$ on an irregular point cloud.

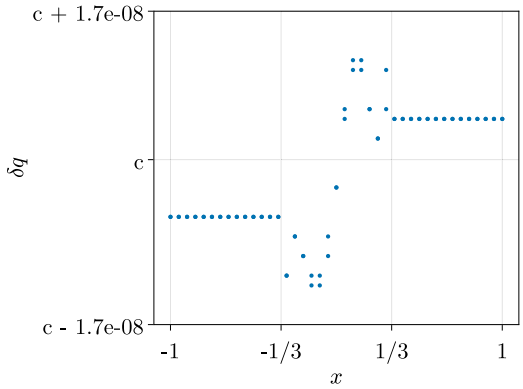


Fig. 18. Profile of $\delta q = \eta \partial_x (u - u_h)$ at $y \approx 0$ for the three-strip test case of hybrid method with conservative Neumann boundary conditions for a jump $\delta\eta_L = 1 \times 10^6$ and $\delta\eta_R = 1 \times 10^{-4}$ on a uniform point cloud. The constant $c \approx -1.834 \times 10^{-02}$ was numerically determined.

4. Discussion

In this section, we give a summary of the results from the previous section and discuss the advantages and disadvantages of the new hybrid method.

Our first observation was that artificial smoothing could stabilize the convergence behavior of the strong form method for very simple test cases, such as the two-strip problem. However, we have also seen other test cases where the strong form method failed to provide accurate solutions, regardless of whether we smooth or not. Thus, the first consequence is that artificial smoothing should be avoided because it distorts the data from the mathematical model and does not yield advantages for more complex test cases.

The hybrid method produces good results and provides first-order convergence for elliptic interface problems. In the performance benchmark in Appendix B, we see that the computation time does not increase compared to the fully strong form method, due to a lower number of iterations in the linear solver. Additionally, our method guarantees the M-matrix property and thus positivity of numerical results and fulfills the discrete divergence theorem locally. However, one limitation is the need for computing a Voronoi cell. We observed that the presented node selection routine only works well with high discontinuities in the diffusion coefficient, as seen in Sections 3.1 and 3.2. It is a major drawback that the node selection depends on the jump magnitude and not the mere existence of a jump in the diffusion coefficient. On top of that, it is also susceptible to low point cloud quality, see Appendix A, but this issue can be managed by better point cloud management routines.

Finally, we saw that placing conservative Neumann boundary conditions is necessary for the three-strip test case.

5. Conclusion

In this paper, we presented two formulations of discretizing the diffusion operator and a new way of combining these approaches in a locally conservative and positivity preserving method. Our approach combines the cell average-based

conservative formulation and the strong form weighted least squares-based formulation into a new method. It puts the finite volume method idea inside the context of GFDM and uses its advantages without sacrificing the advantages that GFDM offers, namely the flexibility of the point cloud. This is a major benefit with respect to time-dependent problems in a Lagrangian formulation with a moving point cloud, which will be an extension of the present work.

We performed an in-depth parameter study with different test cases and methods. We considered jumps up to ten orders of magnitude whereas existing methods in literature were limited to three orders of magnitude. Our numerical results show that the classical weighted least squares approach fails for high jump magnitudes and that using a flux-conserving scheme overcomes these problems. Finally, we have seen that it can be necessary to incorporate conservative Neumann boundary conditions to guarantee the conservation property of the entire numerical scheme. Since we carried out the computations on non-conforming point clouds, we showed that the presented method does not depend on aligning the point cloud to the interface.

The discretization of the diffusion operator is subject to ongoing research and as such open questions remain. For the diffusion operator, it is possible to use the notion of shapeless volumes and higher-order GFDM methods. Additionally, the jump identification technique can be extended for it to be able to identify lower jumps. While this is not needed for our applications, it might be necessary for others.

Acknowledgments

Pratik Suchde would like to acknowledge support from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie Actions grant agreement No. 892761 "SURFING".

Appendix A. Selection of nodes for conservative method

In this section, we discuss the selection of nodes for the conservative method. For that, we define the index set I of interior points. We denote the subset of points where the diagonal dominance criterion (23) triggers by

$$I_{\sigma,0} = \{i \in I \mid \sigma_i > 0\}$$

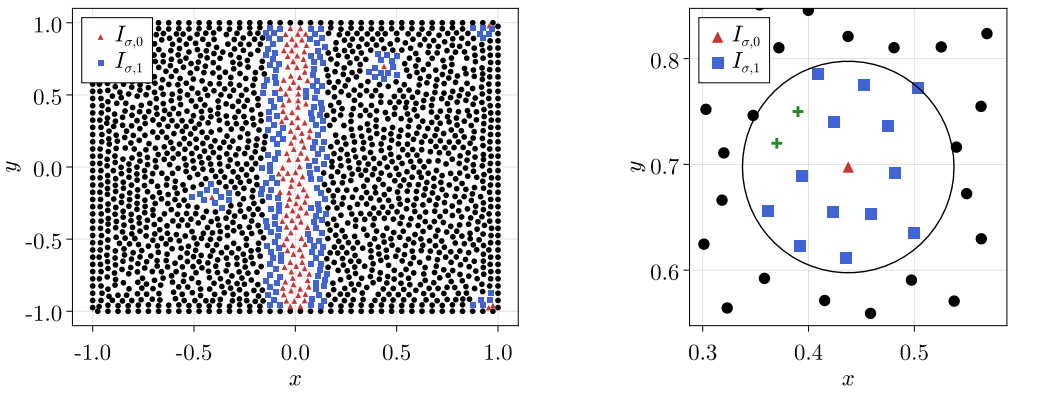
and the index sets of their neighbors by

$$I_{\sigma,1} = \{i \in I \setminus I_{\sigma,0} \mid S_i \cap I_{\sigma,0} \neq \emptyset\}.$$

Note that for $I_{\sigma,0}$, the criterion $\sigma_i > 0$ is unpractical since numerical errors are not factored in and thus the majority of points are selected for the conservative scheme. To overcome this drawback, we provide a tolerance $\epsilon > 0$, in our case $\epsilon = 1 \times 10^{-12}$, and use the criterion $\sigma_i > \epsilon$ instead. We tested other values $\epsilon \in (1 \times 10^{-12}, 1 \times 10^{-4})$ but a remarkable difference in the selected nodes could not be observed. In Fig. A.1a the index sets are depicted for a jump of $\delta\eta = 1 \times 10^4$ and we can see that $I_{\sigma,0}$ generally conforms to the interface

$$\Gamma = \{(x, y)^T \in \mathbb{R}^2 \mid x = 0\}$$

well. But we also see outliers that are in the region where the diffusivity η is smooth, one of which is shown in Fig. A.1b. These outliers are most likely due to the irregularity or locally low quality of the point cloud. The addition of two points at



(a) Global selection (b) Local selection due to irregularity

Fig. A.1. Nodes selected for the hybrid regime by the diagonal dominance criterion for an irregular point cloud with $h = 5 \times 10^{-2}$. Nodes whose addition lead to a diagonally dominant operator are marked with a green plus sign. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table A.1
Node fractions in % for hybrid method with different smoothing lengths h and jump magnitudes $\delta\eta$.

$\delta\eta$	$h = 5 \times 10^{-2}$				$h = 1.25 \times 10^{-2}$			
	$\frac{\lvert I_{\sigma,0}\rvert}{\lvert I\rvert}$	$\frac{\lvert I_c\rvert}{\lvert I\rvert}$	$\frac{\lvert I_c \cap I_\Gamma\rvert}{\lvert I_c\rvert}$	$\frac{\lvert I_c \cap I_\Gamma^+\rvert}{\lvert I_c\rvert}$	$\frac{\lvert I_{\sigma,0}\rvert}{\lvert I\rvert}$	$\frac{\lvert I_c\rvert}{\lvert I\rvert}$	$\frac{\lvert I_c \cap I_\Gamma\rvert}{\lvert I_c\rvert}$	$\frac{\lvert I_c \cap I_\Gamma^+\rvert}{\lvert I_c\rvert}$
$1 \times 10^{+00}$	0.28	2.21	0.00	0.00	0.13	1.57	0.00	0.00
$1 \times 10^{+02}$	5.13	15.72	51.58	85.96	1.35	4.96	41.27	69.19
$1 \times 10^{+04}$	7.00	17.43	46.52	87.34	1.87	5.43	37.64	71.89
$1 \times 10^{+06}$	7.78	18.15	44.68	87.84	2.04	5.59	36.56	72.70
$1 \times 10^{+08}$	8.11	18.53	43.75	88.10	2.12	5.67	36.08	73.06
$1 \times 10^{+10}$	8.33	18.70	43.36	88.20	2.16	5.69	35.93	73.17

the locations $(0.37, 0.72)$ and $(0.39, 0.75)$ in the top-left empty region of the neighborhood, marked as a plus in Fig. A.1b, leads to a diagonally dominant row.

In the following, we will discuss the fraction of nodes that fall under the hybrid regime for the two-strip test case with varying jump magnitude and smoothing length. For this, we define the index set of nodes that are chosen for the conservative scheme by

$$I_c = I_{\sigma,0} \cup I_{\sigma,1}$$

To define the set of nodes that are close to the interface, we exploit the fact that η is piecewise constant and define the proximity to the interface locally through the neighborhoods

$$I_\Gamma = \{i \in I \mid \exists j \in S_i : \eta_i \neq \eta_j\}.$$

Finally, we define the set of nodes that consist of all the neighborhoods of points that are close to the interface

$$I_\Gamma^+ = \bigcup_{i \in I_\Gamma} S_i \cap I.$$

We summarized different node fractions in Table A.1. First, let us take a look at the fraction of nodes that are selected for the hybrid regime by the $\sigma_i > 0$ criterion, given by $\frac{\lvert I_{\sigma,0}\rvert}{\lvert I\rvert}$. In the table, we observe that even without a jump, some points are selected for the hybrid regime. By increasing the jump, we observe that the number of points that do not fulfill the diagonal dominance criterion rises but appears to be bounded for both smoothing lengths. As expected, the fraction of such nodes decreases with a decreasing smoothing length h and thus increasing number of points. The fraction of points chosen for the conservative scheme, indicated by $\frac{\lvert I_c\rvert}{\lvert I\rvert}$, displays the same behavior.

In the next step, we take a look at the fraction of nodes that are selected for the hybrid regime and those that are actually within the region of the jump, given by $\frac{\lvert I_c \cap I_\Gamma\rvert}{\lvert I_c\rvert}$. We observe that for an increasing jump, the number of points that use the hybrid method that are within the interface region decreases, meaning that for an increasing jump, we select points that are not within the interface region. This can be explained by points $\boldsymbol{x}_i$ that have one or two points within their neighborhoods $\boldsymbol{x}_j \in B_i$ that have the other η-value, meaning $\eta_i \neq \eta_j$. For low jumps, it is possible for such points to obtain a diagonally dominant discrete operator; with an increasing jump however, these few neighbors have an increasing influence on the numerical gradient which then leads to instabilities in the discrete operator. This assumption can be confirmed by taking a look at the ratio $\frac{\lvert I_c \cap I_\Gamma^+\rvert}{\lvert I_c\rvert}$ that indicates the fraction of points selected for the hybrid regime that also lie in the extended proximity of the interface I_Γ^+. This ratio is increasing with an increasing jump magnitude.

Appendix B. Performance benchmarks

It is reasonable to assume that the hybrid method presented in Section 2.2.3 would decrease the performance of the numerical solver as compared to using the strong form method from Section 2.2.1 exclusively. Not only do we have to compute a Voronoi cell for the selected points, but we are required to compute the conservative operator in addition to the strong form one. In our self-written Julia code, a global Delaunay triangulation of the point cloud is computed. Since the Voronoi tessellation is the dual of the Delaunay triangulation, we can easily reconstruct the Voronoi cells from the triangulation by connecting the circumcenters of each triangle. For large-scale 3D applications, it is generally not feasible to compute a global Delaunay triangulation. In the software suite MESHFREE [33], a Delaunay triangulation is computed locally for point cloud management purposes. Computing Voronoi cells from an underlying (local) Delaunay triangulation is thus not that expensive compared to constructing Voronoi cells directly.

We measured the runtime of the self-written numerical solver in Julia [35] on an Intel® Core™ i5-8600K processor from the start until the end of the simulation. We used the interior interface test case from Section 3.3 with a jump $\delta\eta = 1 \times 10^{10}$. The global Delaunay triangulation was only computed for the hybrid method using a Julia wrapper of Triangle [36]. The global linear system Eq. (12) was solved using BiCGstab(2) [37] and an incomplete LU preconditioning. For each smoothing length, we computed the mean elapsed time of a sample of ten simulations. The results are summarized in Table B.1.

Table B.1
Number of BiCGstab iterations N_{iter} and mean elapsed time for the strong form and hybrid methods with different smoothing lengths h.

h	Strong Form		Hybrid	
	N_{iter}	Mean elapsed time	N_{iter}	Mean elapsed time
2×10^{-1}	1	28.2 ms	2	28.4 ms
1×10^{-1}	2	92.7 ms	3	88.7 ms
5×10^{-2}	3	136 ms	2	119 ms
2.5×10^{-2}	4	376 ms	3	355 ms
1.25×10^{-2}	9	1.76 s	6	1.62 s
6.25×10^{-2}	18	7.27 s	11	6.98 s
3.125×10^{-2}	33	36.21 s	21	33.93 s

We can see that there is no performance drawback of the new method, the hybrid method is even faster in most cases. This can be explained due to the Delaunay triangulation being not very complex in 2D on a square domain. On the finest point cloud, the computation of the Delaunay triangulation took only 1 s which is roughly 3% of the total computation time. However, the hybrid method needed fewer BiCGstab iterations and thus can compensate for the lost time due to the Delaunay triangulation and additional computation of the conservative operators.

References

[1] T.J. Liszka, J. Orkisz, The finite difference method at arbitrary irregular grids and its application in applied mechanics, Comput. Struct. 11 (1–2) (1980) 83–95, doi:10.1016/0045-7949(80)90149-2.
[2] L. Gavete, F. Ureña Prieto, J.J. Benito, Á. García, M. Ureña, E. Salete, Solving second order non-linear elliptic partial differential equations using generalized finite difference method, J. Comput. Appl. Math. 318 (2017) 378–387, doi:10.1016/j.cam.2016.07.025.
[3] S. Milewski, Combination of the meshless finite difference approach with the Monte Carlo random walk technique for solution of elliptic problems, Comput. Math. Appl. 76 (4) (2018) 854–876, doi:10.1016/j.camwa.2018.05.025.
[4] W. Qu, Y. Gu, Y. Zhang, C.-M. Fan, C. Zhang, A combined scheme of generalized finite difference method and Krylov deferred correction technique for highly accurate solution of transient heat conduction problems, Int. J. Numer. Methods Eng. 117 (1) (2019) 63–83, doi:10.1002/nme.5948.
[5] P.-W. Li, C.-M. Fan, Generalized finite difference method for two-dimensional shallow water equations, Eng. Anal. Bound. Elem. 80 (2017) 58–71, doi:10.1016/j.enganabound.2017.03.012.
[6] J. Kuhnert, Meshfree numerical scheme for time dependent problems in fluid and continuum mechanics, in: Advances in PDE Modeling and Computation, 2014, pp. 119–136.
[7] J. Kuhnert, An upwind finite pointset method (FPM) for compressible Euler and Navier–Stokes equations, in: Lecture Notes in Computational Science and Engineering, Springer Berlin Heidelberg, 2003, pp. 239–249.
[8] P. Suchde, H. Kraus, B. Bock-Marbach, J. Kuhnert, Meshfree one-fluid modelling of liquid-vapor phase transitions, arXiv preprint arXiv:2203.10383(2022).
[9] O. Davydov, M. Safarpoor, A meshless finite difference method for elliptic interface problems based on pivoted QR decomposition, Appl. Numer. Math. 161 (2021) 489–509, doi:10.1016/j.apnum.2020.11.018.
[10] M. Shao, L. Song, P.-W. Li, A generalized finite difference method for solving Stokes interface problems, Eng. Anal. Bound. Elem. 132 (2021) 50–64, doi:10.1016/j.enganabound.2021.07.002.
[11] F. Gibou, L. Chen, D. Nguyen, S. Banerjee, A level set based sharp interface method for the multiphase incompressible Navier–Stokes equations with phase change, J. Comput. Phys. 222 (2) (2007) 536–555, doi:10.1016/j.jcp.2006.07.035.
[12] R. Edgeworth, B.J. Dalton, T. Parnell, The pitch drop experiment, Eur. J. Phys. 5 (4) (1984) 198–200, doi:10.1088/0143-0807/5/4/003.
[13] Y.-C. Yoon, J.-H. Song, Extended particle difference method for weak and strong discontinuity problems: Part I. Derivation of the extended particle derivative approximation for the representation of weak and strong discontinuities, Comput. Mech. 53 (6) (2014) 1087–1103, doi:10.1007/s00466-013-0950-8.
[14] P. Suchde, J. Kuhnert, A meshfree generalized finite difference method for surface PDEs, Comput. Math. Appl. 78 (8) (2019) 2789–2805, doi:10.1016/j.camwa.2019.04.030.
[15] T.-P. Fries, T. Belytschko, The extended/generalized finite element method: an overview of the method and its applications, Int. J. Numer. Methods Eng. 84 (3) (2010) 253–304, doi:10.1002/nme.2914.
[16] S.P.A. Bordas, S. Natarajan, P. Kerfriden, C.E. Augarde, D.R. Mahapatra, T. Rabczuk, S.D. Pont, On the performance of strain smoothing for quadratic and enriched finite element approximations (XFEM/GFEM/PUFEM), Int. J. Numer. Methods Eng. 86 (4–5) (2011) 637–666, doi:10.1002/nme.3156.
[17] E.O. Reséndiz-Flores, F.R. Saucedo-Zendejo, Numerical simulation of coupled fluid flow and heat transfer with phase change using the finite pointset method, Int. J. Therm. Sci. 133 (2018) 13–21, doi:10.1016/j.ijthermalsci.2018.07.008.
[18] F.R. Saucedo-Zendejo, E.O. Reséndiz-Flores, Transient heat transfer and solidification modelling in direct-chill casting using a generalized finite differences method, J. Mining and Metall. Sect. BMetall. 55 (1) (2019) 47–54, doi:10.2298/jmmb180214005s.
[19] N. Trask, M. Perego, P. Bochev, A high-order staggered meshless method for elliptic problems, SIAM J. Sci. Comput. 39 (2) (2017) A479–A502, doi:10.1137/16m1055992.
[20] P. Suchde, J. Kuhnert, S. Schröder, A. Klar, A flux conserving meshfree method for conservation laws, Int. J. Numer. Methods Eng. 112 (3) (2017) 238–256, doi:10.1002/nme.5511.
[21] E. Kwan-yu Chiu, Q. Wang, R. Hu, A. Jameson, A conservative mesh-free scheme and generalized framework for conservation laws, SIAM J. Sci. Comput. 34 (6) (2012) A2896–A2916, doi:10.1137/110842740.
[22] M. Chipot, Elliptic Equations: An Introductory Course, Springer, 2009.
[23] T. Jacquemin, S. Tomar, K. Agathos, S. Mohseni-Mofidi, S.P.A. Bordas, Taylor-series expansion based numerical methods: a primer, performance benchmarking and new approaches for problems with non-smooth solutions, Arch. Comput. Methods Eng. 27 (5) (2020) 1465–1513, doi:10.1007/s11831-019-09357-5.
[24] Z. Li, K. Ito, Maximum principle preserving schemes for interface problems with discontinuous coefficients, SIAM J. Sci. Comput. 23 (1) (2001) 339–361, doi:10.1137/s1064827500370160.
[25] P. Suchde, Conservation and accuracy in meshfree generalized finite difference methods, Technische Universität Kaiserslautern, 2018 Ph.D. thesis.
[26] B. Seibold, M-matrices in meshless finite difference methods, Technische Universität Kaiserslautern, 2006 Ph.D. thesis.
[27] R. Eymard, T. Gallouët, R. Herbin, Finite volume methods, in: Solution of Equation in $\mathbb{R}^n$ (Part 3), Techniques of Scientific Computing (Part 3), in: Handbook of Numerical Analysis, vol. 7, Elsevier, 2000, pp. 713–1018, doi:10.1016/S1570-8659(00)07005-8.

[28] T. Seifarth, Numerische Algorithmen für gitterfreie Methoden zur Lösung von Transportproblemen, Universität Kassel, 2018 Ph.D. thesis.
[29] P.M. Knupp, Algebraic mesh quality metrics, SIAM J. Sci. Comput. 23 (1) (2001) 193–218, doi:10.1137/s1064827500371499.
[30] G. Allasia, R. Besenghi, R. Cavoretto, Adaptive detection and approximation of unknown surface discontinuities from scattered data, Simul. Model. Pract. Theory 17 (6) (2009) 1059–1070, doi:10.1016/j.simpat.2009.03.007.
[31] T. Gutzmer, A. Iske, Detection of discontinuities in scattered data approximation, Numer. Algorithms 16 (2) (1997) 155–170, doi:10.1023/a:1019139130423.
[32] R. Löhner, E. Oñate, An advancing front point generation technique, Commun. Numer. Methods Eng. 14 (12) (1998) 1097–1108, doi:10.1002/(sici)1099-0887(199812)14:12<1097::aid-cnm183)3.0.co;2-7.
[33] MESHFREE Homepage, 2022, Accessed 27-06-2022 (https://www.meshfree.eu).
[34] Z. Li, K. Ito, The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains, Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics, 2006.
[35] J. Bezanson, A. Edelman, S. Karpinski, V.B. Shah, Julia: a fresh approach to numerical computing, SIAM Rev. 59 (1) (2017) 65–98, doi:10.1137/141000671.
[36] J.R. Shewchuk, Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator, Springer Berlin Heidelberg, Berlin, Heidelberg, 1996, pp. 203–222.
[37] G.L. Sleijpen, D.R. Fokkema, BiCGstab(l) for linear equations involving unsymmetric matrices with complex spectrum, Electron. Trans. Numer. Anal. 1 (1993) 11–32.