Journal of Computational Science 79 (2024) 102306

Contents lists available at ScienceDirect

Journal of Computational Science

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

Spatially dependent node regularity in meshless approximation of partial differential equations

Miha Rot [a,b,*], Mitja Jančič [a], Gregor Kosec [a]

[a] Institute Jožef Stefan, Parallel and Distributed Systems Laboratory, Jamova cesta 39, 1000 Ljubljana, Slovenia
[b] International Postgraduate School Jožef Stefan, Jamova Cesta 39, 1000 Ljubljana, Slovenia

ARTICLE INFO

Keywords:
LRBFCM
RBF-FD
RBF
Meshless
Node regularity
Navier–Stokes equation
Natural convection
Navier-Cauchy equation
Boussinesq's problem

ABSTRACT

In this paper, we address a way to reduce the total computational cost of meshless approximation by reducing the required stencil size through spatially varying computational node regularity. Rather than covering the entire domain with scattered nodes, only regions with geometric details are covered with scattered nodes, while the rest of the domain is discretized with regular nodes. A simpler approximation can be used in regions covered by regular nodes, effectively reducing the required stencil size and computational cost compared to the approximation on scattered nodes where a set of polyharmonic splines is added to ensure convergent behaviour.

This paper is an extended version of conference paper entitled "Spatially-varying meshless approximation method for enhanced computational efficiency" (Jančič et al., 2023) presented at "International Conference on Computational Science (ICCS) 2023". The paper is extended with discussion on development and implementation of a hybrid regular-scattered node positioning algorithm (HyNP). The performance of the proposed HyNP algorithm is analysed in terms of separation distance and maximal empty sphere radius. Furthermore, it is demonstrated that HyNP nodes can be used for solving problems from fluid flow and linear elasticity, both in 2D and 3D, using meshless methods.

The extension also provides additional analyses of computational efficiency and accuracy of the numerical solution obtained on the spatially-variable regularity of discretization nodes. In particular, different levels of refinement aggressiveness and scattered layer widths are considered to exploit the computational efficiency gains offered by such solution procedure.

1. Introduction

Although the meshless methods are formulated without any restrictions regarding the node layouts, it is generally accepted that quasi-uniformly-spaced node sets improve the stability of meshless methods [1,2]. Nevertheless, even with quasi-uniform nodes generated with recently proposed node positioning algorithms [3–5], a sufficiently large stencil size is required for stable approximation. A stencil with $n = 2\binom{m+d}{m}$ nodes is recommended [6] for the local Radial Basis Function-generated Finite differences (RBF-FD) [7] method in a d-dimensional domain for approximation order m. The performance of RBF-FD method – with approximation basis consisting of Polyharmonic splines (PHS) and monomial augmentation with up to and including monomials of degree m – has been demonstrated with scattered nodes on several applications [8–10]. On the other hand, approximation on regular nodes can be performed with considerably smaller stencil ($n = 5$ in two-dimensional domain) using only monomial basis [11] or only Radial Basis Function (RBF) [12].

A way to reduce the overall computational complexity while maintaining accuracy is therefore to divide the domain into regions where we need scattered nodes to conform the irregular geometry and the rest of the domain that can be covered with regular nodes that enable using approximation with smaller stencils. While it is not necessary to use regular nodes at all, the more of the domain we can discretize with them without compromising the description of the geometry, the better the expected computational performance.

The spatially varying approximation method has already been introduced in the past to address different problems with different combinations of methods. A hybrid Finite element method (FEM)-meshless method [13] has been proposed to overcome the issues regarding the unstable Neumann boundary conditions in the context of meshless approximation. FEM has been also coupled with meshless method in [14] using approximation constraints to solve Poisson's problem, elasticity and thermo-elasticity problems. Moreover, the authors of [15,

* Corresponding author at: Institute Jožef Stefan, Parallel and Distributed Systems Laboratory, Jamova cesta 39, 1000 Ljubljana, Slovenia.
E-mail addresses: miha.rot@ijs.si (M. Rot), mitja.jancic@ijs.si (M. Jančič), gregor.kosec@ijs.si (G. Kosec).

https://doi.org/10.1016/j.jocs.2024.102306
Received 19 October 2023; Received in revised form 12 March 2024; Accepted 17 April 2024
Available online 30 April 2024

Table 1
Overview of listed hybrid methods.

Methods	Problems tackled	Ref
Meshless/FEM	Nonlinear structural problems	[13]
Meshless/FEM	Poisson's and thermo-elasticity problems	[14]
Meshless/FDM	Flows past a circular cylinder	[15]
Meshless/FDM	Elasticity problems	[16]
Meshless/FDM	Geodynamical simulations	[17]
Meshless/FDM	Flow around solid bodies	[18]

16] proposed a hybrid of Finite Difference Method (FDM) employed on conventional cartesian grid combined with meshless approximation on scattered nodes to solve flows past a circular cylinder and elasticity problems, respectively. FDM has been also coupled with meshless in the context of geodynamical simulations [17], where authors experimented with combination of Eulerian-Lagrangian schemes. These hybrid approaches are well elaborated, provide stable numerical results and are computationally effective, nevertheless, additional implementation-related burden is required on the transition from mesh based discretization to scattered nodes [18], contrary to the objective of this paper relying solely on the framework of meshless methods, including the generation of hybrid regular-scattered nodes. The overview of discussed hybrid methods is presented in Table 1.

In this paper we first propose a dedicated hybrid regular-scattered node positioning algorithm (HyNP) that is capable of handling irregularities in the domain with scattered nodes, while covering the rest of the domain with regular nodes. The algorithm is dimension independent, i.e. the same algorithm can be used to populate n-dimensional domains and inherently supports h-refinement through the spatially dependent node density. The performance of the proposed algorithm is evaluated in terms of two metrics that are commonly used to evaluate the discretization quality, i.e. the distance to the closest neighbours also referred to as separation distance and the largest empty circle radius. Afterwards, the solution procedure based on such spatially-variable node regularity is analysed in terms of computational efficiency and accuracy of the numerical solution. Compared to the original work [19], the analyses in this paper are extended to include different levels of h-refinement aggressiveness and scattered layer widths to further exploit the efficiency gains offered by such solution procedure, as well as an additional test case from liner elasticity, namely the Boussinesq's contact problem.

The paper is organized as follows: In Section 2, the proposed hybrid regular-scattered node positioning algorithm is described, in Section 3, the approximation of linear differential operators using meshless methods is briefly presented, in Section 4 the numerical examples are given. Finally, conclusions and future work opportunities are presented in Section 5.

2. Hybrid regular-scattered node positioning algorithm (HyNP) algorithm

To obtain the hybrid regular-scattered discretization of a d dimensional domain Ω, we propose an extension for the existing variable density scattered node positioning algorithm proposed by Slak and Kosec [3]. The iterative algorithm begins with a given set of seed nodes that are placed in an "expansion queue". In each iteration one node is dequeued and "expanded". The expansion stands for a procedure, where several candidate nodes are uniformly generated on a sphere and then randomly rotated around the expanded node. Candidates that do not violate the proximity criteria (are too close to the existing nodes) and are within the domain, are accepted as new nodes and added to the expansion queue. The iteration continues as long as there are nodes in the expansion queue.

In HyNP, we exploit the advancing front nature of the algorithm to find and fill the regular parts of the domain as shown in Fig. 1. As soon as the advancing front encounters a regular area defined with an user-defined characteristic function

$$g : \Omega \subset \mathbb{R}^d \to \{0, 1\}, \tag{1}$$

where 0 stands for the areas to be populated with scattered nodes and 1 for the areas to be populated with regular nodes. Regular nodes are placed in a similar advancing front fashion as scattered ones [3], where the candidate nodes are positioned regularly around the parent node.

The edges of regular area are then used as seed nodes for further progression of the scattered nodes front. Internodal distance in the regular area is determined by the value of the nodal spacing function

$$h : \Omega \subset \mathbb{R}^d \to (0, \infty) \tag{2}$$

in the first node that is placed within.

Algorithm 1 Hybrid fill algorithm.

Input: A d dimensional domain Ω defined with a characteristic function $\omega : \Omega \subseteq \mathbb{R}^d \to \{0, 1\}$.
Input: A nodal spacing function $h : \Omega \subset \mathbb{R}^d \to (0, \infty)$.
Input: A characteristic function for regular parts of the domain $g : \Omega \subset \mathbb{R}^d \to \{0, 1\}$.
Input: An optional set of boundary and/or seed nodes $X \subseteq \Omega$.
Output: A list of nodes in Ω with regularity based on g and distributed according to spacing function h.

1: **function** HYBRIDFILL(Ω, h, g, X)
2: **if** $\|X\| = 0$ **then**
3: APPEND(X, $p \in \Omega$) ▷ Randomly select a seed node if none were provided.
4: $T_p \leftarrow$ KDTREEINIT(X) ▷ Initialize spatial search structure on points X.
5: $T_r \leftarrow$ KDTREEINIT({}) ▷ **Initialize spatial search structure for removable points.**
6: $toRemove \leftarrow$ {} ▷ **List of nodes to remove due to grid conflicts.**
7: $i \leftarrow 0$ ▷ Current node index.
8: **while** $i < |X|$ **do** ▷ Until the queue is not empty.
9: $p_i \leftarrow X[i]$ ▷ Dequeue current point.
10: **if not** $g(p_i)$ **or** $g(p_i) \neq g(p_{i-1})$ **then** ▷ **For scattered and first grid nodes.**
11: $h_i \leftarrow h(p_i)$ ▷ Compute its nodal spacing.
12: **else**
13: $h_i \leftarrow h_{i-1}$ ▷ **Use previous nodal spacing.**
14: **for each** c **in** CANDIDATES($p_i, h_i, g(p_i)$) **do** ▷ **Generate new candidates.**
15: **if** $c \in \Omega$ **then** ▷ Discard candidates outside the domain.
16: $n_p, d_p \leftarrow$ KDTREECLOSEST(T_p, c) ▷ Find nearest permanent node index and distance.
17: **if** $d_p \geq h_i$ **then** ▷ Test that the candidate is far enough.
18: $n_r, d_r \leftarrow$ KDTREECLOSEST(T_r, c) ▷ **Find the nearest removable.**
19: **if** $g(c)$ **then** ▷ **In regular part of the domain.**
20: PREPEND(X, c) ▷ **Enqueue c as the first element of X.**
21: KDTREEINSERT(T_p, c) ▷ **Insert c into the permanent search structure.**
22: **while** $d_r < h_i$ **do** ▷ **Keep removing the removables while in conflict.**
23: APPEND($toRemove, n_r$) ▷ **Append to the list of conflicting nodes.**
24: KDTREEREMOVE(T_r, n_r) ▷ **Remove from the search structure.**
25: $n_r, d_r \leftarrow$ KDTREECLOSEST(T_r, c) ▷ **Find the next closest removable.**
26: **else**
27: **if** $d_p \geq h_i$ **then** ▷ Test that the candidate is far enough.
28: APPEND(X, c) ▷ Enqueue c as the last element of X.
29: KDTREEINSERT(T_r, c) ▷ Insert c into the removable spatial search structure.
30: $i \leftarrow i + 1$ ▷ Move to the next non-expanded node.
31: REMOVE($X, toRemove$) ▷ Remove the conflicting nodes.
32: **return** X

The modified parts of the Algorithm 1 are highlighted by bold pseudocode comments. The main difference is how the advancing front candidates are generated on line 14. The algorithm uses $k = 15$ randomly placed candidates on a hypersphere around the seed point in the scattered part of the domain and $k = 2d$ candidates $\boldsymbol{c}$ at a distance h from $\boldsymbol{p}$, computed as given in lines 10–13 of Algorithm 1 along a standard basis $\hat{\boldsymbol{e}}_i$ that is,

$$\boldsymbol{c} = \boldsymbol{p} \pm h\hat{\boldsymbol{e}}_i; \qquad i = 1, \ldots, d. \tag{3}$$

The regular candidate basis could easily be variable throughout the domain, allowing for the regular regions to better match the domain description. Note that the regular nodes have priority, i.e. if a previously placed non-seed scattered node would prevent a grid node to be placed within a regular region, the already accepted scattered node is

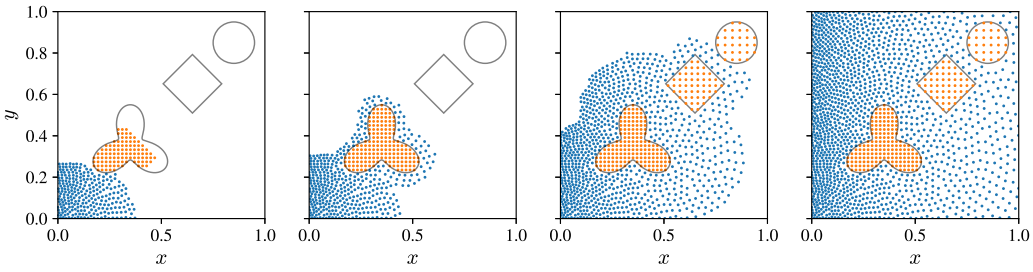

Fig. 1. A visualization of hybrid fill algorithm progression on a domain with variable node density and irregularly shaped areas with regular node positioning.

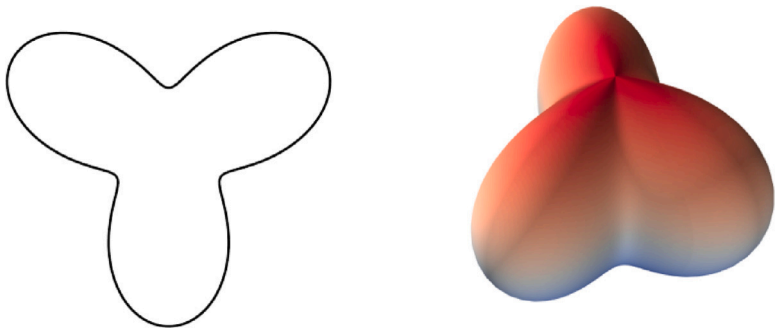

Fig. 2. Visualization of two and three dimensional irregularly shaped clover domains.

removed. This ensures that the maximum possible area is covered by one continuous grid of regular nodes and minimizes issues caused by re-entrant grids.

2.1. Evaluation of discretization quality

To assess the potential degradation of node quality due to combining the two different discretization types we assess separation distance metrics of different orders. The first metric is the distance between the ith node and its jth closest neighbour

$$d_{i,j} = \|\boldsymbol{p}_i - \boldsymbol{p}_{n(i,j)}\|, \tag{4}$$

where $n(i,j)$ is the index of jth closest neighbour for node i. The second metric is a measure of empty space between nodes s_j, i.e., the diameters of the largest hyperspheres that can be inscribed in the empty space between generated nodes. The diameters

$$s_j = 2 \min_i \|\boldsymbol{p}_i - \boldsymbol{v}_j\|, \tag{5}$$

are then determined by constructing a Voronoi diagram seeded by node positions $\boldsymbol{p}$ and calculating the distance between the vertex position $\boldsymbol{v}_j$ and the position of its closest discretization node $\boldsymbol{p}_i$ for all vertices in the Voronoi diagram.

The two metrics are compared on a square domain with an irregularly shaped grid type characteristic function g based on two and three dimensional clover-like shape shown in Fig. 2. The parametric surface of the boundary $\delta\Theta_\square$ of clover-like shape $\Theta_\square$, depends on a scaling parameter l and is given by the following expressions

$$r(l,\varphi) = \frac{l}{3/2}\left[1 - \frac{2}{3}\cos^2\left(\frac{3}{2}\left(\varphi - \frac{\pi}{6}\right)\right)\right], \tag{6}$$

$$\delta\Theta_{2D} = r(l,\varphi)\{\cos(\varphi), \sin(\varphi)\} \quad \text{for} \quad \varphi \in [0, 2\pi], \tag{7}$$

in 2D and

$$r(l,\varphi,\vartheta) = \frac{l}{3/2}\left[1 - \frac{2}{3}\cos^2\left(\frac{3}{2}\left(\varphi - \frac{\pi}{6}\right)\right)\frac{\vartheta(\pi-\vartheta)}{3}\right], \tag{8}$$

$$\delta\Theta_{3D} = r(l,\varphi,\vartheta)\{\cos(\varphi)\sin(\vartheta), \sin(\varphi)\sin(\vartheta), \cos(\vartheta)\} \quad \text{for} \quad \varphi \in [0, 2\pi] \quad \vartheta \in [0, \pi] \tag{9}$$

in 3D with scaling $l = 0.2$. These parametrizations give the boundary between scattered and regular discretization. The parametric definition is practical as it allows for a relatively simple interior check and for the surface $\delta\Theta_\square$ to be populated with nodes using a specialized parametric surface node positioning algorithm [20]. This clover-like shape will also be used as an irregularly shaped obstacle in further numerical tests.

We first compare three discretizations of a square $\Omega = [0,1] \times [0,1]$ domain shown in the top row of Fig. 3. The first column shows a fully scattered domain, the second column shows a domain filled with regular nodes inside Θ and scattered outside, while the third column shows the reverse of the second. In all cases the discretization was started with a seed node in the lower left corner and used a constant internodal distance $h = 0.02$. The second row shows the distribution of distances to jth neighbour for different discretizations and the third row the radii of the largest possible inscribed circles. We can confirm that the neighbour distance distribution is comparable between the purely scattered and hybrid fill results — apart from the structural differences that stem from regularity, i.e. node clusters at multiples of h and diagonals ($\sqrt{2}h$, $\sqrt{5}h$). More importantly, the empty space distribution also stays the same, confirming that there is no problem in coupling the two node arrangements on the irregular boundary. Additionally we can confirm that there is no discernable difference between the algorithm starting from regular or scattered sections. The analysis is repeated for a cube $\Omega = [0,1] \times [0,1] \times [0,1]$ in Fig. 4 with matching conclusions thus confirming the dimensional independence of the algorithm.

Note that the proposed hybrid discretization is general in the sense that the internodal distance h is by no means limited to a constant, spatially independent, value. Spatially dependent declaration $h(\boldsymbol{p})$ can be used to employ h-refinement and locally improve the discretization quality where this is necessary. An example of an h-refined domain discretization with clover-like shaped obstructions is shown in Fig. 5. In this example, the internodal distance linearly increases from h_s on the boundary of the clover to h_r which is equals to the spacing of regularly positioned nodes for a smooth transition. For the demonstration purposes, the width of the scattered node layer δ_h in Fig. 5 has been arbitrarily selected and that the construction of $h(\boldsymbol{p})$ is made through nearest-neighbour search structure, specifically, k-d tree.

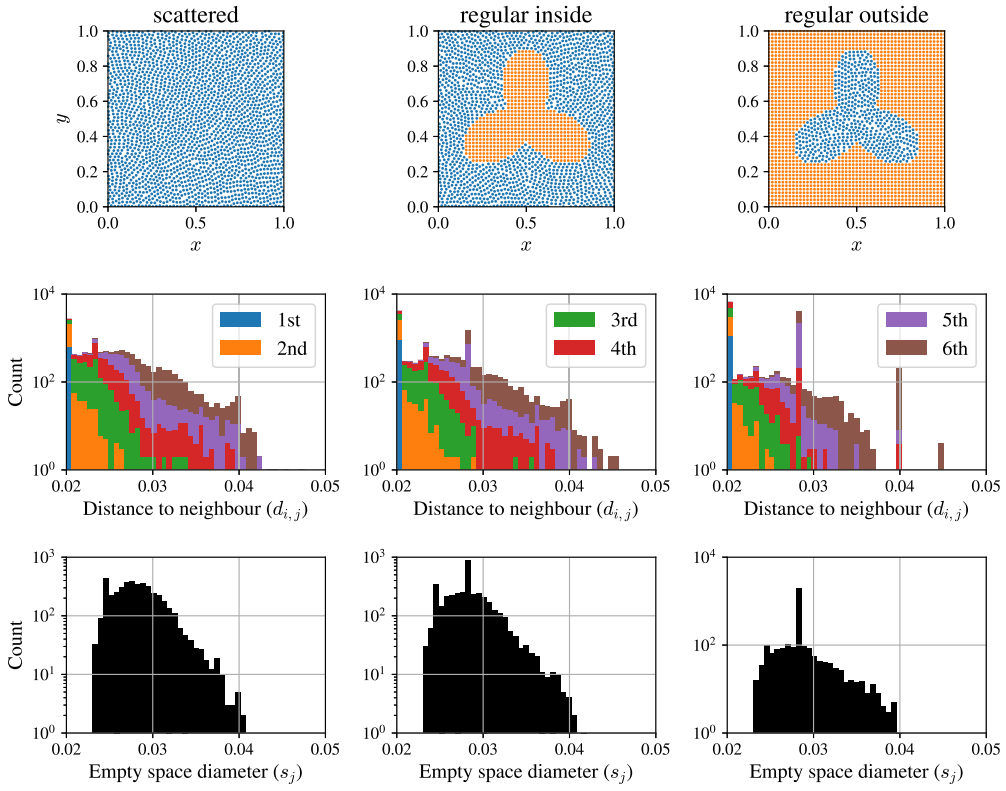

Fig. 3. Visualization of the test domain discretized with different regularity functions g and comparison of fill quality measure distributions between scattered and hybrid fill algorithms on a 2D domain.

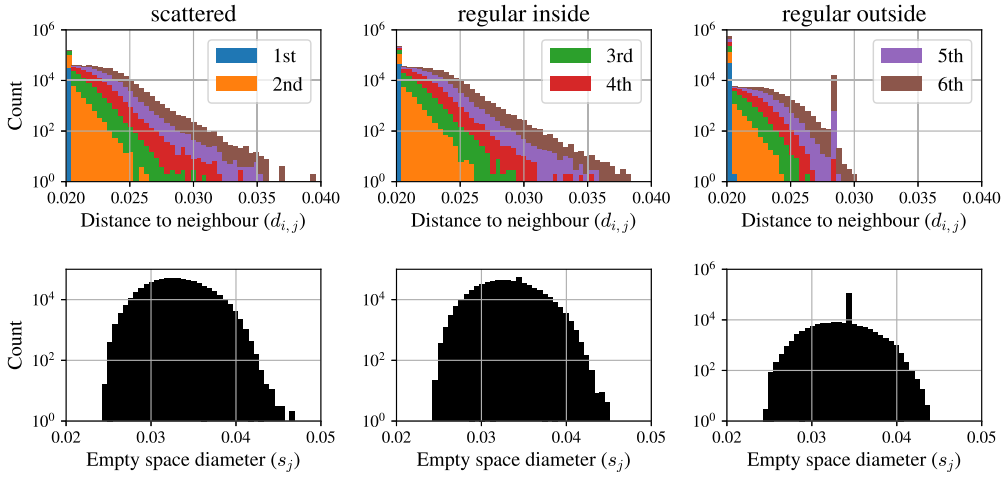

Fig. 4. Comparison of fill quality measure distributions between scattered and hybrid fill algorithms on a 3D domain.

3. Numerical approximation of partial differential equations

With the computational nodes $\boldsymbol{x}_i \in \Omega$ placed using the HyNP algorithm, the differential operators $\mathcal{L}$ can be locally approximated in point $\boldsymbol{x}_c$ over a set of n neighbouring nodes (stencil) $\{\boldsymbol{x}_i\}_{i=1}^n = \mathcal{N}$, using the following expression

$$(\mathcal{L}u)(\boldsymbol{x}_c) \approx \sum_{i=1}^{n} w_i u(\boldsymbol{x}_i). \tag{10}$$

The approximation (10) holds for an arbitrary function u and yet to be determined vector of weights $\boldsymbol{w}$. To determine the weights, the equality of approximation (10) is enforced for a chosen set of basis functions. Here we will use two variants

(i) The first setup uses a shape parameter free Polyharmonic spline (PHS)

$$\varphi(r) = \begin{cases} r^k, & k \text{ odd} \\ r^k \log r, & k \text{ even} \end{cases}, \tag{11}$$

basis augmented with polynomials effectively resulting in a popular *radial basis function-generated finite differences* (RBF-FD) approximation method [7]. Such approximation necessarily reproduces polynomials up to the given order (the order of augmenting monomials), in other words, it is exact for augmenting polynomials, i.e. the approximation is of the same order as the polynomial augmentation [6]. This has been discussed, analysed and demonstrated in several recent publications [6,9,21–23], including the recently introduced hp-adaptive meshless method,

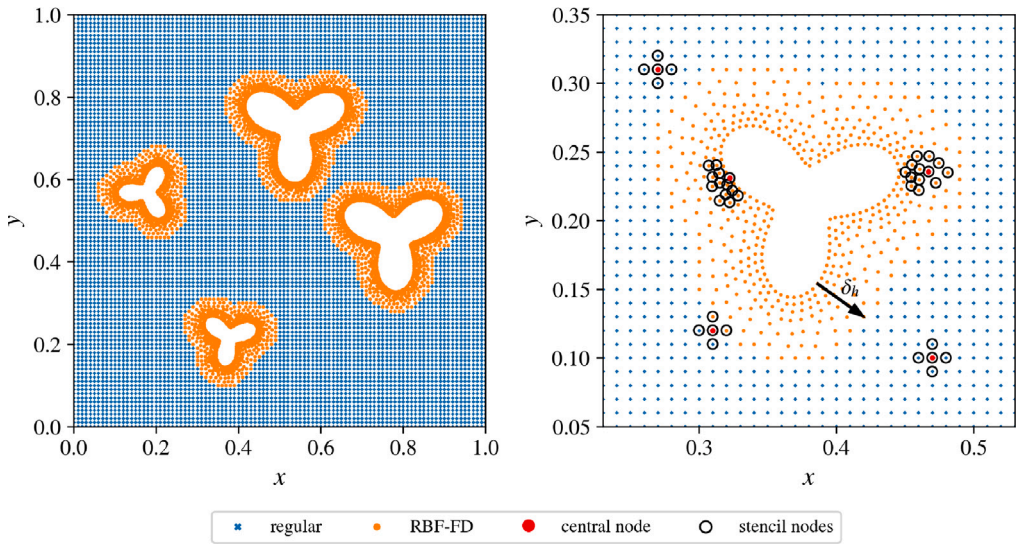

Fig. 5. Irregular domain discretization example *(left)* and spatial distribution of approximation methods along with corresponding example stencils *(right)*.

where authors dynamically adjust the order of the method via order of augmenting monomials [24].

(ii) Second setup uses a set Gaussian functions

$$g(r) = \exp(-r^2/\sigma^2) \tag{12}$$

centred at the stencil nodes, where σ is shape parameter that has to be determined specifically for each stencil size. We refer to this setup to as a local Radial Basis Function Collocation method (LRBFCM) [12,25,26]. In this case, the convergence behaviour is no longer as clear as in the case of augmented approximation. In an experimental study, Ding et al. [27] showed that the error estimate scales with $O(h/\sigma)$. In [27], authors also discussed the dependence of the error estimate on the stencil size, which was later refined by Bayona et al. [28], who found that the method is of second order for stencil sizes between 5 and 12. In both papers [27,28] authors experimented with second-order Poisson's PDE in 2D.

The LRBFCM setup is computationally efficient, but only stable on regular nodes [11,29]. Unless otherwise specified, in a d-dimensional domain the LRBFCM method will be employed using $2d+1$ Gaussian functions using shape parameter $\sigma = 90$. A stencil size equal to the size of the corresponding RBF basis pool is required. For the RBF-FD part, we also resort to the minimal configuration required for 2nd-order operators, i.e., 3rd-order PHS augmented with all monomials up to the 2nd-order ($m = 2$) and we can assume that the approximation of partial differential operators used in this work is of a second order [6,28]. According to the standard recommendations [6], this requires a stencil size of $n = 2\binom{m+d}{m}$ for a stable approximation.

The weights $\boldsymbol{w}$ are calculated by imposing equality in Eq. (10) and solving a system of linear equations $\boldsymbol{A}\boldsymbol{w} = \boldsymbol{b}$ for each computational node

$$\begin{bmatrix} \phi_{1,1} & \cdots & \phi_{1,n} \\ \vdots & \ddots & \vdots \\ \phi_{n,1} & \cdots & \phi_{n,n} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} (\mathcal{L}\phi_{1,1})|_{\boldsymbol{p}_1} \\ \vdots \\ (\mathcal{L}\phi_{1,n})|_{\boldsymbol{p}_1} \end{bmatrix}, \tag{13}$$

where

$$\phi_{i,j} = \Lambda\left(\frac{\|\boldsymbol{p}_i - \boldsymbol{p}_j\|}{\|\boldsymbol{p}_1 - \boldsymbol{p}_2\|}\right) \tag{14}$$

are radial basis functions written with centralized stencil positions $\boldsymbol{p}$ and normalized with the distance between the central node position $\boldsymbol{p}_1$ and the closest stencil position $\boldsymbol{p}_2$. The basis $\Lambda(r) = g(r)$ for LRBFCM and $\Lambda(r) = \varphi(r)$ for RBF-FD. The system for the latter requires an additional augmentation with $s = \binom{m+d}{d}$ monomials q up to the mth order to ensure positive definiteness

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{Q} \\ \boldsymbol{Q}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{w} \\ \lambda \end{bmatrix} = \begin{bmatrix} \boldsymbol{b} \\ c \end{bmatrix},$$
$$\boldsymbol{Q} = \begin{bmatrix} q_1(\boldsymbol{p}_1) & \cdots & q_s(\boldsymbol{p}_1) \\ \vdots & \ddots & \vdots \\ q_1(\boldsymbol{p}_n) & \cdots & q_s(\boldsymbol{p}_n) \end{bmatrix}, \boldsymbol{c} = \begin{bmatrix} (\mathcal{L}q_1)|_{\boldsymbol{p}_1} \\ \vdots \\ (\mathcal{L}q_s)|_{\boldsymbol{p}_1} \end{bmatrix}. \tag{15}$$

The redundant part of the weight vector λ is discarded after computation.

It is important to note the difference in required stencil sizes – 5 vs. 12 nodes in 2D – that only increases in higher dimensions (7 vs. 30 in 3D). This results both in faster computation of the weights $\boldsymbol{w}$ (an $\mathcal{O}(N^3)$[1] operation performed only once for each stencil), in faster evaluation for the $\mathcal{O}(n)$ explicit operator approximation (10) performed many times during the explicit time stepping, and in faster solving of the sparse linear systems.

3.1. Computational stability

By enforcing the equality of approximation (10), we obtain a linear system $\mathbf{M}\boldsymbol{w} = \boldsymbol{\ell}$. Solving the system provides us with the approximation weights $\boldsymbol{w}$, but the stability of such procedure can be uncertain and is usually estimated via the condition number $\kappa(\mathbf{M}) = \|\mathbf{M}\| \left\|\mathbf{M}^{-1}\right\|$ of matrix $\mathbf{M}$, where $\|\cdot\|$ denotes the L^2 norm.

A spatial distribution of condition numbers is shown in Fig. 6. It can be observed that the RBF-FD approximation method generally results in higher condition numbers than the LRBFCM approach. This could be due to the fact that the matrices $\mathbf{M}$ for the RBF-FD part are significantly larger and based on scattered nodes. Nevertheless, it is important to observe that the transition from regular to scattered nodes does not appear to affect the conditionality of the matrices.

3.2. Implementation details

We used `g++ 11.3.0 for Linux` to compile the code with `-O3 -DNDEBUG` flags on `Intel(R) Xeon(R) CPU E5520` computer. To improve the timing accuracy we run the otherwise parallel code in a single thread with the CPU frequency fixed at 2.27 GHz, disabled boost

[1] $N_{\text{RBF-FD}} \sim 3N_{\text{LRBFCM}}$ due to the larger stencil size and the extra PHS in the approximation basis.

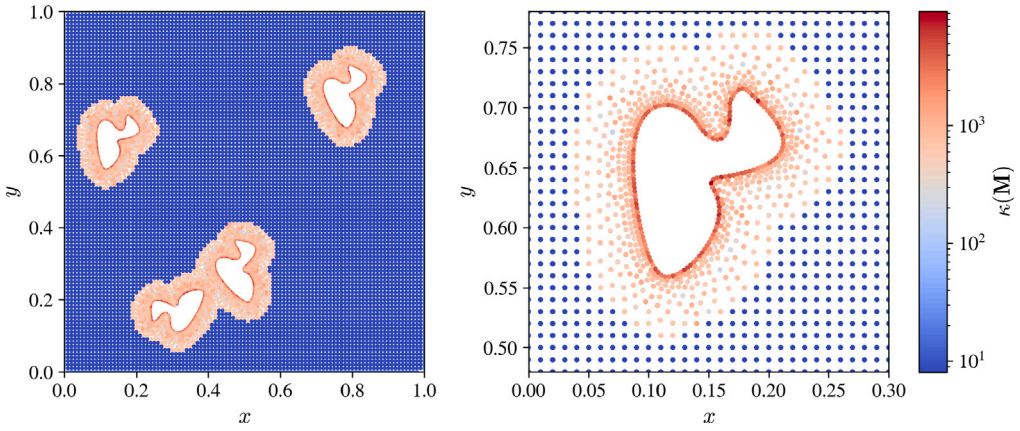

Fig. 6. Condition numbers $\kappa(\mathbf{M})$ for the Laplacian operator: entire computational domain *(left)* and a zoomed-in section around the irregularly shaped obstacle *(right)*.

functionality and assured CPU affinity using the `taskset` command. Post-processing was done using Python 3.10.6 and Jupyter notebooks, also available in the provided git repository.[2]

4. Numerical examples

4.1. Natural convection problem

To objectively assess the advantages of the hybrid discretization method, we first address non-linear natural convection problem that is governed by a system of three PDEs that describe the continuity of mass, the conservation of momentum and the transfer of heat

$$\nabla \cdot \boldsymbol{v} = 0, \tag{16}$$

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \frac{1}{\mathrm{Re}} \nabla \cdot (\nabla \boldsymbol{v}) - g T_{\Delta}, \tag{17}$$

$$\frac{\partial T}{\partial t} + \boldsymbol{v} \cdot \nabla T = \frac{1}{\mathrm{RePr}} \nabla \cdot (\nabla T), \tag{18}$$

$\boldsymbol{v}$ is the velocity vector, p the pressure, T the temperature, and T_{Δ} the offset from reference temperature. The equations are written in a dimensionless form using Reynolds (Re) and Prandtl (Pr) numbers [12, 30] while the results are expressed in terms ob the Rayleigh (Ra) number using the $\mathrm{Ra} = \mathrm{Re}^2\mathrm{Pr}$ relation.

The temporal discretization of the governing equations is solved with the explicit Euler time stepping where we first update the velocity using the previous step temperature field in the Boussinesq term [31]. The pressure-velocity coupling is performed using the Chorin's projection method [32] under the premise that the pressure term of the Navier–Stokes equation can be treated separately from other forces and used to impose the incompressibility condition. The time step is a function of internodal spacing h, and is defined as $\mathrm{d}t = \frac{h}{4}$ to assure stability.

The problem is solved on different geometries employing (i) LRBFCM, (ii) RBF-FD and (iii) their spatially-varying combination. The performance of each approach is evaluated in terms of accuracy of the numerical solution and execution times.

4.1.1. The de Vahl Davis problem

First, we solve the standard de Vahl Davis benchmark problem [33]. The main purpose of solving this problem is to establish confidence in the presented solution procedure and to shed some light on the behaviour of considered approximation methods, the stability of the solution procedure and finally on the computational efficiency. Furthermore, the de Vahl Davis problem was chosen as the basic test case, because the regularity of the domain shape allows us to efficiently discretize it using exclusively scattered or regular nodes and compare the solutions to that obtained with the hybrid regular-scattered discretization.

For a schematic representation of the problem, see Fig. 7 *(left)*. The domain is a unit box $\Omega = [0, 1] \times [0, 1]$, where the left wall is kept at a constant temperature $T_C = -0.5$, while the right wall is kept at a higher constant temperature $T_H = 0.5$. The upper and lower boundaries are insulated, and no-slip boundary condition for velocity is imposed on all walls. Both the velocity and temperature fields are initially set to zero.

To test the performance of the proposed hybrid regular-scattered approximation method, we divide the domain Ω into quarters, where each quarter is discretized using either scattered or regular nodes — see Fig. 7 *(right)* for clarity.

An example solution for $\mathrm{Ra} = 10^6$ and $\mathrm{Pr} = 0.71$ at a dimensionless time $t = 80$ with approximately $N \approx 10\,000$ discretization nodes is shown in Fig. 8.

We use the Nusselt number – the ratio between convective and conductive heat transfer – to determine when a steady state has been reached and as a convenient scalar value for comparison with reference solutions. In the following analyses, the average Nusselt number ($\overline{\mathrm{Nu}}$) is calculated as the average of the Nusselt values at the cold wall nodes

$$\mathrm{Nu} = \frac{L}{T_H - T_C} \left| \frac{\partial T}{\partial \boldsymbol{n}} \right|_{x=0}. \tag{19}$$

Its evolution over time is shown in Fig. 9. In addition, three reference results are also added to the figure. We are pleased to see that our results are in good agreement with the reference solutions from the literature.

Moreover, Fig. 9 also shows the time evolution of the average Nusselt number value for cases where the entire domain is discretized using either scattered or regular nodes. We find that all – hybrid, purely scattered and purely regular domain discretizations – yield results in good agreement with the references. More importantly, the hybrid method shows significantly shorter computational time than that required by the scattered discretization employing RBF-FD, as can be seen in Table 2 for the densest considered discretization with $h = 0.00364$.

To further validate the hybrid method, we show in Fig. 10 the vertical component of the velocity field across the section $y = 0.5$. It is important to observe that the results for the hybrid, scattered and regular approaches overlap, which means that the resulting velocity fields for the three approaches are indeed comparable.

As a final remark, we also study the convergence of the average Nusselt number with respect to the number of discretization nodes in Fig. 11, where we confirm that all our discretization strategies converge

[2] Source code is available at https://gitlab.com/e62Lab/public/2023_cp_iccs_hybrid_nodes under tag *v1.3*.

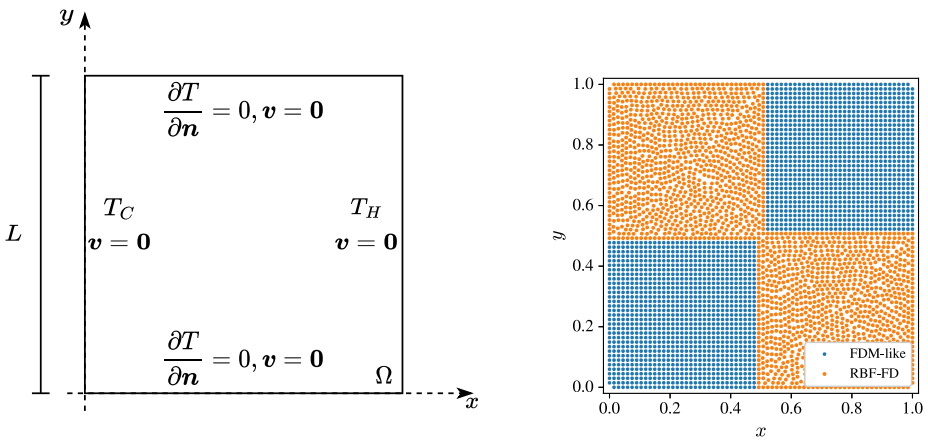

Fig. 7. The de Vahl Davis sketch *(left)* and example hybrid regular-scattered domain discretization *(right)*.

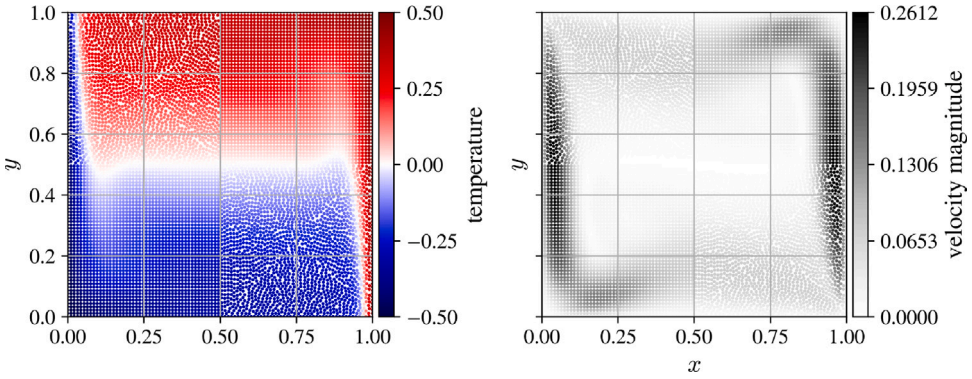

Fig. 8. Example solution in the stationary state. Temperature field *(left)* and velocity magnitude *(right)*.

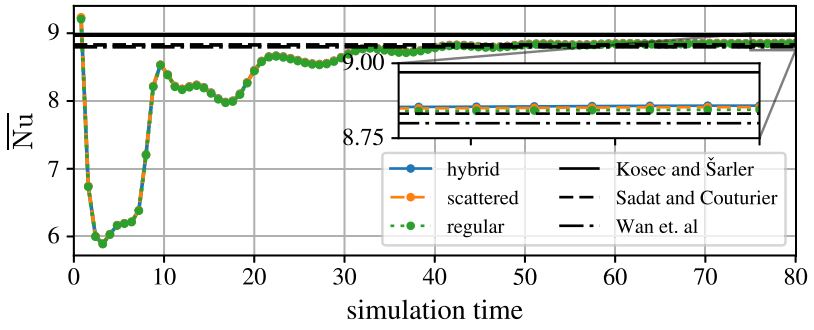

Fig. 9. Time evolution of the average Nusselt number along the cold edge calculated with the densest considered discretization. Three reference results Kosec and Šarler [12], Sadat and Couturier [34] and Wan et al. [35] are also added.

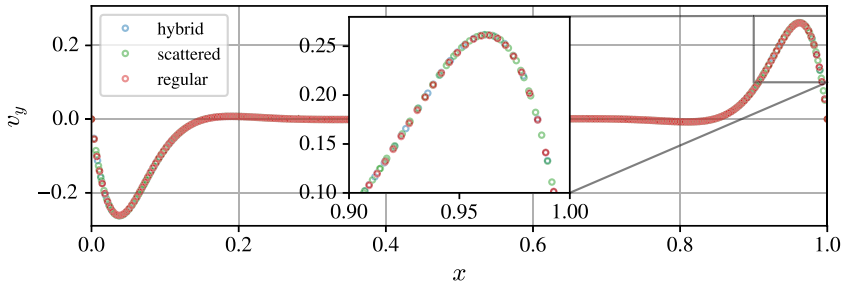

Fig. 10. Vertical velocity component values at nodes close to the vertical midpoint of the domain, i.e., $|y - 0.5| \leq h$ for purely scattered, purely regular and hybrid discretizations.

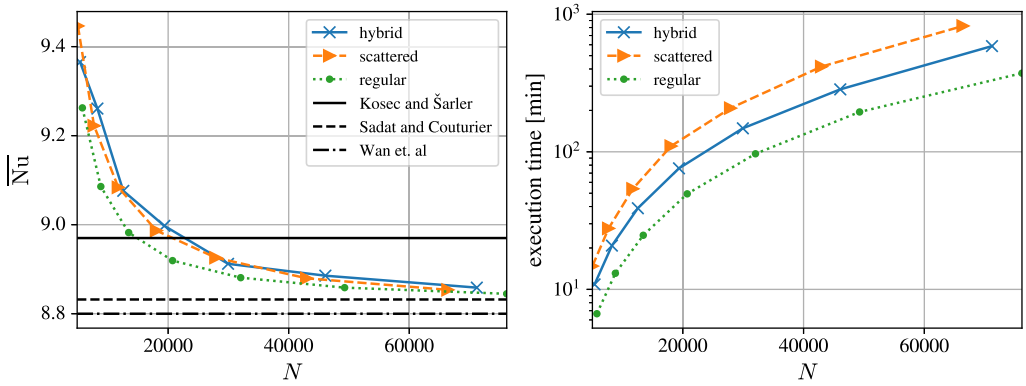

Fig. 11. Convergence of average Nusselt number with respect to discretization quality *(left)* and corresponding execution times *(right)*.

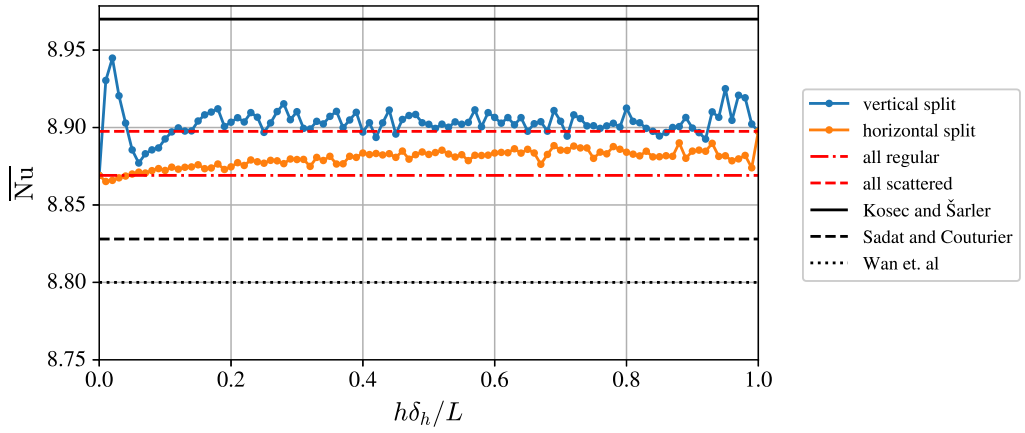

Fig. 12. Demonstration of the scattered node layer width (δ_h) effect on the accuracy of the numerical solution.

Table 2
Average Nusselt along the cold edge along with execution times and number of discretization nodes.

Approximation	$\overline{\mathrm{Nu}}$	Execution time [h]	N
Scattered	8.854	13.7	66 406
Regular	8.845	6.2	76 172
Hybrid	8.856	9.8	71 209
Kosec and Šarler (2007) [12]	8.97	/	10 201
Sadat and Couturier (2000) [34]	8.828	/	22 801
Wan et. al. (2001) [35]	8.8	/	10 201

to a similar value that is consistent with the reference values. Moreover, to evaluate the computational efficiency of the hybrid approach, the execution times are shown on the right. Note that the same values for h were used for all discretization strategies and the difference in the total number of nodes is caused by the lower density of scattered nodes at the same internodal distance.

4.1.2. The effect of the scattered node layer width δ_h

To study the effect of the width of the scattered node layer δ_h, we consider two cases. In both cases, the domain from Fig. 7 is split into two parts at a distance $h\delta_h$ from the origin in the lower left corner. In the first scenario, the split is horizontal, resulting in scattered nodes below the imaginary split and regular nodes above it. In the second scenario, the split is vertical, resulting in scattered nodes to the left of it and regular nodes to the right of it. In both cases, the domain is discretized with purely regular nodes when $h\delta_h = 0$ and with purely scattered nodes when $h\delta_h = L$.

In Fig. 12, we show how the width of the scattered node layer affects the average Nusselt number in stationary state for approximately 40 000 discretization nodes. It is clear that even the smallest values of δ_h yield satisfying results. However, it is interesting to observe that the accuracy is most affected when the boundary between regular and scattered nodes runs across the region with the largest velocity magnitudes, i.e., the first and last couple of vertical split data points in Fig. 12.

4.1.3. Natural convection on irregularly shaped domains

In the previous section we demonstrated that the hybrid regular-scattered approximation method is computationally more efficient than the pure RBF-FD approximation with only minor differences in the resulting fields. However, to truly exploit the advantages of the hybrid method, irregular domains must be studied. Therefore, in this section, the hybrid regular-scattered approach is employed on an irregularly shaped domain. Let the computational domain Ω be a difference between the two-dimensional unit box $\Omega = [0,1] \times [0,1]$ and 4 randomly positioned and sized clover-shaped obstacles Θ defined in Eq. (7).

The dynamics of the problem are governed by the same set of Eqs. (16)–(18) as in the previous section. This time, however, all the boundaries of the box are insulated. The obstacles, on the other hand, are subject to Dirichlet boundary conditions, with half of them at $T_C = -0.5$ and the other half at $T_H = 0.5$. The initial temperature is set to $T_{\text{init}} = 0$.

We have chosen such a problem because it allows us to further explore the advantages of the proposed hybrid regular-scattered discretization. Generally speaking, the clover-shaped obstacles within the computational domain represent an arbitrarily complex shape that requires scattered nodes for accurate description, i.e., reduced discretization-related error.

Moreover, by using scattered nodes near the irregularly shaped domain boundaries, we can further improve the local field description in their vicinity by employing a h-refined discretization. Specifically,

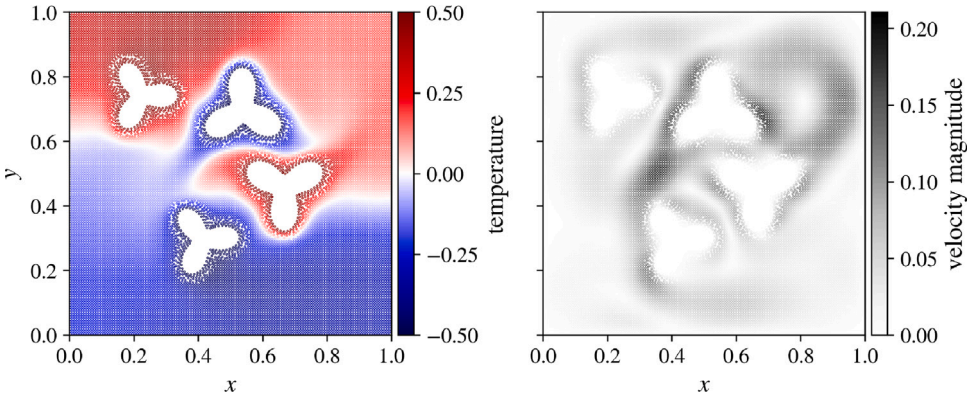

Fig. 13. Example solution on irregular domain. Temperature field *(left)* and velocity magnitude *(right)*.

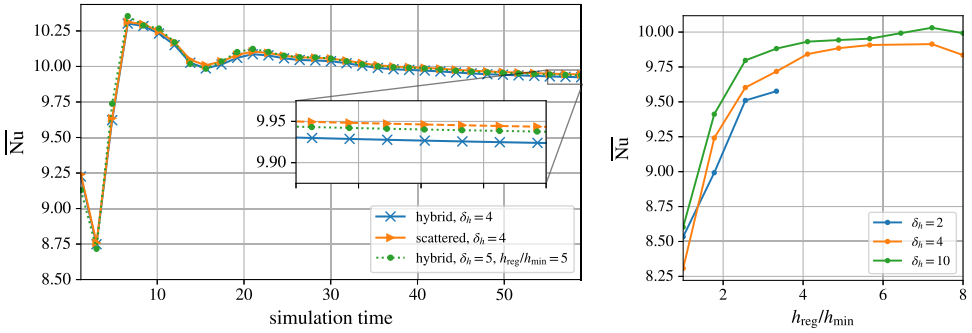

Fig. 14. Time evolution of the average Nusselt number calculated on the cold clover-shaped obstacles of an irregularly shaped domain (*left*) and average Nusselt number for different combinations of refine parameters (*right*).

Table 3
Average Nusselt along the cold edges of the clover along with execution times. Note that both scattered and hybrid in the table were obtained for $\delta_h = 4$, while refined hybrid was obtained for $\delta_h = 5$ and $h_{reg}/h_{min} = 5$.

Approximation	$\overline{Nu}$	Execution time [h]	N
Scattered	9.942	14.21	83 530
Hybrid	9.922	7.17	95 292
Refined hybrid	9.936	1.52	10 952

we employ h-refinement towards the obstacles with linearly decreasing internodal distance from h_{reg} (regular nodes) towards h_{min} (irregular boundary) over a distance of $h_{reg}\delta_h$. Example discretization is shown in Fig. 5 for a scattered node layer width $\delta_h = 4$ and $h_{reg} = 2h_{min} = 0.01$, yielding approximately 10 500 computational points.

Fig. 13 shows an example solution for an irregularly shaped domain. The hybrid scattered-regular solution procedure was again able to obtain a reasonable numerical solution. Fig. 14 *(left)* shows the average Nusselt number along the cold clover edges where we can observe that a stationary state has been reached. The steady state values for all three considered discretizations match closely. It is perhaps more important to note that the execution times gathered in Table 3 show that the uniform density hybrid method effectively reduces the execution time for ∼50% and that the aggressively refined hybrid discretization for ∼90%. The purely regular LRBFCM approximation is omitted from the table as it cannot discretize irregular domains.

The unrefined convergence and computational times are presented in Fig. 15. The results confirm that both the hybrid and the regular discretization converge to a similar Nusselt value and that the hybrid is consistently faster at the same node count while returning a slightly lower value.

Before continuing with refined discretization we look at interplay between the width of the scattered node layer δ_h and aggressiveness of refine ratio $\frac{h_{reg}}{h_{min}}$ shown in Fig. 14 *(right)*. The results confirm our observations from the original work, that the width does not have a significant impact on the result as long as it is wide enough to avoid instability for the selected ratio but there is a slight systematic offset.

We repeat the convergence study for refined discretizations with results shown in Fig. 16. We chose two hybrid discretizations – one with less aggressive $\delta_h = 3$, $\frac{h_{reg}}{h_{min}} = 2$ refine and another larger $\delta_h = 5$, $\frac{h_{reg}}{h_{min}} = 5$ – and a scattered discretization with a directly comparable set of parameters. The results on the left graph of Fig. 16 show that the refined density solutions are faster to converge to the final Nusselt value, as expected, with the aggressively refined hybrid being significantly faster (also demonstrated in Table 3). The corresponding execution time graph on the right seems more surprising at the first glance due to the refined solutions – even hybrid ones – exhibiting comparable or longer times than the uniform scattered discretization but this can be explained by the time-step that is a function of h_{min}.

The true performance in achieving an accurate solution is easier to determine from a graph of average Nusselt number versus the execution time shown in Fig. 17. The results show that the hybrid discretization is slightly faster than the scattered one with comparable refinement and that we can calculate accurate results significantly faster by using a strongly refined hybrid discretization. Surprisingly there is not much difference between weakly refined and unrefined hybrid/scattered approaches.

4.1.4. Natural convection in three-dimensional domains

The de Vahl Davis test is defined on a unit square domain $\Omega = [0, 1] \times [0, 1] \times [0, 1]$., where vertical walls are kept at constant temperatures, while horizontal walls and front/back walls are adiabatic. No-slip velocity boundary conditions are prescribed on all walls. The dynamics are governed by the same set of Eqs. (16)–(18) as in the 2D case from Section 4.1.3.

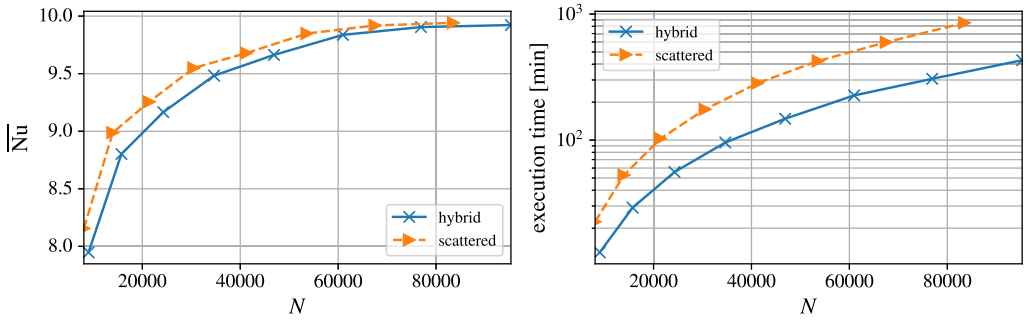

Fig. 15. Convergence of average Nusselt number computed on the cold clover-shaped obstacles (left) accompanied with computational times (right).

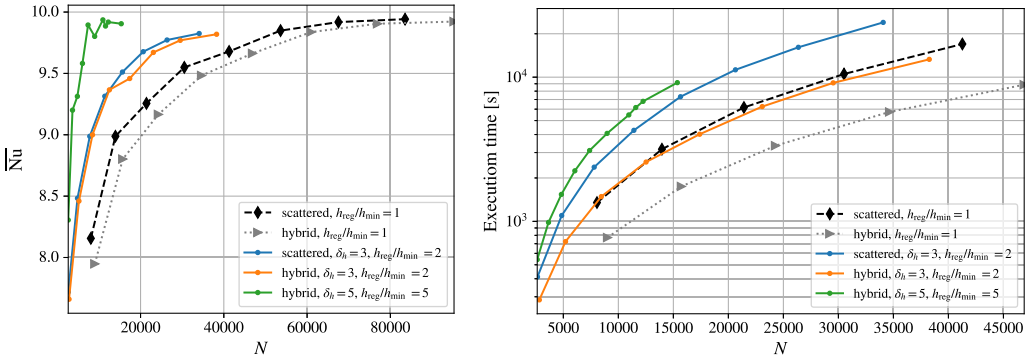

Fig. 16. Convergence of average Nusselt number computed on the cold clover-shaped obstacles *(left)* accompanied with computational times *(right)* for refined discretizations.

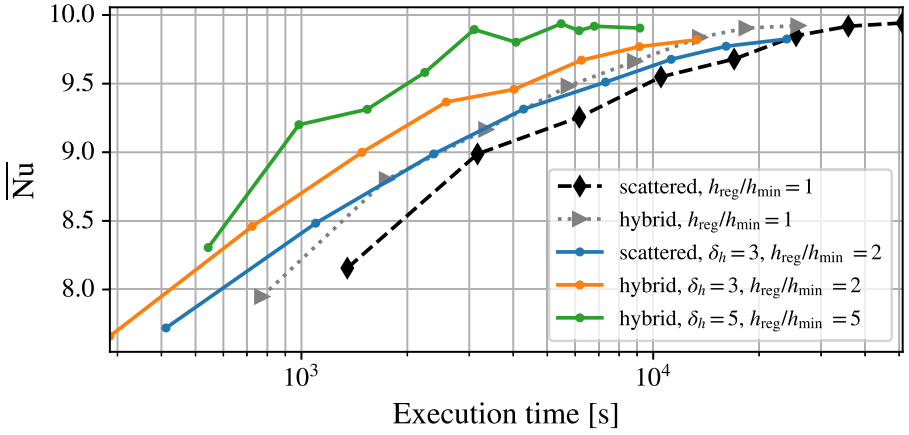

Fig. 17. Average Nusselt number calculated on the cold clover shaped obstacles as a function of computational time for a convergence study with a selection of refined and unrefined discretizations.

Table 4
Peak positions and values of cross section velocities for 3D natural convection test and number of computational elements. The last column contains the single thread execution times for different discretization strategies.

Method	$v_z(x_{max}, 0.5, 0.5)$	x_{max}	$v_x(0.5, 0.5, z_{max})$	z_{max}	N	t [h]
Hybrid	0.2523	0.960	0.0807	0.133	88 725	5.1
Regular	0.2322	0.956	0.0801	0.133	96 800	3.1
Scattered	0.2592	0.960	0.0817	0.147	81 218	7.9
Slak & Kosec (2019) [3]	0.2564	0.961	0.0841	0.143	64 000	/
Wang et. al (2017) [36]	0.2556	0.965	0.0816	0.140	125 000	/
Fusegi et. al (1991) [30]	0.2588	0.966	0.0841	0.144	238 328	/

In Fig. 18 the results for Pr = 0.71 and Ra = 10^6 are visualized for all three discretization variants, namely scattered, regular and hybrid. A more quantitative analysis is presented in Table 4 by comparing characteristic values, i.e. peak positions and values of cross section velocities, with published data.

As a final demonstrative example of natural convection problems, we employ the proposed hybrid regular-scattered approximation method on a three-dimensional irregular domain, where we add to the domain $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ also 4 randomly positioned and sized clover-like obstacles Θ defined in Eq. (9).

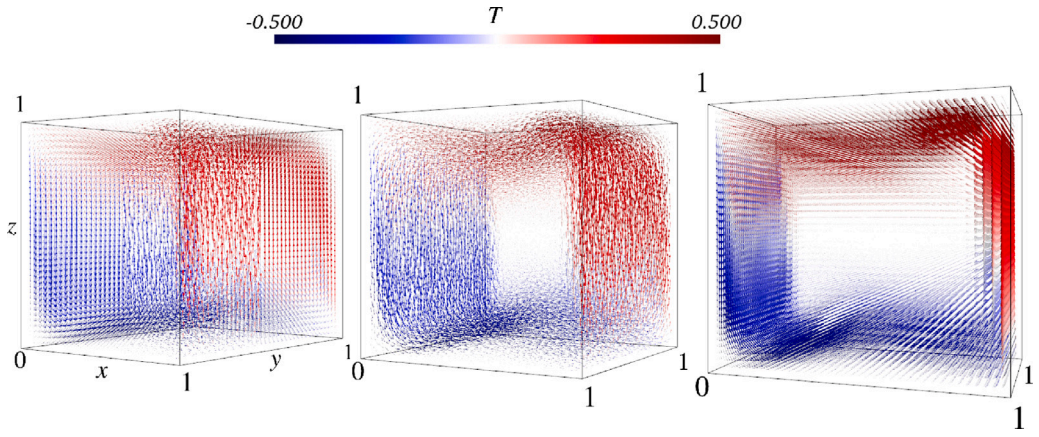

Fig. 18. Example of solution of natural convection test using hybrid (left), scattered (middle) and regular (right) nodes in 3D.

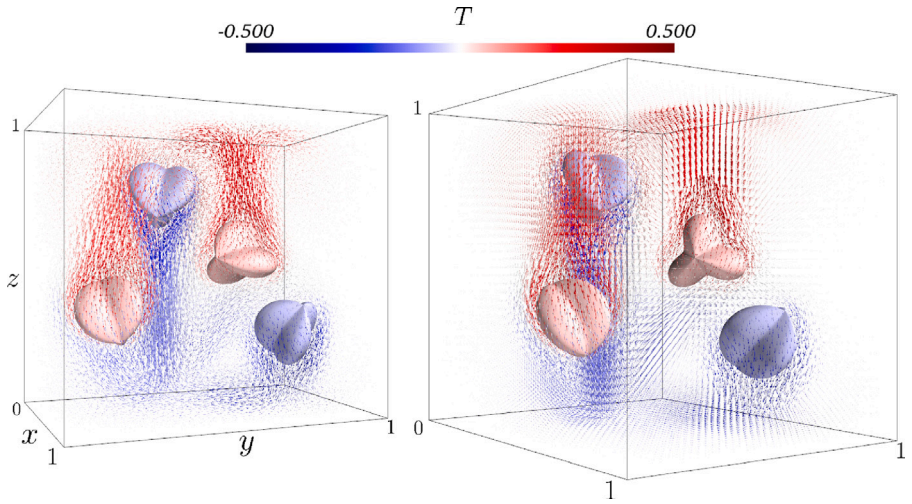

Fig. 19. Examples of natural convection in a 3D irregular domain solved using scattered (left) and hybrid (right) nodes. The arrows show the velocity in computational nodes and are coloured according to the temperature in that node.

To improve the quality of the local field description near the irregularly shaped domain boundaries, h-refinement is employed with a linearly decreasing internodal distance from $h_r = 0.025$ (regular nodes) towards $h_s = 0.018$ (clover shapes). The clover-shaped obstacles were set to a constant temperature, two to $T_C = -0.5$ and two to $T_H = 0.5$. The Rayleigh number was set to 10^6.

Although difficult to visualize, an example solution is shown in Fig. 19. Using the hybrid regular-scattered domain discretization, the solution procedure was again able to obtain a reasonable numerical solution, i.e. the difference in Nusselt number between fully scattered and hybrid approaches is less than 0.5% at the approximately 57% reduction in computation time in favour of the hybrid approach.

4.2. Boussinesq's problem

In this section, the proposed solution procedure is demonstrated on an implicit solution to the three-dimensional Boussinesq's problem [37]. In this problem, a concentrated normal traction acts on an isotropic half-space, as sketched in Fig. 20.

Analytic solution to the problem is given in cylindrical coordinates r, θ and z as

$$u_r = \frac{Pr}{4\pi\mu}\left(\frac{z}{R^3} - \frac{1-2\nu}{R(z+R)}\right), \quad u_\theta = 0, \quad u_z = \frac{P}{4\pi\mu}\left(\frac{2(1-\nu)}{R} + \frac{z^2}{R^3}\right),$$
$$\sigma_{rr} = \frac{P}{2\pi}\left(\frac{1-2\nu}{R(z+R)} - \frac{3r^2 z}{R^5}\right), \quad \sigma_{\theta\theta} = \frac{P(1-2\nu)}{2\pi}\left(\frac{z}{R^3} - \frac{1}{R(z+R)}\right),$$
$$\sigma_{zz} = -\frac{3Pz^3}{2\pi R^5}, \quad \sigma_{rz} = -\frac{3Prz^2}{2\pi R^5}, \quad \sigma_{r\theta} = 0, \quad \sigma_{\theta z} = 0, \tag{20}$$

where P is the magnitude of the concentrated force, ν is the Poisson's ratio, μ is the Lamé parameter and R is the Eucledian distance to the origin. The solution has a singularity at the origin where the concentrated force is applied to the bulk. This makes the problem difficult to solve and consequently a good candidate for treatment with spatially variable node regularity; allowing us to employ h-refined scattered nodes towards the singularity and regular nodes elsewhere.

We consider only a part of the domain, i.e. ε away from the singularity yielding Ω defined as a box, i.e. $\Omega = [-1, -\varepsilon]\times[-1, -\varepsilon]\times[-1, -\varepsilon]$, as schematically shown in Fig. 20. From a numerical point of view, we solve the Navier-Cauchy equation

$$(\lambda + \mu)\nabla(\nabla \cdot \boldsymbol{u}) + \mu\nabla^2\boldsymbol{u} = \boldsymbol{f}, \tag{21}$$

using the Lamé parameters λ and μ with Dirichlet boundary conditions as given in (20).

Even though the analytic solution is given in cylindrical coordinate system, the problem is implemented using cartesian coordinates. For the physical parameters of the problem, the values $P = -1$, $E = 1$ and $\nu = 0.33$ were used.

To employ the hybrid discretization, a cylinder along the edge with applied force is assumed. Points inside the cylinder are scattered, allowing us to employ h-refinement towards the critical edge in the continuation of this work, while regular nodes are positioned elsewhere. To determine if a point $\boldsymbol{p}$ is inside the cylinder with radius R_0, it's

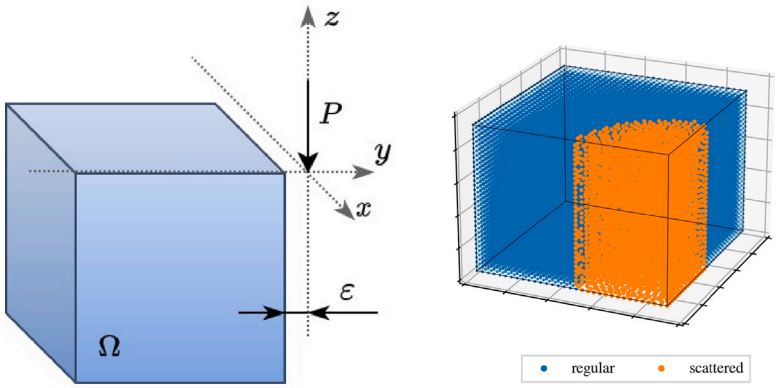

Fig. 20. Schematic presentation of Boussinesq's problem *(left)* and example spatial distribution of computational node regularity *(right)*. For clarity, black lines have been added to mark the domain boundary edges on the right plot.

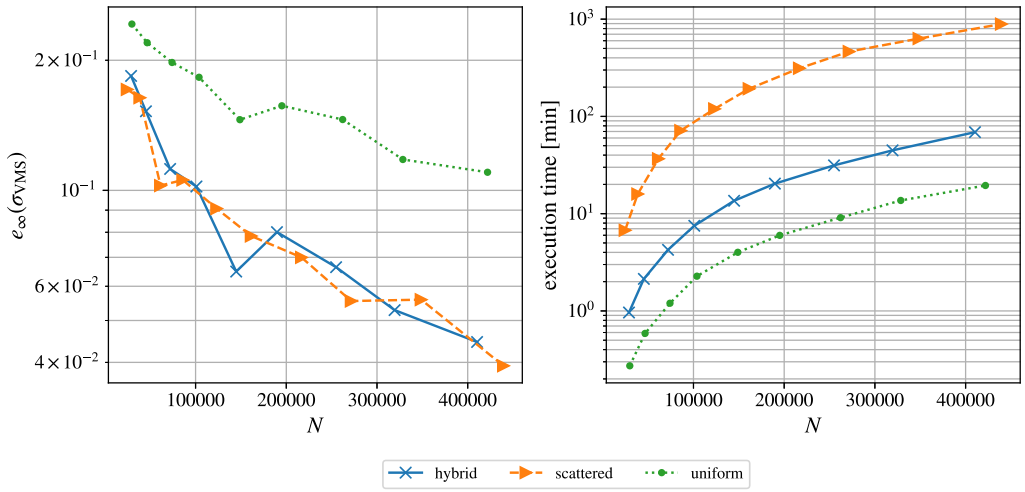

Fig. 21. Convergence analysis using regular, hybrid and scattered discretization in case of Boussinesq's problem *(left)* and execution times *(right)*.

perpendicular distance $\|(\boldsymbol{x}_{\text{corner}} - \boldsymbol{p}) \times \hat{\boldsymbol{e}_z}\|$ to the edge is computed and compared to R_0. Here, $\boldsymbol{x}_{\text{corner}}$ is the domain corner closest to the origin, i.e. $\boldsymbol{x}_{\text{corner}} = (-\varepsilon, -\varepsilon, -\varepsilon)$ and $\hat{\boldsymbol{e}_z}$ is a unit vector along the z axis. For clarity, spatial distribution of the computational node regularity and of the approximation methods are also shown in Fig. 20 *(left)*.

Note that the final sparse system was solved using BiCGSTAB with ILUT preconditioner, where the global tolerance was set to 10^{-16} with a maximum number of 300 iterations and drop-tolerance and fill-factor set to 10^{-6} and 60 respectively. Example solution is shown in Fig. 20 *(left)* with displacement magnitudes on the left and von Mises stress on the right.

The convergent behaviour and the computational gains offered by the proposed solution procedure are studied in Fig. 21. The offset from the origin (ε) was fixed and set to 0.1, the scattered node area was set to $R_0 = 0.4$, and the h-refinement towards the singularity is avoided by setting $h_{\text{reg}} = h_{\text{min}}$ for a fair performance comparison of the different domain discretizations approaches. On the left, we show the infinity norm error in terms of von Mises stresses. We observe that using a purely regular discretization, the error is approximately two times larger compared to fully scattered or hybrid discretizations but retains a similar convergence rate.

Fig. 21 shows the wall-clock times required to obtain the numerical solutions. The LRBFCM approach with purely regular domain discretization is clearly the fastest among the three but also the least accurate. Both hybrid and scattered discretizations show approximately an order of magnitude (or more) longer wall-clock times with comparable accuracy of the numerical solution. Note that the parameter R_0 strongly impacts the computational time required for the hybrid discretization.

In the following set of analyses, the hybrid and scattered discretizations are employed with a linearly decreasing internodal distance $h(\boldsymbol{p})$, from h_{reg} (on regular nodes) to h_{min} (on the edge subjected to force P) depending on the perpendicular distance from the edge with applied force. Thus, the internodal distance h at point $\boldsymbol{p}$ is defined with the following expression:

$$h(\boldsymbol{p}) = \min\left\{ h_{\text{reg}}, h_{\text{min}} + (h_{\text{reg}} - h_{\text{min}}) \frac{\|(\boldsymbol{x}_{\text{corner}} - \boldsymbol{p}) \times \hat{\boldsymbol{e}_z}\|}{R_0} \right\}. \tag{22}$$

In Fig. 22 *(left)*, we study the behaviour in case of different h-refinement aggressiveness. The numerical solutions are again evaluated in terms of the infinity norm of the von Mises stress. We show that the h-refinement towards the edge with applied concentrated force improves the accuracy of the numerical solution by an order of magnitude for both purely scattered and hybrid domain discretization approaches shown for different R_0. We did not encounter any stability related issues in the process – even for the most aggressive h-refinement used, i.e. $h_{\text{reg}}/h_{\text{min}} = 15$. On the other hand the accuracy reaching a plateau with increasing refinement provides an insight into the trade-offs inherently present in the hybrid method. Once the refined part of the domain is discretized with a sufficiently large density of nodes the accuracy becomes bounded by the regular part of the domain. The error plateau exhibited by the scattered and hybrid methods, otherwise discretized for the same $R_0 = 0.4$, directly reflects the difference in method accuracy observed in Fig. 21. With decreasing R_0, the relatively

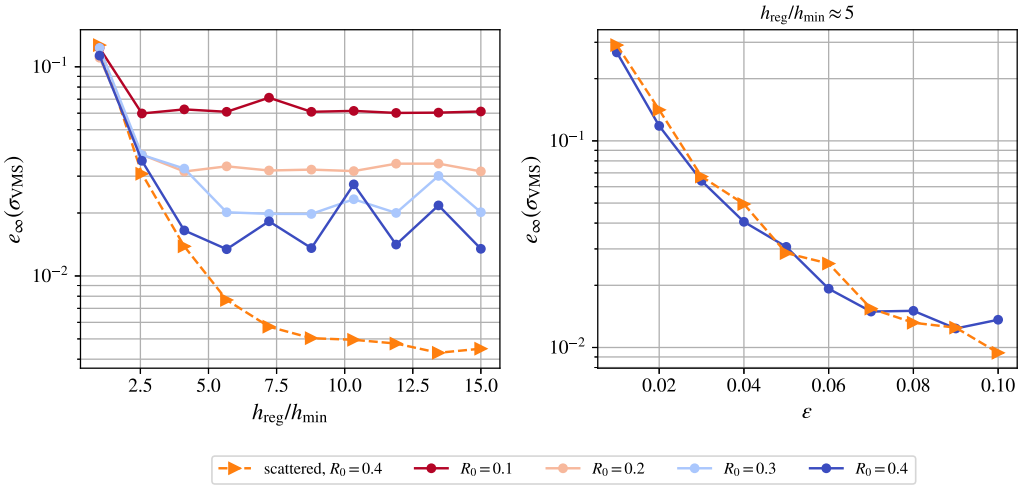

Fig. 22. The impact of h-refine aggressiveness on the maximum error of von Mises stress for scattered and hybrid discretizations with different R_0 *(left)* and the maximum von Mises stress as a function of distance ε between the singularity and the outer edge of the domain *(right)*.

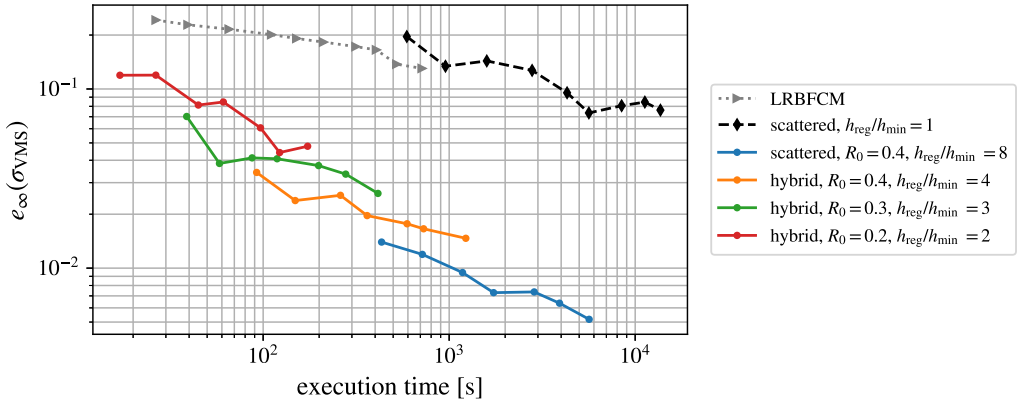

Fig. 23. Maximum error of von Mises stress as a function of computational time for a convergence study with a selection of refined and unrefined discretizations.

weak LRBFCM method is used to discretize areas ever closer to the singularity leading to larger errors irrespective of how well the area in the immediate vicinity of the corner is discretized.

In Fig. 22 *(right)* we show a brief study of hybrid and fully scattered discretization approaches with respect to the domain corner distance ε to the singularity that is present in the origin. As expected, the accuracy of the numerical solutions decreases for smaller distances ε and improves for larger values. It is worth mentioning, that h-refinement following Eq. (22) was used in the process. We used $h_{reg} = (1 - \varepsilon)/40$ and $h_{min} = h_{reg}/5$.

Finally, we can leverage the insights gathered from discretization parameter analysis to repeat the convergence study with a refined scattered and hybrid discretization. We chose 4 configurations to analyse: scattered with $R_0 = 0.4$, $\frac{h_{reg}}{h_{min}} = 8$ as the most accurate, hybrid with $R_0 = 0.2$, $\frac{h_{reg}}{h_{min}} = 2$ as the fastest and hybrids with $R_0 = 0.3$, $\frac{h_{reg}}{h_{min}} = 4$ and $R_0 = 0.4$, $\frac{h_{reg}}{h_{min}} = 4$ as middle ground. The results for error as a function of computational time are shown in Fig. 23, confirming the expected discretization rankings in both accuracy and execution time. All refined discretizations provide lower error than the regular and scattered constant density solutions at a comparable or lower computational time. In this case a hybrid discretization provides significant time reduction compared to a scattered discretization if we are somewhat willing to compromise on accuracy. Once the discretization is sufficiently refined, the regular part dominates the error and we would be better served by replacing LRBFCM with a more accurate, albeit expensive, approximation for the regular part of the hybrid.

5. Conclusions

In this paper we proposed an improvement in terms of computational efficiency for the numerical treatment of problems in which most of the domain can be discretized with regular nodes, while scattered nodes are used only near irregularly-shaped domain boundaries. First, we introduced an algorithm for n-dimensional h-refined meshless node placement that, based on user input, discretizes different regions either with scattered or with regular nodes. We showed that such an approach does not degrade the quality of generated nodes by means of analysing the separation distance and maximal empty sphere radius width of generated nodes.

The remainder of the paper is dedicated to demonstrating how the proposed hybrid regular-scattered discretization performs in different problems. We combined the regular nodes with LRBFCM, a fast but sensitive method, and scattered nodes with RBF-FD, expensive but robust method. With such a setup we solved the de Vahl Davis natural convection and Boussinesq's contact problems, in 2D and 3D. We showed that the proposed hybrid regular-scattered discretization can significantly contribute to the computational efficiency, while introducing minimal to no cost regarding the accuracy of the numerical solution.

Further analysis is required regarding the selection of an appropriate approximation method for the regular part to avoid the regular

part dominating the error as in some of the refined Boussinesq's cases. Additionally, the scattered node layer width and the aggressiveness of *h*-refinement near the irregularly shaped domain boundaries should be investigated, as both affect the computational efficiency and stability of the solution procedure. Future work should also include more difficult problems, such as mixed convection problems and a detailed analysis of possible surface effects, e.g. scattering, at the transition layer between the scattered and regular domains.

CRediT authorship contribution statement

Miha Rot: Conceptualization, Investigation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Mitja Jančič:** Investigation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Gregor Kosec:** Conceptualization, Funding acquisition, Project administration, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the financial support from the Slovenian Research and Innovation Agency (ARIS) research core funding No. P2-0095, Young Researcher programme PR-10468, and research projects No. J2-3048 and No. N2-0275.

Funded by National Science Centre, Poland under the OPUS call in the Weave programme 2021/43/I/ST3/00228. This research was funded in whole or in part by National Science Centre (2021/43/I/ST3/00228). For the purpose of Open Access, the author has applied a CC-BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

References

[1] H. Wendland, Scattered Data Approximation, vol. 17, Cambridge University Press, 2004.
[2] G.-R. Liu, Meshfree Methods: Moving Beyond the Finite Element Method, CRC Press, 2009.
[3] J. Slak, G. Kosec, On generation of node distributions for meshless PDE discretizations, SIAM J. Sci. Comput. 41 (5) (2019) A3202–A3229.
[4] V. Shankar, R.M. Kirby, A.L. Fogelson, Robust node generation for meshfree discretizations on irregular domains and surfaces, SIAM J. Sci. Comput. 40 (4) (2018) 2584–2608, http://dx.doi.org/10.1137/17m114090x.
[5] K. van der Sande, B. Fornberg, Fast variable density 3-D node generation, SIAM J. Sci. Comput. 43 (1) (2021) A242–A257.
[6] V. Bayona, N. Flyer, B. Fornberg, G.A. Barnett, On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs, J. Comput. Phys. 332 (2017) 257–273.
[7] A. Tolstykh, D. Shirobokov, On using radial basis functions in a "finite difference mode" with applications to elasticity problems, Comput. Mech. 33 (1) (2003) 68–79.
[8] J. Slak, G. Kosec, Adaptive radial basis function-generated finite differences method for contact problems, Internat. J. Numer. Methods Engrg. 119 (7) (2019) 661–686, http://dx.doi.org/10.1002/nme.6067.
[9] B. Fornberg, N. Flyer, A Primer on Radial Basis Functions with Applications to the Geosciences, SIAM, 2015.
[10] R. Zamolo, E. Nobile, Solution of incompressible fluid flow problems with heat transfer by means of an efficient RBF-FD meshless approach, Numer. Heat Transfer B 75 (1) (2019) 19–42.
[11] G. Kosec, A local numerical solution of a fluid-flow problem on an irregular domain, Adv. Eng. Softw. 120 (2018) 36–44.
[12] G. Kosec, B. Šarler, Solution of thermo-fluid problems by collocation with local pressure correction, Internat. J. Numer. Methods Heat Fluid Flow (2008).
[13] R. El Kadmiri, Y. Belaasilia, A. Timesli, M.S. Kadiri, A hybrid algorithm using the FEM-MESHLESS method to solve nonlinear structural problems, Eng. Anal. Bound. Elem. 140 (2022) 531–543, http://dx.doi.org/10.1016/j.enganabound.2022.04.018.
[14] J. Jaśkowiec, S. Milewski, Coupling finite element method with meshless finite difference method by means of approximation constraints, Comput. Math. Appl. 142 (2023) 208–224.
[15] H. Ding, C. Shu, K. Yeo, D. Xu, Simulation of incompressible viscous flows past a circular cylinder by hybrid FD scheme and meshless least square-based finite difference method, Comput. Methods Appl. Mech. Engrg. 193 (9–11) (2004) 727–744.
[16] G. Bourantas, K. Mountris, V. Loukopoulos, L. Lavier, G. Joldes, A. Wittek, K. Miller, Strong-form approach to elasticity: Hybrid finite difference-meshless collocation method (FDMCM), Appl. Math. Model. 57 (2018) 316–338, http://dx.doi.org/10.1016/j.apm.2017.09.028.
[17] G. Bourantas, L. Lavier, T. Van Dam, S. Bordas, Hybrid mesh/particle meshless method for modeling geological flows with discontinuous transport properties, in: EGU General Assembly Conference Abstracts, 2015, p. 11519.
[18] A. Javed, K. Djidjeli, J. Xing, S. Cox, A hybrid mesh free local RBF-Cartesian FD scheme for incompressible flow around solid bodies, Int. J. Math. Comput. Natl. Phys. Eng. 7 (2013) 957–966.
[19] M. Jančič, M. Rot, G. Kosec, Spatially-varying meshless approximation method for enhanced computational efficiency, in: J. Mikyška, C. de Mulatier, M. Paszynski, V.V. Krzhizhanovskaya, J.J. Dongarra, P.M. Sloot (Eds.), Computational Science – ICCS 2023, Springer Nature Switzerland, Cham, 2023, pp. 500–514.
[20] U. Duh, G. Kosec, J. Slak, Fast variable density node generation on parametric surfaces with application to mesh-free methods, SIAM J. Sci. Comput. 43 (2) (2021) A980–A1000, http://dx.doi.org/10.1137/20m1325642, Publisher: Society for Industrial & Applied Mathematics (SIAM).
[21] M. Jančič, J. Slak, G. Kosec, Monomial augmentation guidelines for rbf-fd from accuracy versus computational time perspective, J. Sci. Comput. 87 (1) (2021) 1–18.
[22] V. Bayona, An insight into RBF-FD approximations augmented with polynomials, Comput. Math. Appl. 77 (9) (2019) 2337–2353.
[23] N. Flyer, B. Fornberg, V. Bayona, G.A. Barnett, On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy, J. Comput. Phys. 321 (2016) 21–38.
[24] M. Jančič, G. Kosec, Strong form mesh-free hp-adaptive solution of linear elasticity problem, Eng. Comput. (2023) http://dx.doi.org/10.1007/s00366-023-01843-6, URL https://link.springer.com/10.1007/s00366-023-01843-6.
[25] G. Kosec, P. Zinterhof, Local strong form meshless method on multiple Graphics Processing Units, 2013, p. 20.
[26] R. Vertnik, B. Šarler, et al., Local radial basis function collocation method along with explicit time stepping for hyperbolic partial differential equations, Appl. Numer. Math. 67 (2013) 136–151.
[27] H. Ding, C. Shu, D. Tang, Error estimates of local multiquadric-based differential quadrature (LMQDQ) method through numerical experiments, Int. J. Numer. Methods Eng. 63 (11) (2005) 1513–1529.
[28] V. Bayona, M. Moscoso, M. Carretero, M. Kindelan, RBF-FD formulas and convergence properties, J. Comput. Phys. 229 (22) (2010) 8281–8295, http://dx.doi.org/10.1016/j.jcp.2010.07.008, URL https://linkinghub.elsevier.com/retrieve/pii/S0021999110003773.
[29] J. Slak, G. Kosec, Refined Meshless Local Strong Form solution of Cauchy–Navier equation on an irregular domain, Eng. Anal. Bound. Elem. 100 (2019) 3–13, http://dx.doi.org/10.1016/j.enganabound.2018.01.001.
[30] T. Fusegi, J.M. Hyun, K. Kuwahara, B. Farouk, A numerical study of three-dimensional natural convection in a differentially heated cubical enclosure, Int. J. Heat Mass Transfer 34 (6) (1991) 1543–1557.
[31] D.J. Tritton, Physical Fluid Dynamics, Oxford Science Publ, Clarendon Press, 1988, http://dx.doi.org/10.1007/978-94-009-9992-3.
[32] A.J. Chorin, Numerical solution of the Navier-Stokes equations, Math. Comput. 22 (104) (1968) 745–762.
[33] G. de Vahl Davis, Natural convection of air in a square cavity: a bench mark numerical solution, Int. J. Numer. Methods Fluids 3 (3) (1983) 249–264.
[34] H. Sadat, S. Couturier, Performance and accuracy of a meshless method for laminar natural convection, Numer. Heat Transfer B 37 (4) (2000) 455–467, http://dx.doi.org/10.1080/10407790050051146.
[35] G.W.W. D. C. Wan, A new benchmark quality solution for the buoyancy-driven cavity by discrete singular convolution, Numer. Heat Transfer B 40 (3) (2001) 199–228, http://dx.doi.org/10.1080/104077901752379620.
[36] P. Wang, Y. Zhang, Z. Guo, Numerical study of three-dimensional natural convection in a cubical cavity at high Rayleigh numbers, Int. J. Heat Mass Transfer 113 (2017) 217–228.
[37] W.S. Slaughter, The Linearized Theory of Elasticity, Springer Science & Business Media, 2012.