Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Accurate RBF-FD meshless numerical simulation of thermo-fluid problems for generic 3D geometries

To cite this article: Riccardo Zamolo *et al* 2023 *J. Phys.: Conf. Ser.* **2509** 012002

View the article online for updates and enhancements.

You may also like

- Phase field modelling of dendritic solidification by using an adaptive meshless solution procedure
T Dobravec, B Mavri and B Šarler
- Constructing the weighting coefficients for the RBF-Hermite FD scheme under the multiquadric function on irregular meshes
Haifa Bin Jebreen and Fairouz Tchier
- A meshless quasicontinuum method based on local maximum-entropy interpolation
Dennis M Kochmann and Gabriela N Venturini

This content was downloaded from IP address 79.41.233.64 on 08/09/2023 at 11:43

Accurate RBF-FD meshless numerical simulation of thermo-fluid problems for generic 3D geometries

Riccardo Zamolo, Davide Miotti and Enrico Nobile

Dipartimento di Ingegneria e Architettura, Università degli Studi di Trieste, via A. Valerio 10, 34127 Trieste, Italy

E-mail: rzamolo@units.it

Abstract. In contrast to traditional mesh-based methods for the numerical solution of boundary value problems, e.g., Finite Element (FEM) and Finite Volume (FVM), in the recent period many meshfree approaches have been proposed in order to avoid those typical issues due to the mesh. For example, the quality of the mesh greatly affects the reliability of the final solution in the case of CFD problems and the human intervention of a professional is often still needed when dealing with complex-shaped domains. This in turn increases both cost and time required for the reliable simulation of problems of engineering relevance. Meshless methods, on the other side, usually rely on a simpler distribution of nodes and do not require the storage of connectivity information. Among others, one of the most promising meshless methods in terms of accuracy and flexibility is the one based on the Radial Basis Function – Finite Difference (RBF-FD) scheme. RBF-FD methods, however, are usually affected by severe ill conditioning issues when Neumann boundary conditions are employed. This fact is the main responsible for the appearance of large discretization errors near the boundary and for the lack of stability of traditional time integration schemes. In order to address this issue, some new algorithms for the robust treatment of boundary conditions have been developed and successfully employed to solve fluid flow problems with heat transfer. Furthermore, it is well acknowledged that the efficient resolution of boundary layers arising in this class of problems requires an adequate spatial discretization in the neighbourhood of the boundary, i.e., increased node/mesh density along the direction of large gradients only. This result is achieved by employing anisotropic node distributions, which is a novelty in the context of the RBF-FD method to the best of the authors' knowledge. The method described above is successfully employed for the accurate solution of a representative 3D heat transfer problem with incompressible fluid flow.

1. Introduction

Meshless methods applied to Computational Fluid Dynamics (CFD) problems are at the center of an increasingly animated activity of research. The reason for such an interest may be looked for in the practical shortcomings of traditional mesh-based methods adopted in this field, namely Finite Element (FEM) [1] and Finite Volume (FVM) [2] methods.

Indeed, the generation of a mesh over complex 3D geometries is a very delicate process which often requires human intervention by an adequately experienced professional. If done improperly, either the final results end up being unreliable, for example when the mesh is not appropriately refined where required, or the computational cost becomes unreasonable, such as when the refinement is excessive. Automatic mesh generation is not always possible, especially in CFD where proper refinement of the mesh is needed at specific locations of the domain in order to capture local phenomena with adequate accuracy, e.g., boundary layers.


Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

On top of that, even when the mesh is generated properly, the resulting data structure can not be flexible enough to allow large geometric deformations of the domain without any remeshing. This can happen, for instance, when the solver is paired with some optimization algorithm and the geometry needs to be deformed several times.

When meshless methods are used, the mesh generation is avoided and usually replaced by a simple set of nodes which are properly distributed over the computational domain. Afterwards, an interpolation scheme is employed to derive the discrete form of the governing equations of the problem of interest. In our case the Radial Basis Function-Finite Difference (RBF-FD) scheme is adopted [3–8]. Its main advantage lies in the fact that there is no need to store explicitly the usual geometric connectivity information required by a mesh, e.g., those quantities related to faces, elements, etc. This theoretically permits greater freedom in the discretization procedure, and also in the case of node movements during the early stage of node generation or during the simulation, in order to account for domain deformations. The solution is computed only at the node locations and therefore, in order to achieve an adequate accuracy, a corresponding adequate node density must be employed throughout the whole domain and especially near the boundaries, where additional requirements may be necessary to guarantee stable schemes.

In this paper, an implementation of the RBF-FD meshless method is employed to accurately solve a steady-state, incompressible fluid flow problem with heat transfer on a simplified 3D domain. The feasibility of using anisotropic node distributions, i.e., in which the spacing between the nodes depends upon the direction, is investigated in order to assess the capability of the RBF-FD method to face practical CFD problems characterized by large gradients at the boundary layers. Finally, the results are compared with the ones obtained using a commercial FVM-based solver, highlighting an excellent degree of accuracy.

The required computations for the numerical simulation of a CFD problem, i.e., discretization of the domain, discretization of the governing equations and solution of the resulting algebraic equations, are all performed using the Julia programming language [9], which allows extensive code reuse and excellent computational performance already at the development stage.

2. Governing equations

The thermo-fluid problem of interest is described by coupled conservation equations of mass, momentum and energy. An incompressible fluid of density ρ, kinematic viscosity ν, thermal diffusivity α, thermal conductivity k and volumetric temperature expansion coefficient β is considered. The Boussinesq approximation is employed to model the buoyancy effects. The aforementioned equations thus take the following nondimensional form:

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \frac{1}{\sqrt{\mathrm{Gr}_d}} \nabla^2 \boldsymbol{u} + \hat{\boldsymbol{z}} T, \tag{2}$$

$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \frac{1}{\mathrm{Pr}\sqrt{\mathrm{Gr}_d}} \nabla^2 T, \tag{3}$$

where $\hat{\boldsymbol{z}}$ is the unit vector along the vertical direction z. In the above equations, length, velocity $\boldsymbol{u} = (u, v, w)$, time t, pressure p and temperature T are made dimensionless by taking respectively d, $U_0 = \sqrt{g\beta d \Delta T}$, d/U_0, ρU_0^2 and $\Delta T = T_h - T_c$ as reference quantities (see §4), where g is the gravitational acceleration. $\mathrm{Gr}_d = g\beta d^3 \Delta T / \nu^2$ is the Grashof number, $\mathrm{Pr} = \nu/\alpha$ is the Prandtl number and $\mathrm{Ra}_d = \mathrm{Gr}_d \cdot \mathrm{Pr}$ is the Rayleigh number. For the presented computations we consider $\mathrm{Pr} = 0.71$ and three different values $\mathrm{Ra}_d = 1,000$, $5,000$ and $10,000$, leading to steady-state solutions for the presented geometry. Slightly larger values of Ra_d, e.g., above $\mathrm{Ra}_d = 15,000 \div 20,000$, lead to periodic solutions.

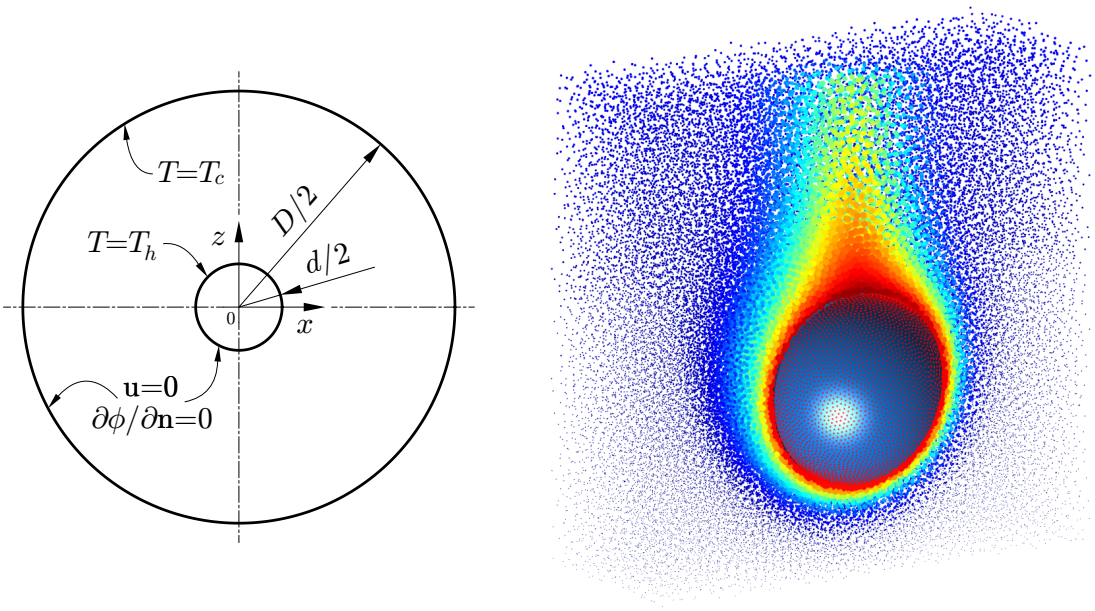

Figure 1. $x - z$ section of the computational domain and adopted boundary conditions (left) and an example of a meshless temperature distribution over the internal nodes, i.e., nodes inside the computational domain (right, 320k nodes, $\mathrm{Ra}_d = 10^4$).

3. Numerical method

Here follows a brief description of the adopted RBF-FD method. An interested reader might refer to more extensive descriptions reported in [3–8].

The whole simulation process is composed by three phases, namely node generation, RBF-FD discretization and final solution, which are briefly described as follows.

3.1. Node generation

Node distributions are represented by a certain number N of points fulfilling a prescribed spacing function $s(\boldsymbol{x})$ scattered within the domain Ω and on its boundary $\partial\Omega$.

Node generation can be further divided in two stages:

- generation of a volumetric node distribution within Ω that satisfies a certain spacing s on average; $\partial\Omega$ is not taken into account yet,
- iterative refinement of the initial node placement through a node-repel approach until a sought node distribution is reached.

During the second stage nodes are moved according to mutual repulsion forces that arise between neighbors. Whenever one of them is pushed outside Ω, it is projected onto the nearest point on $\partial\Omega$. An example of 3D node distribution for the domain described in §4 is depicted in Figure 1 where the quality of the boundary distribution on the sphere due to the iterative refinement process can be appreciated. If the domain is defined implicitly or explicitly by means of analytic functions, the implementation of the projection is straightforward. In the generic case where the domain is defined by a CAD geometry, e.g., by means of a `.stl` (stereolithography) file, the projection operation can be efficiently performed by exploiting an octree data structure for the boundary surface [10–12].

Isotropic node distributions are obtained when the mutual repulsion forces that arise between neighbor nodes during the iterative refinement are equally weighted regardless of their direction.

Smoothness	Type	Definition $\Phi(r)$	Positive definite, order
Infinitely	Gaussian (GA)	$e^{-(\varepsilon\, r)^2}$	strictly
	Multiquadric (MQ)	$\sqrt{1+(\varepsilon\, r)^2}$	conditionally, 1
	Inverse Multiquadric (IMQ)	$1/\sqrt{1+(\varepsilon\, r)^2}$	strictly
	Inverse Quadratic (IQ)	$1/(1+(\varepsilon\, r)^2)$	strictly
Piecewise	Monomial PHS	r^{2k+1}, $k \in \mathbb{N}$	conditionally, $k+1$
	Thin Plate Spline PHS	$r^{2k}\log r$, $k \in \mathbb{N}$	conditionally, $k+1$

Table 1. Most common Radial Basis Functions.

If the component of the repulsion force along a particular direction $\boldsymbol{e}_{an}$ has a different weighting than the orthogonal directions, the resulting node distribution will be anisotropic. Such anisotropy can then be described by two quantities: the anisotropic direction $\boldsymbol{e}_{an}(\boldsymbol{x})$ and the anisotropic factor $\kappa(\boldsymbol{x}) \geq 1$, both functions of the position $\boldsymbol{x}$. The corresponding linear transformations, which are mutually inverse, are simply given by:

$$\boldsymbol{v} \rightarrow \boldsymbol{v} + (\kappa - 1)(\boldsymbol{v} \cdot \boldsymbol{e}_{an})\boldsymbol{e}_{an} \tag{4}$$

$$\boldsymbol{w} \rightarrow \boldsymbol{w} - (1 - \kappa^{-1})(\boldsymbol{w} \cdot \boldsymbol{e}_{an})\boldsymbol{e}_{an} \tag{5}$$

representing, respectively, an expansion and a contraction of the space along $\boldsymbol{e}_{an}$ by a factor $\kappa \geq 1$. The effect of this anisotropy on the node distribution is that the node spacing along $\boldsymbol{e}_{an}$ will be $s(\boldsymbol{x})/\kappa(\boldsymbol{x})$, i.e., more nodes along $\boldsymbol{e}_{an}$, while the spacing along the orthogonal directions will be unchanged, i.e., $s(\boldsymbol{x})$. The resulting node density, i.e., nodes per unit volume, is therefore:

$$\delta(\boldsymbol{x}) = \frac{2\kappa(\boldsymbol{x})}{\zeta s^D(\boldsymbol{x})} \tag{6}$$

where $\zeta = \sqrt{5 - D}$ in D dimensions.

In order to enhance the stability of the RBF-FD scheme employed to solve the time dependent equations (1)-(3), after the iterative refinement phase a single layer of internal nodes is added from the boundary nodes in the opposite direction to the outer normals. Such strategy has a beneficial effect when dealing with Neumann boundary conditions for the auxiliary variable ϕ, equation (19).

3.2. RBF-FD collocation method

The starting point in the RBF-FD method is the construction of a suitable function that interpolates nodal values. Such interpolant is a linear combination of m Radial Basis Functions (RBFs) $\Phi(||\boldsymbol{x} - \boldsymbol{x}_i||_2)$, and the set $\{\boldsymbol{x}_i\}_1^m$ is the interpolation stencil. The most commonly used RBFs are reported in Table 1, where they are defined in terms of $r = ||\boldsymbol{x} - \boldsymbol{x}_i||_2$, for $\boldsymbol{x}_i$ fixed. In this paper the Multiquadric (MQ) radial basis function is adopted. In order to improve accuracy [13], a polynomial basis $\{p_j\}_1^q$ of the multivariate polynomial space Π_P^d of total degree P in d dimensions is also included in the RBF expansion as follows:

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \varphi_i(\boldsymbol{x}) + \sum_{j=1}^{q} \beta_j p_j(\boldsymbol{x}) \tag{7}$$

where the notation $\varphi_i(\boldsymbol{x}) = \Phi(||\boldsymbol{x} - \boldsymbol{x}_i||_2)$ is used.

The dependence of the weights $\{\alpha_1, \ldots, \alpha_m\}$ and $\{\beta_1, \ldots, \beta_q\}$ on the nodal values is obtained by considering the following interpolation conditions:

$$u^h(\boldsymbol{x}_i) = u(\boldsymbol{x}_i), \quad i = 1, \ldots, m \tag{8}$$

When boundary nodes are included in the stencil, the corresponding interpolation conditions of equation (8) are replaced by the respective boundary conditions. In the case of Robin b.c., i.e., $au + b\,\partial u/\partial \mathbf{n} = g$, applied to the i^{th} node of the stencil, the corresponding equation becomes:

$$a(\boldsymbol{x}_i)\, u^h(\boldsymbol{x}_i) + b(\boldsymbol{x}_i)\, \frac{\partial u^h}{\partial \mathbf{n}}(\boldsymbol{x}_i) = g(\boldsymbol{x}_i), \quad \boldsymbol{x}_i \in \partial\Omega \tag{9}$$

where a, b, g are all known functions defined on the boundary.

In order to make the system solvable, other q equations are attained by enforcing the following additional orthogonality conditions:

$$\sum_{i=1}^{m} \alpha_i p_j(\boldsymbol{x}_i) = 0, \quad j = 1, \ldots, q \tag{10}$$

By substituting equation (7) into (8) and (9), the unknown values of α_i and β_j can be calculated by solving the following linear system:

$$\underbrace{\begin{bmatrix} \boldsymbol{\varphi}_{BC} & \mathbf{P}_{BC} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}}_{\mathbf{M}} \begin{Bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{u} \\ \boldsymbol{g} \\ \mathbf{0} \end{Bmatrix} \tag{11}$$

where the block matrix $\boldsymbol{P}^T$ comes from equation (10), $\{\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_q\}$ is the vector of unknowns and both $\boldsymbol{\varphi_{BC}}$ and $\boldsymbol{P_{BC}}$ come from equations (8) and (9).

Given a linear partial differential equation $\mathcal{L}(u) = f$ in the unknown field u, the RBF expansion (7) is made valid at each node $\boldsymbol{x}_i$ which does not lie on the boundary. The following sparse linear system is thus obtained:

$$\mathbf{C}\left\{\boldsymbol{u}^h\right\} = \boldsymbol{q} - \boldsymbol{f} \tag{12}$$

where $\boldsymbol{u}^h = \{u^h(\boldsymbol{x}_1), \ldots, u^h(\boldsymbol{x}_{N_I})\}$ is the vector of unknown scalar values $u^h(\boldsymbol{x}_i)$ evaluated at all N_I inner nodes, i.e. those contained in Ω, while the vector $\boldsymbol{q}$ comes from the enforcement of boundary conditions at the interpolant level. Each row $\mathbf{c}_i$ of the matrix $\mathbf{C}$ therefore correspond to the inner node at $\boldsymbol{x}_i$ and is attained through the solution of the local linear system (13), associated to the stencil built around that node.

$$\mathbf{M}^T\, \mathbf{c}_i = \begin{Bmatrix} \boldsymbol{\Psi}(\boldsymbol{x}_i) \\ \boldsymbol{\Pi}(\boldsymbol{x}_i) \end{Bmatrix} \tag{13}$$

where $\boldsymbol{\Psi}(\boldsymbol{x}_i) = \{\mathcal{L}\varphi_i(\boldsymbol{x}_i) \ldots \mathcal{L}\varphi_m(\boldsymbol{x}_i)\}$ and $\boldsymbol{\Pi}(\boldsymbol{x}) = \{\mathcal{L}p_1(\boldsymbol{x}_i) \ldots \mathcal{L}p_q(\boldsymbol{x}_i)\}$ are column vectors and $\mathbf{M}$ is the same as in equation (11).

3.3. Solution procedure

The computation of velocity, pressure and temperature through equations (1)-(3) is decoupled at each time step by using a projection scheme with a three-level Gear scheme for the

time discretization. A tentative velocity $\boldsymbol{u}^*$ is computed from the linearized nondimensional momentum equation:

$$\frac{3\boldsymbol{u}^* - 4\boldsymbol{u}^l + \boldsymbol{u}^{l-1}}{2\Delta t} + \boldsymbol{u}^l \nabla \boldsymbol{u}^* = -\nabla p^l + \frac{1}{\sqrt{\mathrm{Gr}_D}} \nabla^2 \boldsymbol{u}^* + \hat{\boldsymbol{z}} T^l, \tag{14}$$

where l is the time level and Δt is the nondimensional time step size, which is chosen to be $\Delta t = 0.25$ for the presented results.

The tentative velocity $\boldsymbol{u}^*$ is then forced to satisfy the continuity equation (1) by means of an irrotational correction $\boldsymbol{u}^{l+1} = \boldsymbol{u}^* - \nabla \phi$, leading to the following Poisson equation in the auxiliary variable ϕ:

$$\nabla^2 \phi = \nabla \cdot \boldsymbol{u}^* \tag{15}$$

The pressure is then updated as $p^{l+1} = p^l + \phi / \Delta t$ and the temperature is computed from the discretized nondimensional energy equation:

$$\frac{3T^{l+1} - 4T^l + T^{l-1}}{2\Delta t} + \boldsymbol{u}^{l+1} \nabla T^{l+1} = \frac{1}{\mathrm{Pr}\sqrt{\mathrm{Gr}_D}} \nabla^2 T^{l+1}. \tag{16}$$

The previous equations are discretized in space with the RBF-FD scheme presented in §3.2. Explicit hyperviscosity ∇^4 [14] is employed to stabilize the time evolution of the variables $\boldsymbol{u}$, p and T. Sparse linear systems like the one in Equation (12) are attained for each component of velocity, for the temperature and for the auxiliary variable ϕ. Such sparse linear systems are preconditioned with an Incomplete LU factorization (ILU) [15] (package `IncompleteLU` in `Julia`). In the case of the Poisson equation for the velocity correction, equation (15), the ILU factorization can be performed only once at the beginning of the simulation. The Biconjugate Gradient Stabilized Method [16] (package `IterativeSolvers` in `Julia`) is then employed as iterative solver using a relative tolerance of 10^{-10}. The computational time required for each time step ranges from 2 seconds for $N \approx 100,000$ nodes to approximately 9 seconds for $N \approx 500,000$ nodes on a laptop equipped with a quad-core Intel i7 2.6GHz processor.

3.4. Convergence criterion

The convergence to steady-state through the time stepping procedure explained in §3.3 is declared when the average of the root-mean-square (RMS) time derivatives for the velocity components and temperature become less than $tol = 5 \cdot 10^{-6}$. The RMS time derivative $\dot{f}_\Omega$ for a generic field f over the domain Ω is defined as:

$$\dot{f}_\Omega = \sqrt{\frac{1}{\mu(\Omega)} \int_\Omega \left(\frac{\partial f}{\partial t} \right)^2 d\Omega} \approx \sqrt{\frac{1}{\sum_i s_i^3} \sum_i \left(\frac{f_i^{l+1} - f_i^l}{\Delta t} \right)^2 s_i^3} \tag{17}$$

where $\mu(\Omega)$ is the measure of the domain Ω and the notation $f_i = f(\mathbf{x}_i)$ for $\mathbf{x}_i \in \Omega$ is used. Therefore the convergence criterion can be explicitly written as:

$$\sqrt{\frac{\dot{u}_\Omega^2 + \dot{v}_\Omega^2 + \dot{w}_\Omega^2 + \dot{T}_\Omega^2}{4}} < tol \tag{18}$$

4. Geometry, boundary conditions and generic definitions

A 3D sphere of diameter d, assumed as the reference length, is enclosed in a spherical cavity with diameter $D = 5d$, i.e, a spherical shell. Both spheres share the same center point, which is

also the origin of the considered cartesian reference frame, as shown in Figure 1. The following boundary conditions are enforced:

$$\begin{cases} \text{no-slip condition } \mathbf{u} = \mathbf{0} \text{ on all boundaries,} \\ T = T_h \text{ on the surface of the internal sphere of diameter } d, \\ T = T_c \text{ on the surface of the external sphere of diameter } D, \\ \dfrac{\partial \phi}{\partial \mathbf{n}} = 0 \text{ on all boundaries.} \end{cases} \tag{19}$$

The last condition in (19) is due to the fact the domain is a cavity, i.e., it has no inlets or outlets. Therefore equation (15) must satisfy the compatibility condition $\int_\Omega \nabla \cdot \mathbf{u}^* = 0$ in order to be solvable. At the PDE level such condition is always met in a cavity because of the divergence theorem. However, when considering the discretized version of the Poisson equation (15), i.e., the linear system $\mathbf{A}\boldsymbol{\phi} = \boldsymbol{b}$, the compatibility condition is:

$$\boldsymbol{w}^T \boldsymbol{b} = 0 \tag{20}$$

where $\boldsymbol{w}$ is any vector of the 1D nullspace of $\mathbf{A}^T$, i.e., $\mathbf{A}^T \boldsymbol{w} = \mathbf{0}$. The compatibility condition (20) is not generally met when using non-conservative schemes like the bare RBF-FD method. In order to meet such condition, each entry of the right hand side vector $\boldsymbol{b}$ of the discretized Poisson equation can be corrected by subtracting the value $\boldsymbol{w}^T \boldsymbol{b} / \sum w_i$.

The mean Nusselt number over the surface Γ of the internal sphere is given by:

$$\overline{\mathrm{Nu}}_d = \frac{d}{\Delta T \cdot \mu(\Gamma)} \int_\Gamma \frac{\partial T}{\partial \mathbf{n}} d\Gamma \;\approx\; \frac{d}{\Delta T \sum_b s_b^2} \sum_b \frac{\partial T}{\partial \mathbf{n}}\Big|_b s_b^2 \tag{21}$$

where the index b runs over each boundary node on Γ.

The spacing function $s(\boldsymbol{x})$ and the anisotropic factor $k(\boldsymbol{x})$ are defined as follows:

$$s(\boldsymbol{x}) = s_{min} + (s_{max} - s_{min}) \Delta r \frac{1 + \lambda_s}{1 + \lambda_s \Delta r} \tag{22}$$

$$\kappa(\boldsymbol{x}) = \kappa_{max} - (\kappa_{max} - 1) \Delta r^2 \frac{1 + \lambda_\kappa^2}{1 + \lambda_\kappa^2 \Delta r^2} \tag{23}$$

where s_{min} and s_{max} are the minimum and the maximum values of the spacing function, respectively, and κ_{max} is the maximum value of the anisotropic factor. $\Delta r = (2r - d)/(D - d)$ where r is the radius from the center of the spheres and the following parameters are employed: $s_{max}/s_{min} = 3$, $\lambda_s = 3$ and $\lambda_k = 10$. These definitions are such as to obtain node distributions with a small spacing s_{min} at the surface of the internal sphere and a large spacing s_{max} at the surface of the external one. The anisotropic factor grows from $\kappa = 1$ at the external sphere to $\kappa = \kappa_{max}$ at the internal one while the anisotropic direction $\boldsymbol{e}_{an}$ is chosen to be radial, i.e., it points towards the center of the spheres.

5. Results

5.1. Preliminary considerations on a 2D Poisson equation

In order to highlight the advantages and the basic feasibility of the use of an anisotropic node distribution in the context of the RBF-FD method, let us consider the following 2D Poisson equation:

$$\nabla^2 u = b \tag{24}$$

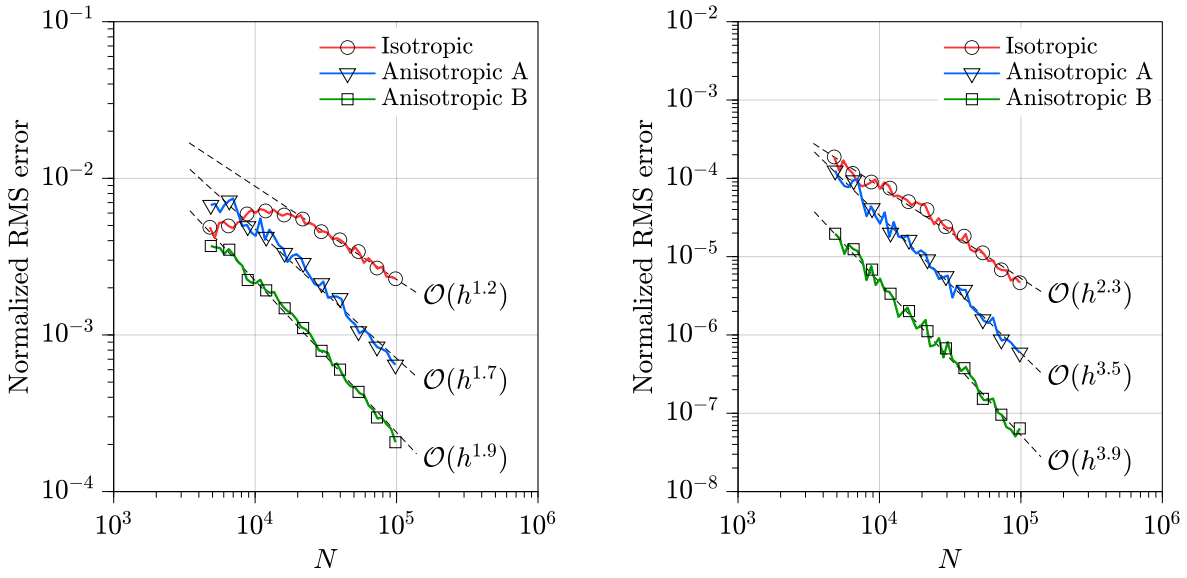

Figure 2. Convergence curves for a 2D Poisson equation in the case of a solution with large gradients, solved by RBF-FD with polynomial degree $P = 2$ (left) and $P = 4$ (right) and by using isotropic and non-isotropic node distributions.

with Dirichlet b.c. on the unit square $[0, 1]^2$. The chosen analytical solution is $u = e^{-50x}$, which has a very large gradient at $x = 0$ which requires a localized increase in the node density in order to be accurately resolved, for example by using the following spacing function:

$$s(\boldsymbol{x}) = s_{min} + (s_{max} - s_{min})\frac{1 - e^{-3x}}{1 - e^{-3}} \tag{25}$$

where $s_{max} = 3s_{min}$ is chosen. The resulting node distribution is isotropic and more refined near $x = 0$, although the increase in the node density along y is unnecessary since the large gradient of u is along x whereas u is constant along y. Therefore it would be much more effective to employ an anisotropic node distribution with increased node density only along x, i.e., with a spacing along x defined by equation (25), $s_x = s(\boldsymbol{x})$, and a constant spacing along y, $s_y = s_{max}$. The convergence curves for these two approaches are shown in Figure 2 for polynomial degrees $P = 2$ and $P = 4$ and $n = 2q$ local nodes. In strategy A the space for the interpolant is scaled with shape factors $\varepsilon_x = 0.5/s_x$ along x and $\varepsilon_y = 0.5/s_y$ along y, i.e., the interpolant "sees" isotropic nodes, while in strategy B the interpolant is not scaled, i.e., $\varepsilon = 0.5/s_y$ as usual. From Figure 2 we see that the anisotropic node distributions provide a significant increase in accuracy over the isotropic distributions. We can also see that strategy B seems to be the best in terms of both rate of convergence (1.9 for $P = 2$ and 3.9 for $P = 4$, as expected) and absolute error. This is due to the fact that the scaling employed in strategy A is equivalent to an increase in the shape factor ε, which is known to affect the accuracy of the RBF interpolation.

This example shows that, as expected, the most effective way to accurately capture a rapidly varying solution is to increase the node density only in the direction of the largest gradient.

5.2. Natural convection in a spherical shell

In order to highlight the properties of the employed RBF-FD approach for the solution of equations (1)-(3), several simulations are carried out for different node distributions using a total number of nodes ranging from $N \approx 100,000$ nodes to $N \approx 500,000$ nodes. Polynomial degree $P = 3$ is employed with $n = 2q = 40$ local nodes and $\varepsilon = 0.5/s(\boldsymbol{x})$.

Table 2. Mean Nusselt number.

		$\overline{\mathrm{Nu}}_d$		
	N	$\mathrm{Ra}_d = 10^3$	$\mathrm{Ra}_d = 5 \cdot 10^3$	$\mathrm{Ra}_d = 10^4$
Isotropic	$100k$	4.0336	5.3949	6.1342
	$130k$	4.0340	5.4000	6.1352
	$165k$	4.0321	5.3879	6.1198
	$320k$	4.0393	5.3958	6.1262
	$455k$	4.0399	5.3964	6.1262
Anisotropic	$120k$	4.0300	5.3858	6.1144
	$150k$	4.0324	5.3896	6.1176
	$190k$	4.0366	5.3938	6.1245
	$360k$	4.0361	5.3921	6.1223
	$515k$	4.0351	5.3905	6.1204
	Fluent	4.0415	5.3994	6.1300

The results are summarized in Table 2, where the computed mean Nusselt number $\overline{\mathrm{Nu}}_d$ is reported for the three chosen values of the Rayleigh number $\mathrm{Ra}_d = 1,000,\ 5,000$ and $10,000$, and for the two strategies employing isotropic and anisotropic node distributions. In the last case, the anisotropic factor is chosen to be $\kappa_{max} = 2$ for which the interpolation scaling is chosen to be halfway between strategies A and B, briefly presented in §5.1. This choice is due to the fact that strategy B turns out to be the most unstable, despite it seems to be the most accurate.

Table 2 shows also the values of $\overline{\mathrm{Nu}}_d$ computed with Ansys Fluent 2022R1 as reference, where the axial symmetry of the problem is exploited in order to provide a reliable and accurate solution. From the comparison of the values of this table we can observe a very strong agreement between the presented results and the reference ones for each of the three values of Ra_d, even for the smallest number of nodes, i.e., $N \approx 100k$ nodes, with errors always much lower than 1%.

From these values it can also be observed that the anisotropic results are slightly worse than the isotropic ones, when compared to the reference values computed by Fluent. This can be due to the interpolant scaling strategy which is not the optimal one, i.e., strategy B presented in §5.1. In other words, the nodal spacing is reduced anisotropically along the radial direction $\boldsymbol{e}_{an}$ when approaching the surface of the internal sphere, while at the same time the shape factor for the RBF interpolant increases along the same direction. This last effect seems to be predominant in the presented cases, where the gradients of the field variables in the boundary layers are such as to be sufficiently resolved without the need of anisotropic distributions.

Nonetheless, these results confirm that the RBF-FD approach can operate with consistency also with anisotropic node distributions, and therefore it can be employed to face practical engineering problems at large values of Rayleigh or Reynolds numbers. The consistency of the presented anisotropic RBF-FD approach is also confirmed by the fact that, in the presented cases, the computed velocity and temperature profiles, like the ones presented in Figure 3, are indistinguishable from the isotropic ones.

Figure 3 shows different comparisons between the profiles of some variables of interest, obtained with an isotropic distribution and $N \approx 320k$ node, for each of the considered values of Ra_d. Reference values obtained with Fluent are also shown for comparison. Figures 3a and b show, respectively, the temperature T and the vertical velocity w along the horizontal line $y = z = 0$. Figure 3c shows the local Nusselt number Nu_d at the surface of the internal sphere as a function of the angle α taken from the horizontal line $y = z = 0$.

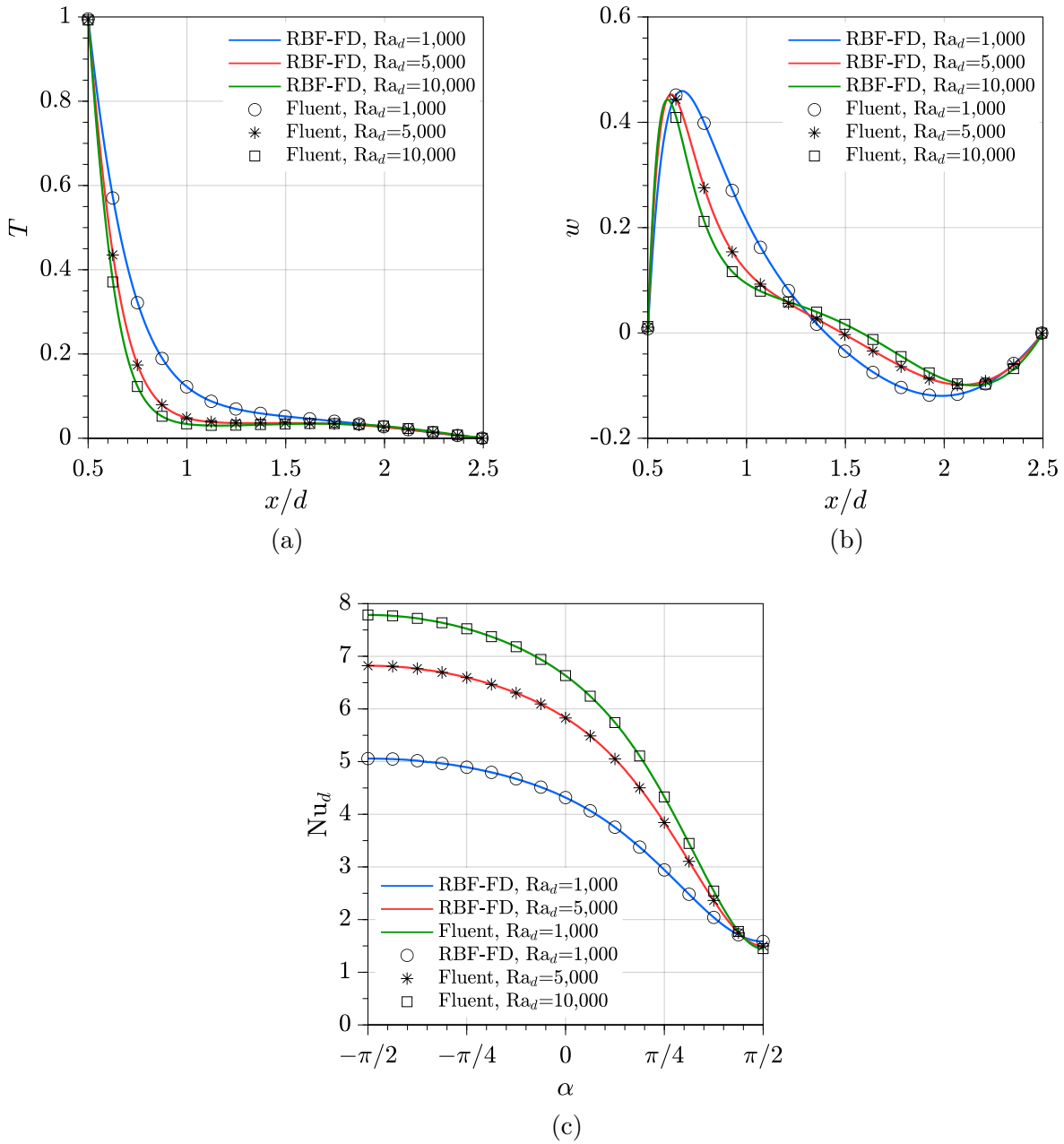

Figure 3. Comparison of temperature T (a) and vertical z-velocity w (b) profiles along the horizontal line $y = z = 0$. Comparison of local Nusselt number Nu_d (c) on the internal sphere.

Again, all the previous graphical comparisons show an excellent agreement between the computed profiles and the reference ones.

6. Conclusions

In this work the RBF-FD meshless method is applied to the simulation of a natural convection problem with laminar flow in a spherical shell at Rayleigh numbers $\mathrm{Ra}_d = 1,000$, $5,000$ and $10,000$. Excellent agreements with reference results are consistently obtained. Particular attention was reserved to the development and testing of a new method for attaining anisotropic

node distributions. As a matter of fact, while node generation algorithms and meshless methods in general are theoretically capable of coping with very complex shapes, isotropic node placements might lead to unnecessary node concentrations at certain spots of the domain, leading to a waste of computational resources. The adoption of anisotropic node distributions addresses this issue by increasing the node density only along a single meaningful direction in order to provide an efficient resolution of large gradients, e.g., boundary layers.

Future developments will insist at perfecting the algorithms for the optimal node placement and for stable anisotropic RBF approaches. For instance, the natural evolution of the anisotropic node generation lies in its integration within an adaptive node refinement algorithm. Indeed, it appears clear that, in order to increase both convenience and accuracy of collocation meshless methods, a lot of effort needs to be invested in the development of efficient and totally autonomous node generation algorithms.

The attained results suggest that the RBF-FD method with isotropic and even anisotropic node distributions has the potential to provide both high accuracy and ease of use for the numerical solution of generic heat transfer problems of engineering interest, and therefore represents an ideal candidate for the emerging needs of the industry.

References

[1] Zienkiewicz O C 1977 *The finite element method* vol 3 (McGraw-hill London)
[2] Versteeg H K and Malalasekera W 2007 *An introduction to computational fluid dynamics: the finite volume method* (Pearson education)
[3] Fornberg B and Flyer N 2015 *A primer on radial basis functions with applications to the geosciences* (SIAM)
[4] Divo E and Kassab A J 2007 *J HEAT TRANSF* **129** 124–136
[5] Zamolo R and Nobile E 2019 *NUMER HEAT TR B-FUND* **75** 19–42
[6] Šarler B and Vertnik R 2006 *COMPUT MATH APPL* **51** 1269–1282
[7] Waters J and Pepper D W 2015 *NUMER HEAT TR B-FUND* **68** 185–203
[8] Kosec G and Slak J 2020 *AIP Conference Proceedings* **2293** 420094
[9] Bezanson J, Edelman A, Karpinski S and Shah V B 2017 *SIAM REV* **59** 65–98
[10] Frey P and George P 2013 *Mesh Generation: Application to Finite Elements* ISTE (Wiley) ISBN 9781118623824
[11] de Berg M, Cheong O, van Kreveld M and Overmars M 2008 *Computational Geometry: Algorithms and Applications* (Springer Berlin Heidelberg) ISBN 9783540779742
[12] Zamolo R and Nobile E 2021 *J PHYS CONF SER* **2116**
[13] Bayona V, Flyer N, Fornberg B and Barnett G A 2017 *J COMPUT PHYS* **332** 257–273
[14] Fornberg B and Lehto E 2011 *J COMPUT PHYS* **230** 2270–2285
[15] Saad Y 2003 *Iterative Methods for Sparse Linear Systems* (Philadelphia, Pennsylvania: SIAM) chap 10: Preconditioning Techniques, pp 297–368 2nd ed
[16] van der Vorst H 1992 *SIAM J SCI COMPUT* **13**(2) 631–644