INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Radial Basis Function Based Partition of Unity Method for Two-Dimensional Unsteady Convection Diffusion Equations

G Krishna Veni[1]*, Chirala Satyanarayana[2], M Chenna Krishnareddy[1]

1 Department of Mathematics, Osmania University College for Women, Hyderabad
2 Department of Mathematics, Ecole Centrale School of Engineering, Mahindra University, Hyderabad

OPEN ACCESS

Received: 09-06-2023
Accepted: 17-06-2023
Published: 20-07-2023

Citation: Veni GK, Satyanarayana C, Krishnareddy MC (2023) Radial Basis Function Based Partition of Unity Method for Two-Dimensional Unsteady Convection Diffusion Equations. Indian Journal of Science and Technology 16(27): 2090-2101. https://doi.org/10.17485/IJST/v16i27.336

* **Corresponding author**.

krishnavenig843@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: The present method aims to solve and investigate the efficiency, accuracy, and stability of the 2D unsteady Navier-Stokes equation in stream function vorticity formulation and Taylor's vortex problem. **Method**: RBF partition of unity method (RBF-PUM) was implemented to solve the two-dimensional Navier- Stokes equations in stream function vorticity formulation and Taylor's vortex problem. **Findings:** RBF-PUM results show good agreement with the exact solutions. The numerical approach is found to be efficient and accurate while maintaining stability even for a Reynolds number as high as 1000. The global RBF method's high computational cost can be overcome by using the RBF-PUM. **Novelty and applications**: The RBF-PU methodology is extended to solve the two-dimensional Navier- Stokes equations in stream function vorticity formulation and Taylor's vortex problem, which were not discussed earlier in the literature. The adaptive spatial refinement within the partitions may be performed independently using the RBF-PUM. Then it may be extended to the more complex problems in CFD.

Keywords: Mesh Free Methods; RBF- PUM; Navier- Stokes Equations; Taylor's Vortex Problem; CFD

1 Introduction

Several scientific and technical applications involve partial differential equations (PDEs), which appear as initial-boundary value problems. These later mechanisms include slow evolution methods like diffusion and heat conduction. The non-linearities of the investigated procedures, the instability of numerical approaches, and the increase in data due to additional dimensions make this class of problems exceedingly difficult analytically and numerically in several different ways. Furthermore, because most PDEs lack an exact solution, it has become crucial in many research areas to develop accurate and effective approximation techniques for calculating the numerical solution of differential equations.

Recent years have seen much work researching "Meshfree" methods[1,2]. The purpose of Meshfree methods is to eliminate the structure of the mesh and approximate the solution entirely using the nodes or data points as a scattered or quasi-random set of points rather than nodes of grid-based discretization. Meshfree character, adaptability
in handling geometrical complexity, spectral rate of convergence, and ease of application to multi-dimensional problems are the meshless approaches' most salient features[3]. One of the meshless techniques was made possible by the groundbreaking work of Kansa, who used Radial Basis Functions (RBFs) to solve PDEs. A scaling parameter is present in the RBFs shown in Table 1. It is widely recognized that the selection of the scaling parameter affects the RBFs' accuracy. There are a few numerical techniques for selecting the ideal scaling parameter value[4].

However, filled collocation matrices produced by using globally supported basis functions are more unreliable and computationally expensive as data set size increases. Restricting the basis function support is the simplest way to localize an RBF collocation method[5]. In this situation, nonzero functions only within their support are used instead of the standard base functions globally supported.

In this section, the RBF-PUM approach[6–8] will be discussed. This strategy uses localization to cut the computational cost of RBF-based methods. The well-known RBF-PUM creates a sufficient number of overlapping subdomains or patches to encapsulate the problem's original domain completely. The global fit is then produced by integrating these approximations after a local RBF approximant is created on each subdomain using compactly supported PU weight functions.

The RBF-PU approach is used in literature to solve PDEs like the pseudo-Parabolic Problem, Poisson Problem, and Unsteady Convection Diffusion Equation. Since the RBF-PUM methodology did not handle Taylor's vortex problem or the 2D Navier-Stokes equations in stream function vorticity formulation, we extended the RBF-PU methodology to solve these problems in the current study. Even for a Reynolds number as high as 1000, the results produced using this method are well aligned with the analytical solutions.

The article's structure is as follows. In section 2, we review the main theoretical concepts of RBF approximation, we briefly present the PUM collocation by RBFs, RBF-PUM method for time dependent PDEs and Poisson problem and stability analysis of our present method. The results of our numerical experiments are presented in section 3. Finally, section 4 is devoted to brief conclusions and future work.

2 Methodology

2.1 Radial Basis Function approximation

2.1.1 Radial Basis Function approximation using conditionally positive definite function

The function $u(x)$ is approximated in a domain $\Omega \subseteq R^d$, at distinct data points or centers $X = \{x_1, ..., x_N\}$, and interpolated by $s_{u,X} : \Omega \to \mathbb{R}$ of the form

$$s_{u,X}(x) = \sum_{j=1}^{N} \alpha_j \emptyset_\varepsilon \left(\left\| x - x_j \right\| \right) + \sum_{k=1}^{Q} \beta_k p_k(x), \quad x \in \mathbb{R}^d \tag{1}$$

α_j & β_k are unknowable real coefficients, $\|.\|$represents the Euclidean distance, and $\varnothing_\varepsilon : (0, +\infty) \to R$ is an RBF with scaling parameter $\varepsilon > 0$ such that $\emptyset_\varepsilon \left(\left\| x - x_j \right\| = \emptyset \left(\varepsilon \left\| x - x_j \right\| \right), \forall x \in \Omega \right.$. A few well-known RBFs are provided in Table 1, together with their smoothness rankings[1]. In equation (1) $p_1, \ldots, p_Q$ form a basis for the space $\pi_{m-1}(R^d)$ with dimension Q whose polynomials of total degree $\leq m-1$ in d variables. To deal with higher degrees of freedom, the interpolation conditions

$$s_{u,X}(x_i) = u(x_i), i = 1, \ldots\ldots, N$$

Are fulfilled by the extra conditions

$$\sum_{j=1}^{N} \alpha_j p_k(x_j) = 0 \quad k = 1, \ldots\ldots, Q \tag{2}$$

A linear equation system is obtained by solving the interpolation problem (1).

$$\begin{pmatrix} A & P \\ P^T & O \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} u \\ \overline{0} \end{pmatrix} \tag{3}$$

where the matrix $A \in R^{N \times N}$ is provided by $A_{ij} = \emptyset \left(\varepsilon \left\| x_i - x_j \right\| \right), i, j = 1, \ldots\ldots, N; P \in \mathbb{R}^{N \times Q}$has entries

$P_{j,k} = (p_k(x_j)), \quad j = 1, \ldots.., N, \quad k = 1, \ldots.., Q, \alpha = (\alpha_1, \ldots., \alpha_N)^T, \ \beta = (\beta_1, \ldots., \beta_Q)^T \ , \quad u = (u_1, \ldots., u_N)^T, \ 0 = (0, \ldots., 0)^T$ and O is a $Q \times Q$ zero matrix. The coefficient matrix on the left must be invertible for the system (3) to be properly solved.

Table 1. Some notable RBF examples using the Euclidean norm $r = ||\cdot||$, and $\nu \in N$.

RBF	$\varnothing\varepsilon(r)$
Gaussian $C^{\infty}(GA)$	$e^{-\varepsilon^2 r^2}$
Multi Quadric $C^{\infty}(MQ)$	$(1+\varepsilon^2 r^2)^{1/2}$
Inverse Multi Quadric$C^{\infty}(IMQ)$	$(1+\varepsilon^2 r^2)^{-1/2}$
Thin plate spline $C^{\nu+1}(TPS)$	$(-1)^{\nu+1} r^{2\nu} logr$
Matérn $C^4(M4)$	$e^{-\varepsilon r}(\varepsilon^2 r^2+3\varepsilon r+3)$
Matérn $C^2(M2)$	$e^{-\varepsilon r}(\varepsilon r+1)$
Wendland $C^4(W4)$	$(1-\varepsilon r)^6_+(35\varepsilon^2 r^2+18\varepsilon r+3)$
Wendland$C^2(W2)$	$(1-\varepsilon r)^6_+(4\varepsilon r+1)$

However, the polynomial $\sum_{k=1}^{Q}\beta_k p_k(x)$, in equation (1), is often needed if ϕ is conditionally positive definite. In the case of positive definite RBFs, such as Gaussian, Multi-Quadric, Inverse Multi-Quadric or Matérn, the polynomial term is not required in equation (1). For any differential operator, Lu can also be approximated, in a similar manner as in (1), as follows:

$$Lu(x) \cong \sum_{j=1}^{N}\alpha_j L\emptyset\left(\varepsilon\left\|x-x_j\right\|\right)+\sum_{k=1}^{Q}\beta_k Lp_k(x).$$

2.1.2 Radial Basis Function approximation using positive definite function

For positive definite function ϕ, the interpolant of $u(x)$ can be formulated without the polynomial term in equation (1). That is,

$$s_{u,X}(x)=\sum_{j=1}^{N}\alpha_j\emptyset\left(\varepsilon\left\|x-x_j\right\|\right) \tag{4}$$

To determine the coefficients $\alpha_1,\ldots,\alpha_N$ the interpolation requirements must be

$$s_{u,X}(x_i)=u(x_i), \quad i=1,\ldots\ldots,N$$

The system of symmetric linear equations can be constructed by imposing these conditions as

$$A\alpha = u, \tag{5}$$

Where $A_{ij}=\left(\emptyset\left(\varepsilon\left\|x_i-x_j\right\|\right), i,j=1,\ldots\ldots,N \quad \alpha=(\alpha_1,\ldots,\alpha_N)^T u=(u_1,\ldots,u_N)^T\right.$. The RBF interpolation problem is well posed if ϕ is a positive definite, in which the corresponding matrix A is invertible. As a result, there is a distinct and existing solution to the problem.

We may therefore calculate the RBF interpolant at any data point x as soon as the vector α is found, i.e.,

$$s_{u,X}(x_i)=\varnothing^T(x)\,\alpha \tag{6}$$

Where $\emptyset^T(x)=(\emptyset(\varepsilon\|x-x_1\|),\ldots,\emptyset(\varepsilon\|x-x_N\|))$.

When a Lagrangian-form RBF interpolant is used, we may solve a time-dependent PDE and approximate the solution $u(x,t)$ as follows.

$$s_{u,X}(x,t)=\sum_{j=1}^{N}\psi_j(x)\,u_j(t)\ , t\geq 0 \tag{7}$$

where $\psi_j(x),\ j=1,\ldots,N$, is the cardinal basis function and has the property

$$\psi_j(x_i)=\begin{cases}1 \text{ if } i=j\\ 0 \text{ if } i\neq j\end{cases}, \quad j=1,\ldots,N,$$

and $u_j(t)=u(x_j,t)$ are the known functions to be determined.

2.2 Method of Partition of Unity using RBFs

2.2.1 RBF- PUM to solve interpolation problems

Let $\Omega \subseteq R^d$ be an open bounded domain and let $\{\Omega_j\}_{j=1}^M$ be an open and bounded covering of Ω that satisfies a minimal overlap requirement between the subdomains (or patches) Ω_j. Moreover, the set $I(x) = \{j : x \in \Omega_j\}$ is uniformly bounded by the constant K – independent of M – on Ω, namely $\text{card}(I(x)) \leq K$, for $x \in \Omega$, where $\Omega \subseteq \bigcup_{j=1}^M \Omega_j$. We construct a partition of unity $\{\omega_j\}_{j=1}^M$ that is the subordinate to the covering $\{\Omega_j\}_{j=1}^M$ in relation to the subdomains such that,

$$\sum_{j=1}^{M} \omega_j(x) = 1,\ x \in \Omega,$$

where the weight function $\omega_j : \Omega_j \to R$ is a compactly supported, nonnegative and continuous with $supp(\omega_j) \subseteq \Omega_j$. Thus, we can establish a local RBF interpolant $s_{u_j,X_j} : \Omega_j \to R$ such that

$$s_{u_j,X_j}(x) = \sum_{j=1}^{N_j} \alpha_i^j \emptyset\left(\varepsilon \left\|x - x_i^j\right\|_2\right) \tag{8}$$

Where N_j is the number of centres in Ω_j i.e., the point $x_i^j \in X_j = X \cap \Omega_j$. Therefore, the global RBF-PUM interpolating polynomial is defined as

$$P_{u,X} = \sum_{j=1}^{M} \omega_j(x)\, s_{u_j,X_j}(x), \qquad x \in \Omega, \tag{9}$$

If the function $s_{u_j,X_j},\ j = 1,\ldots,M$, satisfy the interpolation conditions

$$s_{u_j,X_j}\left(x_i^j\right) = u\left(x_i^j\right), x_i^j \in \Omega_j, i = 1,\ldots,N_j \tag{10}$$

Following that, the global interpolating polynomial (9) acquires the interpolation characteristic of the local interpolants, i.e.,

$$P_{u,X}\left(x_i^j\right) = \sum_{j=1}^{M} \omega_j\left(x_i^j\right) s_{u_j,X_j}\left(x_i^j\right) = \sum_{j=1}^{M} u\left(x_i^j\right) \omega_j\left(x_i^j\right) = u\left(x_i^j\right) \tag{11}$$

Solving the $j-th$ interpolation problem (10) leads to the local RBF linear system.

$$A_j \alpha_j = u_j$$

Where A_j is a $N_j \times N_j$ matrix of entries $A_{ik}^j = \emptyset\left(\varepsilon \left\|x_i^j - x_k^j\right\|\right), i,k = 1,\ldots,N, \alpha_j = \left(\alpha_1^j,\ldots,\alpha_N^j\right)^T$ and $u_j = (u_1^j,\ldots,u_N^j)^T$. It should be noted that similar to (5), the employment of positive definite RBFs $\varnothing$ ensures the solution's existence and uniqueness as well as the non-singularity of the local matrix A_j [9]. The Shepard approach can be utilized to build the PUM weight function ω_j and is given by

$$\omega_j(x) = \frac{\varphi_j(x)}{\sum_{k \in I(x)} \varphi_k(x)}, \quad j = 1,\ldots,M \tag{12}$$

$\varphi_j(x)$ is a compactly supported function with support for Ω_j like the Wendland C^2 function in Table 1. With a shape parameter, such functions are scaled to obtain $\varphi_j(x) = \varphi\left(\varepsilon \left\|x - \xi_j\right\|\right)$, where ξ_j is the center of the function. As a result, $\omega_j(x) = 0, \forall j \notin I(x)$.

Therefore, equation (9) might be revised as

$$P_{u,X}(x) = \sum_{j \in I(X)} \omega_j(x)\, s_{u_j,X_j}(x) \tag{13}$$

And the weight function $\omega_j(x)$ in (12) fulfil the property of partition of unity

$$\sum_{j \in I(X)} \omega_j(x) = 1$$

Therefore, similarly to Equations (11)and (12), if the local fits $s_{u_j,X_j}(x)$ in Eq. (13) interpolate at a specific data point x_i, i.e. $s_{u_j,X_j}(x_i) = u(x_i)$ for each node $x_i \in \Omega_j$, then the global fit likewise interpolates at that point, that is

$$P_{u,X}(x_i) = \sum_{j \in I(x_i)} \omega_j(x_i)\, s_{u_j,X}(x_i) = u(x_i) \sum_{j \in I(x_i)} \omega_j(x_i) = u(x_i)$$

2.2.2 Differential problems with RBF- PUM

When an RBF- PUM technique is used to solve a PDE numerically with time dependence, the differential problem's solution, $u(x,t)$, is approximately determined by the global interpolant.

$$P_{u,X}(x,t) = \sum_{j\in I(X)} \omega_j(x)\, s_{u_j,X_j}(x,t), \quad t \geq 0, \tag{14}$$

Where, analogous to equation (9) $s_{u_j,X_j}(x,t)$ is local RBF interpolant defined on Ω_j of the form

$$s_{u_j,X_j}(x,t) = \sum_{k\in J(\Omega_j)} \psi_k(x)\, u_k(t) \tag{15}$$

With $J(\Omega_j) = (k:\ x_k \in \Omega_j\}$ that specifies the set of nodes in Ω_j. According to equations (14) and (15), the RBF-PUM interpolant can be represented as follows.

$$P_{u,X}(x,t) = \sum_{j\in I(X)} \omega_j(x) \sum_{k\in J(\Omega_j)} \psi_k(x)\, u_k(t) = \sum_{j\in I(X)} \sum_{k\in J(\Omega_j)} (\omega_j(x)\,\psi_k(x)) u_k(t) \tag{16}$$

Therefore, if a time-dependent problem's initial condition was interpolated, we have

$P_{u,X}(x_k,0) = u(x_k,0)\,\forall k$ while $P_{u,X}(x_k,t) \approx u(x_k,t)$ *for* $t > 0$ [10]

We may calculate a derivative term of order μ of the global fit (16) by denoting by μ and ν the multi-indices for common rules and then using Leibniz's rule to generate the derivative rule.

$$\begin{aligned}\frac{\partial^{|\mu|}}{\partial x^\mu} P_{u,X}(x,t) &= \sum_{j\in I(X)} \sum_{k\in J(\Omega_j)} \frac{\partial^{|\mu|}}{\partial x^\mu} (\omega_j(x)\psi_k(x))\, u_k(t) \\ &= \sum_{j\in I(x)} \sum_{k\in J(\Omega_j)} \left(\sum_{\nu\leq\mu} \binom{\mu}{\nu} \frac{\partial^{|\mu-\nu|}\omega_j}{\partial x^{\mu-\nu}}(x) \frac{\partial^{|\nu|}\psi_k}{\partial x^\nu}(x) \right) u_k(t)\end{aligned} \tag{17}$$

If we fix $x = x_i$ and k in equation (17), we obtain the ik - element of the global differentiation matrix.

2.3 Convection-diffusion problem - RBF-PUM scheme

In this subsection, we apply RBF-PUM for time dependent PDE, specifically to solve unsteady convection-diffusion equation.

Let us consider the following unsteady convection-diffusion equation

$$\frac{\partial u}{\partial t} = k\Delta u(x,t) + v.\nabla u(x,t) \equiv \mathscr{L}u(x,t), x \in \Omega \subseteq \mathbb{R}^d, t > 0, \tag{18}$$

where Laplacian and gradient operators are denoted by Δ and ∇ respectively, and L is the convection-diffusion operator [11]. In addition, $u(x,t)$ stands for concentration or temperature for mass or heat transport, v is the constant velocity vector and k is the diffusion coefficient. Adding an initial condition to equation (18) of the form is necessary.

$$u(x,0) = u_0(x) \tag{19}$$

And boundary conditions

$$\mathscr{B}u(x,t) = g(x,t), x \in \partial\Omega, t > 0 \tag{20}$$

Here $g(x,t)$ is a known function, $\mathfrak{P}$ is a boundary operator, that may be Dirichlet, Neumann or mixed type, and $\partial\Omega$ denote the boundary of Ω. We can discretize the time derivative of PDE (18) in the case of Dirichlet boundary conditions using the standard FD formula and the θ -weighted scheme

$$\frac{u^{n+1}(x) - u^n(x)}{\delta t} = \theta\left(k\,\triangle u^{n+1}(x) + v.\nabla u^{n+1}(x)\right) + (1-\theta)\left(k\,\triangle u^n(x) + v.\nabla u^n(x)\right) \tag{21}$$

Or,

$$\frac{u^{n+1}(x) - u^n(x)}{\delta t} = \theta L u^{n+1}(x) + (1-\theta)\,\mathscr{L}u^n(x) \tag{22}$$

where $0 \leq \theta \leq 1$, $u^{n+1}(x) = u\left(x, t^{n+1}\right)$, $t^{n+1} = t^n + \delta t$, δt is the time step size.
From equation (21) or (22), we get

$$u^{n+1}(x) + \eta\left(k\Delta u^{n+1}(x) + v \cdot \nabla u^{n+1}(x)\right) = u^n(x) + \zeta\left(k\Delta u^n(x) + v \cdot \nabla u^n(x)\right) \tag{23}$$

Or,

$$u^{n+1}(x) + \eta \mathscr{L} u^{n+1}(x) = u^n(x) + \zeta \mathscr{L} u^n(x) \tag{24}$$

where $\eta = -\theta\delta t$ and $\zeta = (1-\theta)\delta t$.

Relating to equation (16), the approximated value of $u^n(x)$ is

$$u^n(x) \simeq p^n(x) = \sum_{j \in I(x)} \sum_{k \in J\left(\Omega_j\right)} \left(\omega_j(x)\psi_k(x)\right) u_k^n \tag{25}$$

The $N \times N$ matrix A can be divided into two matrices A_I and A_B, if I and B are internal and boundary points, respectively, and let N be the total number of centres, which is equal to $N_I + N_B$. Hence A can be written as

$$A = A_I + A_B \tag{26}$$

where,

$$A = \left[\omega_j(x_i)\,\psi_k(x_i) \text{ for } \left(j \in I(x_i), k \in J(\Omega_j), i = 1, 2, \ldots, N\right) \textit{ and } 0 \textit{ elsewhere}\right]_{N \times N},$$

$$A_I = \left[a_{ij} \text{ for } (i \in I, 1 \leq j \leq N) \textit{ and } 0 \textit{ elsewhere}\right],$$

$$A_B = \left[a_{ij} \text{ for } (i \in B, 1 \leq j \leq N) \textit{ and } 0 \;\; \textit{elsewhere}\right]$$

Substituting (25) in equation (23) together with (20), the resulting sparse system can be expressed in matrix form

$$Cu^{n+1} = Du^n + v^{n+1} \tag{27}$$

where,

$$C = A + \eta k \Delta A_I + \eta v \cdot \nabla A_I,$$

$$D = A + \zeta k \Delta A_I + \zeta v \cdot \nabla A_I,$$

$$v^{n+1} = \left[g_i^{n+1} \; for \; (i \in B) \; and \; 0 \; elsewhere\right]^T,$$

$$u^n = (u_1^n, \ldots, u_N^n)^T$$

Combining equations (21), which deals with internal points, and (20), which deals with boundary points, yields the system (27). Therefore, using equation (19), which represents the internal condition, we may determine u^{n+1} by solving (27). Afterwards, by replacing such values of u^n in

$$p^n = Au^n \tag{28}$$

at time level n, we can get the approximate solution to the PDE.

2.4 RBF- PUM for Poisson problem

The Poisson problem with Dirichlet boundary conditions is defined by the Laplace operator $L = \triangle$ as

$$\triangle u(x) = f(x), \quad x \in \Omega \tag{29a}$$

$$u(x) = g(x), \quad x \in \partial\Omega \tag{29b}$$

After discretizing the problem (29) on a global set of collocation points

$$X_N = X_{N_i} \cup X_{N_b} = \{x_1, \ldots, x_N\} = \{x_{i,1}, \ldots, x_{i,N}\} \cup \{x_{b,1}, \ldots, x_{b,N}\}$$

where the numbers of interior and boundary nodes areN_i and N_b respectively. Using the assumption that the Poisson problem allows for an approximate solution of the type (9), we obtain,

$$\Delta P_{u,X}(x_i) = \sum_{j=1}^{d} \Delta\left(w_j(x_i)\, s_{u_j,X_j}(x_i)\right) = f(x_i), x_i \in \Omega \tag{30a}$$

$$P_{u,X}(x_i) = \sum_{j=1}^{d} \left(w_j(x_i)\, s_{u_j,X_j}(x_i)\right) = g(x_i), x_i \in \partial\Omega \tag{30b}$$

This allows the Laplace operator $\triangle$to be expanded as

$$\triangle\left(w_j(x_i)\, s_{u_j,X_j}(x_i)\right) =$$

$$\Delta w_j(x_i)\, s_{u_j,X_j}(x_i) + 2\nabla w_j(x_i) \cdot \nabla s_{u_j,X_j}(x_i) + w_j(x_i)\, \Delta s_{u_j,X_j}(x_i), x_i \in \Omega \tag{31}$$

The local nodal values vector can be defined as $s_{u_j,X_j} = \left(s_{u_j,X_j}\left(x_1^j\right), \ldots, s_{u_j,X_j}\left(x_{N_j}^j\right)\right)^T$, while the local coefficient vector $\alpha_j = \left(\alpha_1^j, \ldots, \alpha_{N_j}^j\right)^T$ is such that $\alpha_j = A_j^{-1} s_{u_j,X_j}$. So, we get

$$\Delta s_{u_j,X_j} = A_j^{\Delta} A_j^{-1} s_{u_j,X_j}, \quad \nabla s_{u_j,X_j} = A_j^{\nabla} A_j^{-1} s_{u_j,X_j} \tag{32}$$

Where $A_j^{\triangle}$ and A_j^{∇} are the matrices with entries

$$\left(A_j^{\Delta}\right)_{ki} = \Delta\emptyset\left(\left\|x_k^j - x_i^j\right\|_2\right), \text{and } \left(A_j^{\nabla}\right)_{ki} = \nabla\emptyset\left(\left\|x_k^j - x_i^j\right\|_2\right), j = 1, \ldots, N_j.$$

Additionally, the diagonal matrix is defined

$$W_j^{\triangle} = diag(W_j^{\triangle}\left(x_1^j\right), \ldots, W_j^{\triangle}\left(x_{N_j}^j\right))$$

corresponding to each subdomain, and similarly $W_j^{\triangle}$ and W_j. We must differentiate (30) using a product derivative rule before using the relation in (32) to derive the discrete operator L_j. Boundary conditions and equation (31) allow us to express the discrete local Laplacian as

$$(L_j)_{ki} = \begin{cases} (\bar{L}_j)_{ki}, x_i^j \in \Omega \\ \delta_{ki}, x_i^j \in \partial\Omega \end{cases}$$

where δ_{ki} is the Kronecker delta and

$$\bar{L}_j = \left(W_j^{\triangle} + 2W_j^{\nabla}.A_j^{\nabla} + W_j A_j^{\triangle}\right) A_j^{-1}.$$

The global discrete operator is then obtained by combining the local matrices L_j into the global matrix L of entries

$$(L_j)_{ki} = \sum_{j=1}^{d} \quad (L_j)_{\eta_{kj},\eta_{ij}} \qquad k, i = 1, \ldots, N.$$

Consequently, we must resolve the linear system

$$Lz = u \tag{33}$$

where $z = (P_{u,X}(x_1), \ldots, P_{u,X}(x_N))^T$ and $u = (u_1, \ldots, u_N)^T$ defined by,

$$u_i = \begin{cases} f(x_i), & x_i \in \Omega \\ g(x_i), & x_i \in \partial\Omega \end{cases}$$

2.5 Stability analysis for RBF-PUM

This section contains a numerical stability study of RBF-PUM that explicitly considers the time-dependent equation under consideration. We introduce an error in equation (27). $e^n = u^n - p^n$ is the value we choose, where u^n is the exact solution and p^n is the approximate solution. Now, the error's equation can be written as

$$e^{n+1} = Ke^n \tag{34}$$

where $K = AC^{-1}DA^{-1}$ is the amplification matrix. The numerical method will be stable if as $n \to \infty$ the error $e^n \to 0$. This can be ensured if $\rho(K) < 1$, where $\rho(K)$ stands for the amplification matrix's spectral radius. When K is substituted in equation (34), we get,

$$CA^{-1}e^{n+1} = DA^{-1}e^n \tag{35}$$

Implying Dirichlet boundary conditions, equation (35) can be written as

$$[I - \theta\delta tM]e^{n+1} = [I + (1-\theta)\delta tM]e^n, \tag{36}$$

Where the identity matrix $I \in R^{N\times N}$ and the matrix $M = \mathscr{L}A_I A^{-1}$.

Equation (36) shows that stability is guaranteed if every eigenvalue of the matrix $[I + \theta\delta tM]^{-1}[I - (1-\theta)\delta tM]$ is less than unity, i.e.,

$$\left|\frac{1 + (1-\theta)\delta t\lambda_M}{1 - \theta\delta t\lambda_M}\right| \leq 1, \tag{37}$$

where M is a matrix, and λ_M is one of its eigenvalues. the eigen values of the matrix M can be determined by figuring out the generalized eigen value problem,

$$\mathscr{L}A_I s = \lambda_M A s$$

In the case of the Crank-Nicholson scheme, that is, for $\theta = \frac{1}{2}$ the inequality (37) is always satisfied if $\lambda_M \leq 0$. This indicates that the numerical system is unconditionally stable when $\lambda_M \leq 0$.

3 Results and Discussion

Using some of the RBFs in Table 1, the numerical outcomes of the PUM approach can be demonstrated. In the equation (8), the basis functions for our analysis are the Gaussian (GA), Multi Quadric (MQ), and Inverse Multi Quadric (IMQ). The function of Shepard's weight is localized using the compactly supported RBF W2 in (12). The maximum absolute error (MAE) can be calculated to check the accuracy. The PU covering is composed instead of M circular patches that are centred at a uniform grid points, where the overlap of the patches is 20% of the distance between the centers [10].

In spite of the fact that the approximation method discussed in section 4 is applicable for all values of $\theta \in [0, 1]$, we focus on $\theta = \frac{1}{2}$, which identifies the well-known Crank-Nicholson scheme. The numerical approach is found to be efficient, accurate, and stable. The proposed method is used for two unsteady test problems to examine their validity and efficacy. These unsteady problems are (i) the two- dimensional Navier-Stokes equation in stream function and vorticity formulation and (ii) the Taylor's vortex problem. Since these problems can be solved analytically, Dirichlet boundary conditions are employed.

Problem 1

The problem of flow of decayed viscosity[12] is governed by two- dimensional Navier- Stroke's equation in non- dimensional form for an incompressible flow[13] can be written as

$$\frac{\partial u}{\partial x}+\frac{\partial u}{\partial y}=0 \tag{38}$$

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{\partial P}{\partial x}+\frac{1}{Re}\nabla^2 u \tag{39}$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=-\frac{\partial P}{\partial y}+\frac{1}{Re}\nabla^2 v \tag{40}$$

in the square $0\leq x,y\leq \pi, t>0$. Here u and v are the velocities in the directions x and y, Re is the Reynold's number, and P is the pressure. The initial conditions are

$$u(x,y,0)=\sin x\cos y \tag{41a}$$

$$v\,(x,y,0)=-cosx\,siny \tag{41b}$$

and the boundary conditions at $x=0,\ x=\pi; y=0, y=\pi$ are provided by the relations

$$u(x,y,t)=\sin x\cos y e^{\frac{-2t}{Re}} \tag{42a}$$

$$v(x,y,t)=-\cos x\sin y e^{\frac{-2t}{Re}} \tag{42b}$$

The analytical solution to this problem is

$$u(x,y,t)=\sin x\cos y e^{\frac{-2t}{Re}} \tag{43a}$$

$$v(x,y,t)=-\cos x\sin y e^{\frac{-2t}{Re}} \tag{43b}$$

$$P=-\frac{1}{4}(\cos 2x+\sin 2y)e^{\frac{-2t}{Re}} \tag{43c}$$

Introducing stream- function ψ and vorticity ω, equations (38) to (40) can be written as

$$\frac{\partial \omega}{\partial t}+u\frac{\partial \omega}{\partial x}+v\frac{\partial \omega}{\partial y}=\frac{1}{Re}\nabla^2\omega \tag{44}$$

$$\nabla^2\psi=-\omega \tag{45}$$

In cartesian coordinate system u and v are given by

$u=\frac{\partial \psi}{\partial y}$ and $v=-\frac{\partial \psi}{\partial y}$

where the initial and boundary conditions for ψ and ω can be derived from equations (41) and (42).

The analytic solution to this problem is,

$$u = \sin(x)\cos(y)e^{-2t/Re} \tag{46a}$$

$$v = -\cos(x)\sin(y)e^{-2t/Re} \tag{46b}$$

and

$$\psi = \sin(x)\sin(y)e^{-2t/Re} \tag{47a}$$

$$\omega = 2\sin(x)\sin(y)e^{-2t/Re} \tag{47b}$$

Results for three different Reynolds numbers viz. 50, 100 and 1000 for a grid size 21×21 are shown in the tables Tables 2, 3 and 4. In all cases, we fix $\delta t = 0.01$ and final time $T = 1$. The numerical results of RBF- PUM for 2D Navier- Stroke's equation in stream function and vorticity formulation are well-aligned with the exact solution.

Table 2. RBF-PUM using Gaussian function

Grid	Re	MAE in ψ	MAE in ω
21×21	50	$4.21e-04$	$1.80e-01$
	100	$4.31e-04$	$2.63e-02$
	1000	$4.41e-04$	$5.12e-04$

Table 3. RBF-PUM using Multi Quadric function

Grid	Re	MAE in ψ	MAE in ω
21×21	50	$9.06e-04$	$4.68e-02$
	100	$9.29e-04$	$2.23e-02$
	1000	$9.51e-04$	$1.72e-03$

Table 4. RBF-PUM using Inverse Multi Quadric function

Grid	Re	MAE in ψ	MAE in ω
21×21	50	$3.45e-03$	$1.05e-01$
	100	$3.54e-03$	$4.52e-02$
	1000	$3.62e-03$	$3.92e-03$

Problem 2

In this example, we consider the Taylor's vortex problem [12], equations (44) and (45) with the initial conditions,

$$u(x,y,0) = \sin(\mathscr{P}x)\cos(\mathscr{P}y), 0 \leq x,y \leq 2\pi \tag{48a}$$

$$v(x,y,0) = -\cos(\mathscr{P}x)\sin(\mathscr{P}y), 0 \leq x,y \leq 2\pi \tag{48b}$$

The analytical solution is given by,

$$u = \sin(\mathscr{P}x)\cos(\mathscr{P}y)e^{-2\mathscr{P}^2 t/Re} \tag{49a}$$

$$v = -\cos(\mathscr{P}x)\sin(\mathscr{P}y)e^{-2\mathscr{P}^2 t/Re} \tag{49b}$$

and

$$\psi = \sin(\mathscr{P}x)\sin(\mathscr{P}y)e^{-2\mathscr{P}^2 t/Re} \tag{50a}$$

$$\omega = 2\sin(\mathscr{P}x)\sin(\mathscr{P}y)e^{-2\mathscr{P}^2 t/Re} \tag{50b}$$

where P is an integer.

Numerical results are presented in Tables 5, 6 and 7 for Reynold's number $Re = 1000$ and for $P = 1,\ 2$ and 4. The RBF-PUM results and the exact solution are in good agreement.

Table 5. RBF-PUM using Gaussian function (Re=1000)

Grid	$\mathscr{P}$	MAE in ψ	MAE in ω
21×21	1	$1.66e-02$	$8.24e-03$
	2	$6.72e-01$	$1.43e-02$
	4	$7.98e-01$	$1.82e-02$

Table 6. RBF-PUM using Multi Quadric function (Re=1000)

Grid	$\mathscr{P}$	MAE in ψ	MAE in ω
21×21	1	$2.38e-02$	$4.90e-03$
	2	$6.70e-01$	$1.12e-02$
	4	$8.00e-01$	$3.09e-02$

Table 7. RBF-PUM using Inverse Multi Quadric function (Re=1000)

Grid	$\mathscr{P}$	MAE in ψ	MAE in ω
21×21	1	$3.21e-02$	$8.68e-03$
	2	$6.72e-01$	$1.74e-02$
	4	$8.02e-01$	$4.25e-02$

4 Conclusions

Using three different Radial Basis Functions— The Gaussian, The Multi Quadric, and The Inverse Multi Quadric Basis functions—we have implemented and tested the RBF-PUM scheme to solve the 2D unsteady Navier-Stokes equation in stream function vorticity formulation, and Taylor's vortex problem, which was not previously covered by the RBF-PU methodology. The global RBF method's high computational cost can be overcome using the RBF-PUM, and the results obtained using this method are well-aligned with the analytical solutions even for a Reynolds number as high as 1000.

Preconditioning is required since the RBF-PUM matrix is non-symmetric, ill-conditioned, and sparse. In future work, the adaptive spatial refinement within the partitions may be performed independently using the RBF-PUM. Then it may be extended to the more complex problems in CFD.

References

1) Jeong SM, Lee CY. Weighted Moving Square-Based Solver for. *Unsteady Incompressible Laminar Flow Simulations Applied Sciences*. 2022;12(7):3519. Available from: https://doi.org/10.3390/app12073519.
2) Sheikhi N, Najafi M, Enjilela V. Extending the Meshless Local Petrov–Galerkin Method to Solve Stabilized Turbulent Fluid Flow Problems. *International Journal of Computational Methods*. 2019;16(01). Available from: https://doi.org/10.1142/S021987621850086X.
3) Barik NB, Tatavarthi V. A modified multilevel meshfree algorithm for steady convection-diffusionproblems. 2021. Available from: https://doi.org/10.1002/fld.4967.
4) Sanyasiraju YVSS, Satyanarayana C. On optimization of the RBF shape parameter in a grid-free local scheme for convection dominated problems over non-uniform centers. *Applied Mathematical Modelling*. 2013;37(12-13):7245–7272. Available from: https://doi.org/10.1016/j.apm.2013.01.054.
5) Schoder S, Roppert K, Weitz M, Junger C, Kaltenbacher M. Aeroacoustic source term computation based on radial basis functions. 2020. Available from: https://doi.org/10.1002/nme.6298.

6) Ben-Ahmed EH, Sadik M, Wakrim M. A Stable Radial Basis Function Partition of Unity Method with d-Rectangular Patches for Modelling Water Flow in Porous Media. *Journal of Scientific Computing*. 2020;84(1). Available from: https://doi.org/10.1007/s10915-020-01273-2.
7) Cavoretto R. Adaptive Radial Basis Function Partition of Unity Interpolation: A Bivariate Algorithm for Unstructured Data. *Journal of Scientific Computing*. 2021;87(2):41. Available from: https://doi.org/10.1007/s10915-021-01432-z.
8) Ben-Ahmed EH, Sadik M, Wakrim M. Radial basis function partition of unity method for modelling water flow in porous media. *Computers & Mathematics with Applications*. 2018;75(8):2925–2941. Available from: https://doi.org/10.1016/j.camwa.2018.01.022.
9) Fasshauer GE. Meshfree Approximation Methods with Matlab. *World Scientific*. 2007;6:491. Available from: https://doi.org/10.1142/6437.
10) Safdari-Vaighani A, Heryudono A, Larsson E. A radial basis function partition of unity collocation method for convection-diffusion equations. *Journal of Scientific Computing*. 2015;64:341–367. Available from: https://doi.org/10.1016/j.cam.2017.01.024.
11) Garmanjani G, Cavoretto R, Esmaeilbeigi M. A RBF partition of unity collocation method based on finite difference for initial–boundary value problems. *Computers & Mathematics with Applications*. 2018;75(11):4066–4090. Available from: http://doi.org/10.1016/j.camwa.2018.03.014.
12) Jiten C, Kalita DC, Dalal, Anoop K. Dass: A class of higher order compact schemes for the unsteady two- dimensional convection- diffusion equation with variable convection coefficients. 2002. Available from: https://doi.org/10.1002/fld.263.
13) Balmus M, Hoffman J, Massing A, Nordsletten DA. A stabilized multidomain partition of unity approach to solving incompressible viscous flow. *Computer Methods in Applied Mechanics and Engineering*. 2022;392(2):114656. Available from: https://doi.org/10.1016/j.cma.2022.114656.