

Enhancing RBF-FD Efficiency for Highly Non-Uniform Node Distributions via Adaptivity

Siqing Li¹, Leevan Ling^{2,*}, Xin Liu³, Pankaj K. Mishra⁴,
Mrinal K. Sen⁵ and Jing Zhang¹

¹ College of Mathematics, Taiyuan University of Technology, Taiyuan, China

² Department of Mathematics, Hong Kong Baptist University, Hong Kong, SAR, China

³ Institute of Advanced Technology, University of Science and Technology of China, Hefei, China

⁴ Geological Survey of Finland

⁵ University of Texas at Austin, USA

Received 12 August 2023; Accepted (in revised version) 8 January 2024

Abstract. Radial basis function generated finite-difference (RBF-FD) methods have recently gained popularity due to their flexibility with irregular node distributions. However, the convergence theories in the literature, when applied to nonuniform node distributions, require shrinking fill distance and do not take advantage of areas with high data density. Non-adaptive approach using same stencil size and degree of appended polynomial will have higher local accuracy at high density region, but has no effect on the overall order of convergence and could be a waste of computational power. This work proposes an adaptive RBF-FD method that utilizes the local data density to achieve a desirable order accuracy. By performing polynomial refinement and using adaptive stencil size based on data density, the adaptive RBF-FD method yields differentiation matrices with higher sparsity while achieving the same user-specified convergence order for nonuniform point distributions. This allows the method to better leverage regions with higher node density, maintaining both accuracy and efficiency compared to standard non-adaptive RBF-FD methods.

AMS subject classifications: 65N12, 65N35

Key words: Partial differential equations, radial basis functions, meshless finite difference, adaptive stencil, polynomial refinement, convergence order.

1. Introduction

In the past two decades, there has been important progress in developing adaptive mesh methods for PDEs. Mesh adaptivity is usually of two types in form: local mesh refinement and moving mesh method.

*Corresponding author. *Email address:* lisiqing@tyut.edu.cn (S. Li)

Radial basis functions (RBFs) have been a popular choice in the development of kernel-based meshless methods for solving partial differential equations (PDEs) numerically. Besides collocation methods, the localized RBF-FD method has gained popularity in recent years due to its many advantages, including numerical stability on irregular node layouts, high computational speed and accuracy, easy local adaptive refinement, and excellent opportunities for large-scale parallel computing.

The idea of RBF and kernel-based differentiation can be traced back to [28], and it was formally introduced as the RBF-FD in [29]. Since then, a significant amount of research has been dedicated towards the robust development of the RBF-FD method [2, 3, 6, 9, 10, 17, 19, 20, 25, 30], as well as its application to various problems in science and engineering [4, 5, 7, 11–15, 26, 27]. In addition to the RBF-FD method, other collocation methods based on radial basis functions have also been proposed. For example, a global radial basis function collocation method in [21] was successfully applied to solve a computational fluid dynamic problem, and local RBF collocation methods were used to solve the diffusion problem in [22] and Hamiltonian PDEs in [32].

The RBF-FD method is advantageous since it works with scattered nodes, allowing for stencils with different configurations and overcoming the fixed grid/element limitation of conventional numerical methods. Unlike global meshless methods, RBF-FD computes weights locally using RBFs expanded at a fixed number of nearest nodes. Once weights at each node are computed, they can be stored and used for next-step computation, making weight computation a pre-processing step in solving time-dependent PDEs. Furthermore, weight computations at each node are independent processes, making RBF-FD a desirable method for parallel computing.

It has been demonstrated [2, 6] that combining polynomial basis with polyharmonic spline radial basis functions (PHS+Poly) in the RBF-FD formulation leads to considerable improvements in robustness. Key benefits of the PHS+Poly approach include:

1. It is free of shape parameter, simplifying the formulation and eliminating the need for fine-tuning.
2. It is efficient compared to stable RBF-FD formulations based on infinitely smooth RBFs [20].
3. This method ensures accuracy near boundaries without ghost-nodes where stencils become highly one-sided [1].
4. It has the potential to maintain accuracy for large and sparse linear systems.
5. The convergence order depends mainly on the augmented polynomial degree, which determines the stencil size.

Existing convergence theories for scattered nodes require shrinking fill distance and fail to leverage regions of high node density. Numerically, non-adaptive methods using uniform stencil size and polynomial degree may exhibit loss of accuracy in low-density regions while adding unnecessary complexity in high-density regions,

decreasing computational efficiency. Rather than a fixed stencil size and augmented polynomial degree, we propose an adaptive RBF-FD algorithm that allows the user to define a global convergence order with respect to the total number of nodes in the domain. The method can efficiently obtain the user-defined convergence order in a highly nonuniform node-layout with a significantly large mesh ratio. Note that the mesh ratio is defined as the fill distance over minimum separating distance, which plays a role similar to mesh ratio in finite element methods.

The rest of the paper is organized as follows. Section 2 discusses the general formulation of RBF-FD and polynomial augmentation, along with some insights for its stable implementation. In Section 3, we provide a precise definition of the global convergence order in a nonuniform node-layout. We also present a method that connects the global convergence order to the required degree of polynomial at a local scale, depending on the local fill distance. This is followed by a discussion of the weight computation approach through the adaptive RBF-FD method. In Section 4, numerical examples are illustrated to show the advantages of the proposed method. Finally, Section 5 is the conclusions drawn from our study.

2. RBF-FD approximation

We first review the basic RBF-FD method. Let $\Omega \subset \mathbb{R}^d$ be a computational domain with a set of distinct scattered nodes $X := \{x_1, x_2, \dots, x_N\} \subset \Omega$. Given a linear differential operator \mathcal{L} , we aim to approximate the values of $\mathcal{L}u$ at a center/point of interest $x^c \in X$ based on the nodal values of u at some stencil or set of neighboring nodes of x^c in X . Fig. 1 shows the prototype of the scatter data distribution and local stencil for the point of interest.

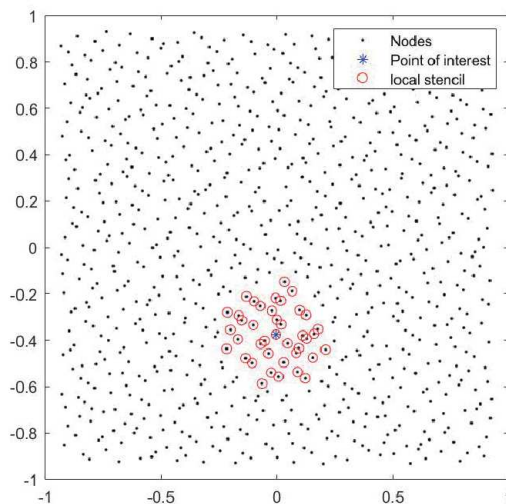


Figure 1: A prototype figure showing centers and local stencil.

Let the stencil of \mathbf{x}_c be denoted by $X_c := \{\mathbf{x}_1^c, \dots, \mathbf{x}_{n_s}^c\} \subseteq X$, where n_s^c is the stencil size of \mathbf{x}_c . Then, the value of $\mathcal{L}u(\mathbf{x}_c) \in \mathbb{R}$ can be approximated by as

$$\mathcal{L}u(\mathbf{x}_c) \approx \sum_{k=1}^{n_s^c} w_k^c u(\mathbf{x}_k^c) = \mathbf{w}_c u|_{X_c}, \quad (2.1)$$

where $u|_{X_c} := [u(\mathbf{x}_1^c), \dots, u(\mathbf{x}_{n_s}^c)]^T$ is the nodal values of u evaluated at stencil X_c . We need to compute the weights $\mathbf{w}_c := [w_1^c, w_2^c, \dots, w_{n_s}^c]^T$ at \mathbf{x}_c for this finite difference approximation.

In RBF-FD method, interpolants were used as surrogate. Using some symmetric positive definite (SPD) radial basis function $\phi : [0, \infty) \rightarrow \mathbb{R}$, the RBF surrogate interpolates the data at $(X_c, u|_{X_c}) \in \mathbb{R}^d \times \mathbb{R}$ while keeping function values $u|_{X_c}$ as unknown. The RBF-FD weights \mathbf{w}_c are computed by applying \mathcal{L} to the surrogate and, then, evaluating at \mathbf{x}_c . This is equivalent to solving

$$\underbrace{[\phi(\|\mathbf{x}_i^c - \mathbf{x}_j^c\|)]_{1 \leq i, j \leq n_s^c}}_{=: A_c} \mathbf{w}_c = \underbrace{[\mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_i^c\|)|_{\mathbf{x}=\mathbf{x}_c}]_{1 \leq i \leq n_s^c}}_{=: [\mathcal{L}\phi]|_{\mathbf{x}_c}}. \quad (2.2)$$

Since ϕ is SPD, A_c is SPD and (2.2) is uniquely solvable. It is shown in [23] that (2.2) is error optimal in the native space norm corresponding to the radial basis function ϕ . We refer readers to the article for details.

2.1. RBF-FD using PHS+Poly

In contrast to global meshless methods, the local nature of RBF-FD rules out exponential convergence for approximating functions in certain native spaces. Even with infinitely smooth RBFs, the convergence is limited for RBF-FD by stencil sizes. Motivated by the discussion in [24], we use a conditionally positive definite kernel with finite order of smoothness in the RBF-FD method. For example, we use a conditionally SPD polyharmonic spline (PHS) kernel

$$\phi(r) = r^{2m+1}, \quad m \in N^+.$$

To ensure unique solvability, we augment polynomial basis of sufficient degrees to ensure exact reproduction of low order polynomials. To compute RBF-FD weights in (2.1) by PHS+Poly, the surrogate $I(\mathbf{x})$ now uses RBFs and polynomial basis up to degree p as basis functions

$$I(\mathbf{x}) \approx \sum_{k=1}^{n_s^c} \lambda_k \phi(\|\mathbf{x} - \mathbf{x}_k^c\|) + \sum_{l=1}^{n_p^c} \gamma_l p_l(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (2.3)$$

subject to interpolation conditions at data $(X_c, u|_{X_c}) \in \mathbb{R}^d \times \mathbb{R}$ and constraints

$$\sum_{k=1}^{n_s^c} \gamma_l p_l(\mathbf{x}_k^c) = 0, \quad l = 1, 2, 3, \dots, n_p^c, \quad (2.4)$$

where n_p^c is the number of augmented multivariate polynomial basis and n_s^c is again the stencil-size of \mathbf{x}_c .

The counterpart to the linear system (2.2) for RBF-FD using PHS+Poly is given by

$$\begin{bmatrix} A_c & P_c \\ P_c^T & O \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{w}_e \end{bmatrix} = \begin{bmatrix} [\mathcal{L}\phi]_{|\mathbf{x}_c} \\ [\mathcal{L}p]_{|\mathbf{x}_c} \end{bmatrix}, \quad (2.5)$$

which can be interpreted as an equality-constrained quadratic programming problem (see [2] for more details on this). In (2.5), matrices A_c and $[\mathcal{L}\phi]$ are defined as in (2.2). The $n_s^c \times n_p^c$ matrix P_c and $n_p^c \times 1$ vector $[\mathcal{L}p]_{|\mathbf{x}_c}$ are given as

$$P_c = [p_j(\mathbf{x}_i^c)]_{1 \leq i \leq n_s^c, 1 \leq j \leq n_p^c} \quad \text{and} \quad [\mathcal{L}p]_{|\mathbf{x}_c} = [\mathcal{L}p_l(\mathbf{x})]_{|\mathbf{x}=\mathbf{x}_c, 1 \leq l \leq n_p^c}.$$

From here and on, we drop the super/subscript c from notations for simplicity, and all computations need to be done center by center for all $\mathbf{x}_c \in X$. The number of polynomial basis n_p up to degree p in d -dimensional is given by

$$n_p = C(p + d, d) = \frac{(p + d)!}{p!d!}. \quad (2.6)$$

Table 1 lists out polynomial terms for different degrees in some cases in 2D, see [6] for more. For example, $n_p = 3$ in 2D, corresponds to appending polynomial up to first order.

Table 1: The polynomial basis of different degree for 2D case.

Polynomial degree (p)	Polynomial basis	$n_p = C(p + 2, 2) = \frac{(p+2)!}{p!2!}$
0	1	1
1	1, x , y	3
2	1, x , y , x^2 , y^2 , xy	6
3	1, x , y , x^2 , y^2 , xy , x^3 , y^3 , x^y , xy^2	10

3. Adaptive PHS+Poly RBF-FD methods

Adaptive methods are quite popular in the context of finite element methods (FEM), and many algorithms such as h -FEM, p -FEM, or hp -FEM were proposed. In order to meet a certain order of accuracy throughout the domain, the idea behind adaptivity is to vary parameters affecting the accuracy of the algorithm locally. In this section, we aim to develop an adaptive PHS+Poly RBF-FD scheme that:

1. Achieve the expected order of accuracy in the domain, including evaluation points near boundaries where stencils are highly one-sided.
2. Maintain accuracy and obtain linear systems with higher sparsity for dealing with large-scale problems.

3.1. Polynomial refinement and adaptive stencil size

Given a set of N scattered data $X \subset \mathbb{R}^d$, its fill and separation distances are respectively given by

$$h_{X,\Omega} := h_X = \sup_{\mathbf{x} \in \Omega} \min_{\mathbf{x}_j \in X} \|\mathbf{x} - \mathbf{x}_j\|_{\ell_2(\mathbb{R}^d)}, \quad q_X := \frac{1}{2} \min_{i \neq j} \|\mathbf{x}_i - \mathbf{x}_j\|_{\ell_2(\mathbb{R}^d)}, \quad (3.1)$$

where $\rho_X = h_X/q_X \geq 1$ is the mesh ratio of X . A quasi-uniform data set refers to a sequence of data sets where the mesh ratio ρ_X remains bounded for all N as X refines with increasing N . The number of nodes N typically scales as $N \sim q_X^{-d}$, where \sim means proportional up to a constant. Fig. 2 shows node distributions of $N = 4,096$ under mesh ratios $\rho_X = 1, 10^2$, and 10^5 , respectively.

Data sets with large mesh ratios will be considered highly nonuniform, which is the target of our proposed adaptive approach. Nonuniform nodes are often associated with rapidly varying regions of a function or other physical quantities. For example, the adaptive node distributor in NodeLab [8, 16] can generate a sequence of scattered data sets with large mesh ratios ρ that are denser in user specified problematic spatial regions. The folklore is that high point density will induce a smaller stencil radius even if the stencil size n_s is fixed at a constant for all centers. Yet this could yield an inefficient algorithm, which we aim to improve upon.

In the RBF-FD method, for quasi-uniform node distribution, the convergence can be estimated in a straightforward manner. Suppose we want to approximate a k -th order differential operator \mathcal{L} by PHS+Poly RBF-FD with appended polynomial up to degree p . According to results in [2, 6], for a quasi-uniform node distribution, the convergence order of a RBF-FD method is $\sim h_X^{p-k+1}$. It has been found in [2, 6] that the convergence order of PHS+Poly RBF-FD was governed by the degree of augmented polynomial p , while the smoothness order of the PHS has a marginal effect on the accuracy and no influence on the order of convergence.

Let us first consider an ideal quasi-uniform point set X of N points in Ω (with minimal mesh ratio) that has an equispaced fill distance proportional to

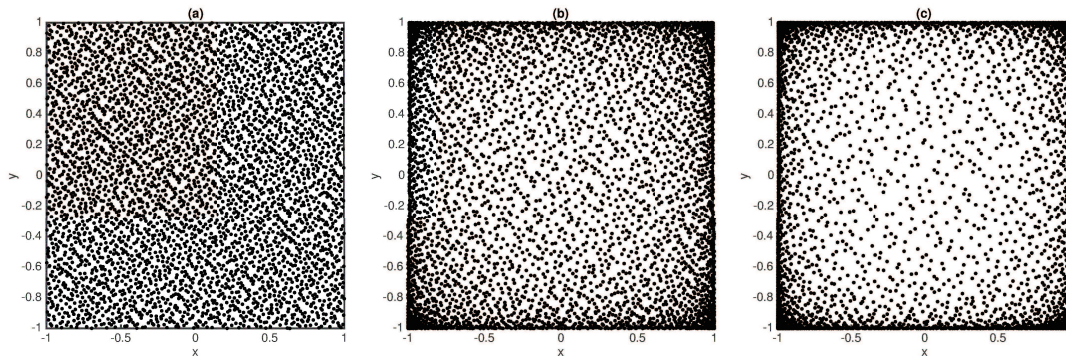


Figure 2: Node distribution under different mesh ratio with $N = 4096$ quasi-uniform data points in unit square $[-1, 1]^2$ with various mesh ratios: (a) $\rho_X \approx 1$, (b) $\rho_X \approx 10^2$, and (c) $\rho_X \approx 10^5$.

$$h_{e,\Omega} = \left(\frac{\text{Vol}(\Omega)}{N} \right)^{1/d}.$$

Suppose we want to design an adaptive PHS+Poly RBF-FD algorithm to approximate a k -th order differential operator \mathcal{L} with user-specified g -th order convergence. The approximation error is then expected to behave like $\varepsilon \sim h_{e,\Omega}^g$. Moving to highly nonuniform nodes with large mesh ratio, we define global convergence with respect to $h_{e,\Omega}$ by formulating an adaptive RBF-FD approach with adaptive polynomial refinement.

For a general data set $X \subset \Omega$ with no constraint on its mesh ratio, let $X_c \subset X$ denote the stencil associated with the point of interest $\mathbf{x}^c \in X$, and $\Omega_c = \text{conv}(X_c)$ be its associated local sub-domain. Also let $\Omega_c := \text{conv}(X_c)$ be its local sub-domain defined by convex hulls and h_{X,Ω_c} be the associated local fill distance. If we employ a g -th order RBF-FD scheme at this center, then the local error ε_c will reduce as

$$\varepsilon_c \sim h_X^g = (\rho_X q_X)^g \sim \rho_X^g N^{-g/d}.$$

For uniform node with $\rho_X = 1$, the last factor $N^{-g/d} \sim h_e^g$ is exactly what we expect from a g -th order finite difference scheme. In cases when $\rho_X \gg 1$, the asymptotic convergence rate remains order- g but with a large leading constant ρ_X^g that reduces the final accuracy in the error bound. To sum up, the large mesh ratio of highly nonuniform nodes weakens the approximation accuracy in terms of magnitude, but not order of convergence.

Since the (local) convergence order by PHS+Poly RBF-FD algorithm is decided by the polynomial order and (local) fill distance h_{X,Ω_c} , we can obtain g -th convergence by adaptively varying the degree p_c of augmented polynomial order that satisfies

$$h_{X,\Omega_c}^{p_c-k+1} = h_{e,\Omega}^g. \quad (3.2)$$

In this context, non-adaptive standard approach takes $p_c - k + 1 = g$ for all centers \mathbf{x}^c . Note that, by the definition of fill distance, at least one of these local fill distance h_{X,Ω_c} must coincide with the global fill distance $h_{X,\Omega}$, that acts as an upper bound. This particular stencil is the limiting factor for the final accuracy.

Given the equispaced fill distance $h_{e,\Omega}$ (or $\text{Vol}(\Omega)$ and N), the local fill distance h_{X,Ω_i} for each point of interest, and the order k of the differential operator \mathcal{L} , the local polynomial orders for g -th convergence can be determined from (3.2) as

$$\mathbb{N} \ni p_c \geq g \frac{\log_{10} h_{e,\Omega}}{\log_{10} h_{X,\Omega_c}} + k - 1. \quad (3.3)$$

For quasi-random nodes with small mesh ratio, we expect $h_{X,\Omega_i} \approx h_{e,\Omega}$ and p_c in (3.3) simplifies to

$$p = g + k - 1 \quad \text{or} \quad g = p - k + 1. \quad (3.4)$$

Our adaptive scheme aligns with existing non-adaptive methods [2, 6] in this case. By adaptively determining p_c from (3.3), we can achieve better accuracy for highly nonuniform nodes.

The stencil size n_s also impacts the accuracy and convergence of RBF-FD. Studies have shown that when the polynomial order is sufficiently high, increasing n_s beyond the number of polynomial basis functions n_p does not significantly improve accuracy or convergence order, see [2, 6]. For lower-order polynomials, increasing n_s can improve accuracy but convergence order remains unchanged. This independence on stencil size is important for our adaptive scheme. In denser regions, we can use a smaller polynomial degree (and n_s) without significant losses. A stencil size of $n_s = n_p$ is suggested in [2, 6] for distributions with ghost nodes outside the boundary to avoid stagnation error. Using PHS only without appended polynomials in RBF-FD, stagnation errors stem mainly from boundary errors. However, with polynomial augmentation and $n_s \geq 2n_p$, RBF-FD remains free from stagnation error.

Based on these findings, we propose: after computing p_c from (3.3) for a user-specified order of convergence g , we use $n_s = 2n_p + 1$ as suggested in [2, 6] to ensure the solvability and the stability of our algorithm. Since oversized stencil size does not improve accuracy, a smaller n_s can be used in regions with smaller h_{X,Ω_i} without significantly affecting accuracy or stability. Fig. 3 illustrates our proposed adaptive PHS+poly RBF-FD scheme. Since the computational cost of the RBF-FD depends on the stencil-size n_s , using smaller local n_s in denser regions can improve efficiency. The resulting FD-differentiation matrix also enjoys higher sparsity. Algorithm 3.1 summarizes the RBF-FD weight computation using our adaptive PHS+Poly RBF-FD scheme.

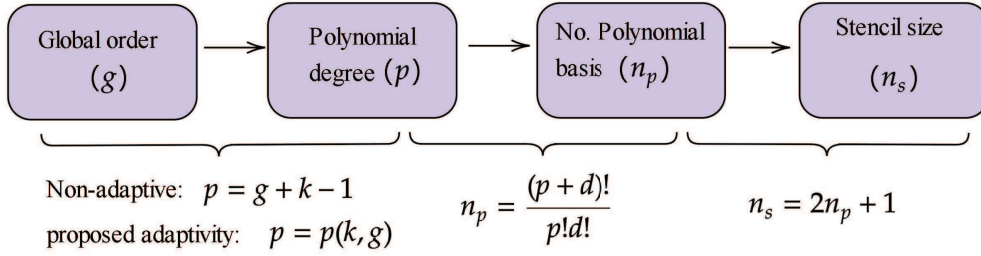


Figure 3: The process of achieving a global order of convergence in an PHS+Poly RBF-FD formulation. The order of convergence g input by the user. The algorithm decides the degree of polynomial to be augmented, which also gives the number of polynomial terms required (n_p) and consequently the stencil size (n_s).

Algorithm 3.1 Adaptive PHS+Poly RBF-FD Method

- 1: **procedure** INPUT($g \in \mathbb{N}$, $X \subset \Omega \subset \mathbb{R}^d$) \triangleright Input expected order and nodes.
- 2: Compute total number of nodes $N = |X|$.
- 3: Compute $h_{e,\Omega} = \left(\frac{\text{Vol}(\Omega)}{N}\right)^{1/d}$.
- 4: Determine maximum stencil size $n_{s,\max}$.
- 5: **for** each point of interest x_c **do**
- 6: Find $n_{s,\max}$ – nearest neighbors for each x_c . \triangleright We have used knn-search.
- 7: Find h_{X,Ω_c} . \triangleright Estimate local fill-distance.
- 8: $p_i = \lceil g \frac{\log_{10} h_{e,\Omega}}{\log_{10} h_{X,\Omega_c}} + k - 1 \rceil$. \triangleright Polynomial degree required to maintain order g .


```

9:       $n_p = C(p + d, d) = \frac{(p+d)!}{p!d!}$ . ▷ Number of polynomial basis.
10:      $n_s = 2n_p + 1$ . ▷ Stencil size just enough to support the polynomial  $p$ .
11:      $r = \sqrt{(x^2 + y^2)}$ .
12:      $A0 = r^m$ . ▷ The RBF interpolant.
13:      $RHS0 = [rhs_L]$ . ▷ The RHS column in Eq. (2.1).
14:      $A = A0 + Poly(p)$ . ▷ The RBF interpolant after augmenting polynomial of degree  $p$ .
15:      $RHS = RHS0 + Poly(p)$ . ▷ Adding corresponding polynomial of degree  $p$ .
16:      $w_L = A/RHS$ . ▷ Solve the linear system for weights.
17:      $w_L = w(1 : n_s)$ . ▷ Ignore the weights corresponding to polynomial augmentation.
18:     end
19: end

```

4. Numerical experiments

This section presents four numerical experiments demonstrating the performance of the proposed adaptive method. First, illustrative examples show the key concepts and behavior of the adaptive method. Then, a benchmark problem evaluates accuracy and convergence. Finally, real-world applications demonstrate effectiveness for both steady and time-dependent problems, showing the method's robustness, accuracy, and adaptivity.

4.1. Example 1. Performance of the proposed adaptive method

In the first test, we investigate the proposed adaptive PHS+Poly RBF-FD scheme by solving an elliptic partial differential equation in domain $\Omega = [-1, 1]^2$ with boundary $\Gamma_1 : \{(x, y) \mid -1 \leq x \leq 1, y = 1\}$ and $\Gamma_2 := \partial\Omega \setminus \Gamma_1$. The problem is set up as follows: Find $u(x, y)$ such that

$$-\nabla^2 u(x, y) = f(x, y), \quad (x, y) \in \Omega,$$

subject to boundary conditions

$$\begin{aligned} \frac{\partial u(x, y)}{\partial \mathbf{n}} &= \cos(x^2 + y), \quad (x, y) \in \Gamma_1, \\ u(x, y) &= \sin(x^2 + y), \quad (x, y) \in \Gamma_2. \end{aligned}$$

The analytical solution and the source term are given, respectively, as

$$u(x, y) = \sin(x^2 + y), \quad f(x, y) = -2 \cos(x^2 + y) - (4x^2 + 1) \sin(x^2 + y).$$

We test two quasi-uniform point distributions with $N = 2500$ under two mesh ratio, see Fig. 4. Nonuniform distributed nodes with large mesh ratio are obtained by the transformation $z = \sin((\pi z)/2)$ in all coordinates.

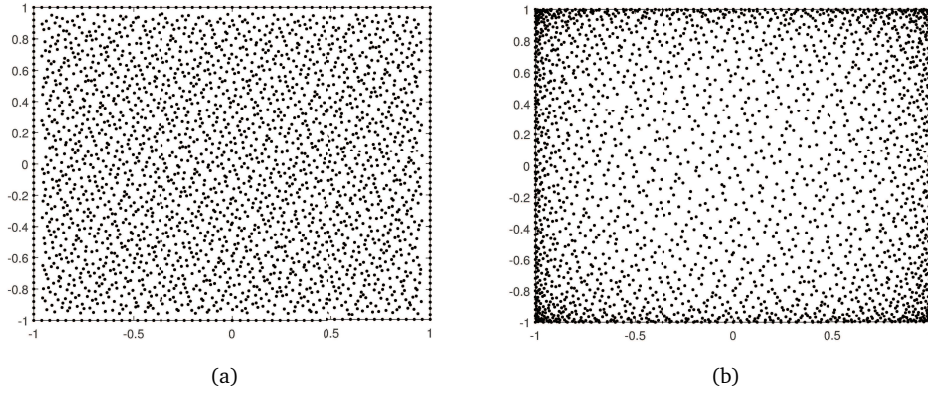


Figure 4: Example 1. In $\Omega = [-1, 1]^2$, setting $N = 2500$ (a) Halton points, and (b) nonuniform nodes with large mesh ratio by sine-transform.

For a target convergence rate of $g = 3$, Figs. 5(a) and 5(b) show the sparsity patterns and eigenvalues of \mathcal{L} of the globally assembled differentiation matrix for uniform nodes and Figs. 5(c) and 5(d) for nonuniform nodes. Recall that the adaptive method is equivalent to the standard RBF-FD on uniform node distribution. From Figs. 5(a)

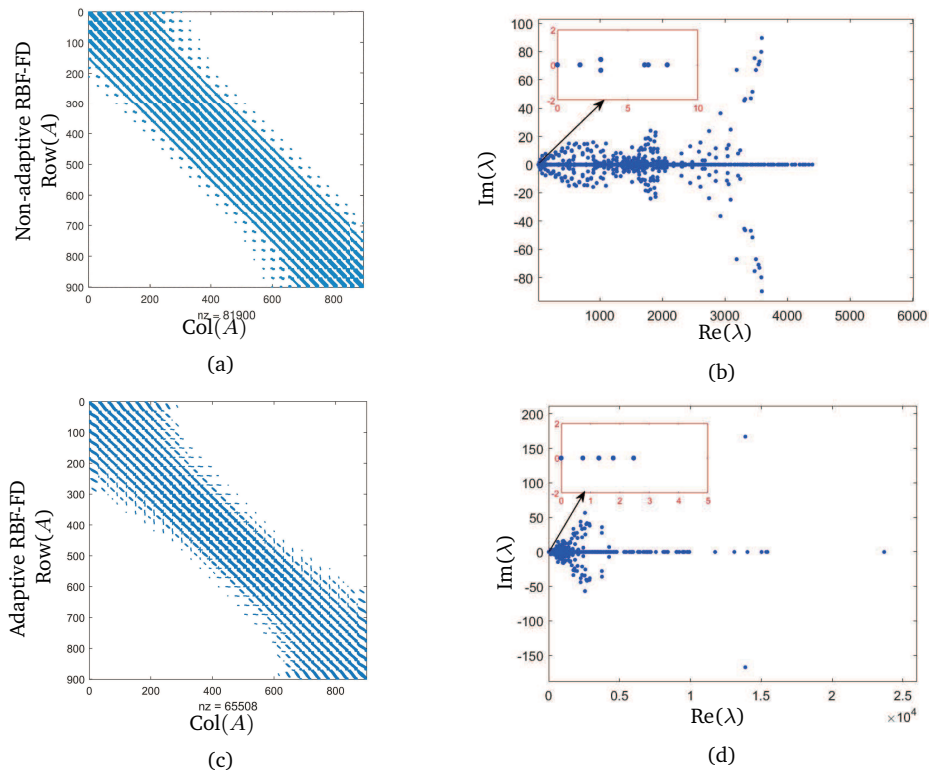


Figure 5: Example 1. Results by adaptive PHS+Poly RBF-FD method for uniform and nonuniform node distribution by $N = 900$, (a,c): Sparsity pattern and (b,d): eigenvalue spectra of the system matrix.

and 5(c), it can be seen that the adaptive method has a smaller bandwidth in the range [12, 15] (except near boundaries) compared to the fixed bandwidth [15] for standard RBF-FD. The total nonzero elements are 6.5×10^4 , an 20% reduction. From Figs. 5(b) and 5(d), the real parts of the eigenvalues computed from the differentiation matrix are all non negative, showing that stable approximations can be obtained by our adaptive approach with higher sparsity.

Fig. 6 shows the convergence profile of ℓ^∞ -error

$$\ell^\infty\text{-error} = \max_{1 \leq i \leq N} |u(x_i) - \hat{u}(x_i)|,$$

versus $N^{1/d}$ for different target orders of convergence $g = 3, 5, 7$ with N and d being the number of nodes in the domain and dimension of the problem. The adaptive RBF-FD gives higher accuracy while maintaining a similar (desired) convergence rate and at the same time, it leads to a sparser system matrix. For $g = 7$ and $N = 2500$, the adaptive method has 1.72×10^5 nonzeros versus 2.14×10^5 for standard RBF-FD. From Figs. 5 and 6, the proposed method is cheaper and more accurate for the example.

Table 2 shows the frequency of polynomial degree when our adaptive scheme is applied to larger data sets with $N = 6241$. For uniform nodes, $p = g + 1 = 8$ were used at all centers. For non-uniform nodes, p ranges from 5 to 8 based on local node density. Hence, the proposed adaptive PHS+Poly RBF-FD scheme achieves higher accuracy at a lower computational cost by adaptive selection of augmented polynomial degree p_c .

What is more, we also extend the adaptive method to solve three dimensional problems. The Poisson equation $-\nabla^2 u(x, y, z) = f(x, y, z)$ is solved in domain $\Omega = [-1, 1]^3$. The exact solution is set as $u^*(x, y, z) = \sin(\pi x/2) \cos(\pi y/2) \cos(\pi z/2)$. The Dirichlet boundary condition is imposed on the $\partial\Omega$. Fig. 7 shows interior nodes of $N = 42875$

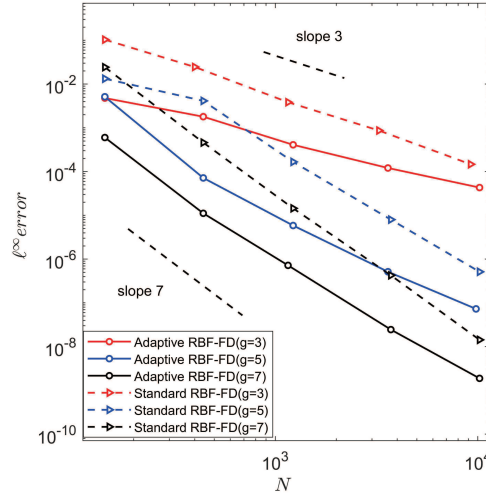


Figure 6: Example 1. Convergence comparison of the adaptive PHS+Poly RBF-FD algorithm with nonuniform nodes and standard RBF-FD with uniform nodes.

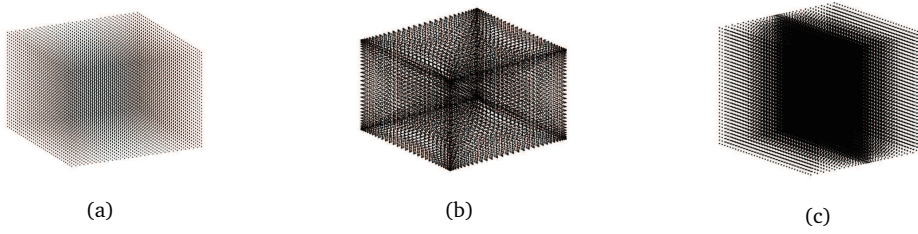


Figure 7: Example 1. For 3D case, in $\Omega = [-1, 1]^3$ with $N = 42875$ (a) uniform node distribution, (b) non-uniform distributed nodes with large mesh ratio generated by sine-transform, and (c) non-uniform nodes with large mesh ratio generated by signed-squared transform at y axis.

Table 2: Example 1. The frequency of polynomial degree by the adaptive PHS+Poly RBF-FD algorithm for uniform and nonuniform node distribution in Fig. 4(b).

Polynomial degree (p_i)	5	6	7	8
Non-adaptive RBF-FD	0	0	0	6241
Proposed adaptive RBF-FD	110	1727	2694	1710

Table 3: Example 1. For 3D case, fixed convergence rate $g = 4$ and node number $N = 13824$, the frequency of polynomial degree (p_i) by the adaptive RBF-FD algorithm for uniform and non-uniform node distributions in Fig. 7.

Polynomial degree (p_i)	3	4	5
RBF-FD	0	0	13824
Adaptive RBF-FD(sine transform)	800	11984	1040
Adaptive RBF-FD(signed squared transform)	0	5776	8048

by uniform points (a) and non-uniform points (b) and (c) which generated by sine-transformation in all directions and sign-transformation $z = \text{sign}(z) * z^2$ along y axis respectively. Fig. 8 shows the convergence behavior of the RBF-FD with uniform nodes and the adaptive RBF-FD with non-uniform nodes shown in Fig. 7 by setting the global order $g \in \{2, 4\}$. It can be seen that both the adaptive RBF-FD and standard RBF-FD can obtain desired convergence order in 3D cases. For the global convergence rate $g = 4$, Table 3 presents frequency of the three-dimensional polynomial degree (p_i) by the standard and the adaptive RBF-FD methods under two kinds of large mesh ratio node distribution respectively. It shows again that the adaptive method can have stable solutions with low computational cost by adaptively selecting the augmented polynomial basis.

4.2. Example 2. Benchmark test for adaptive FEM

This example considers a benchmark test problem from US National Institute for Standards and Technology (NIST) [18] for adaptive finite element method algorithms. It is a Poisson equation $-\Delta u(x, y) - f(x, y) = 0$ in $\Omega = [0, 1]^2$ with exact solution being

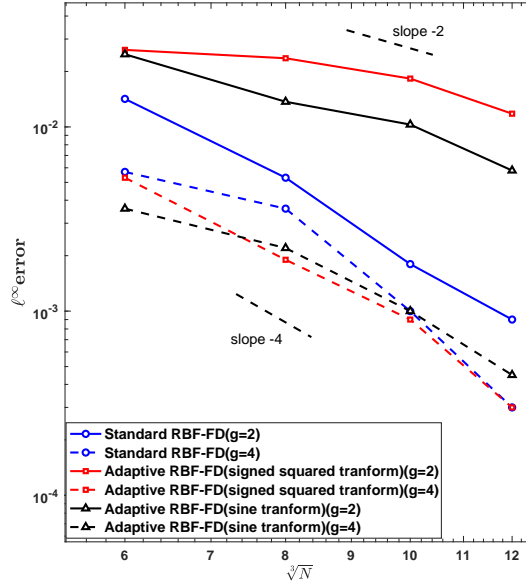


Figure 8: Example 1. For 3D case, convergence comparison of RBF-FD with uniform nodes and the adaptive RBF-FD with non-uniform nodes by $g \in \{2, 4\}$.

an exponential peak

$$u(x, y) = \exp(-\alpha((x - x_0)^2 + (y - y_0)^2)),$$

where (x_0, y_0) is the peak location and α controls the peak strength. Dirichlet boundary conditions matching the exact solution are imposed. A typical value $\alpha = 1000$ is suggested for the test. The right hand side function $f(x, y)$ satisfying the exact solution is

$$f(x, y) = -4(\alpha^2(x - x_0)^2 + \alpha^2(y - y_0)^2) \exp(-\alpha((x - x_0)^2 + (y - y_0)^2)).$$

The reference solution with $\alpha = 1000$ is shown in the Fig. 9.

Firstly, we solve the problem with standard RBF-FD approach over quasi-uniform nodes in the domain and a fixed polynomial degree $p = 6$. Figs. 10(a), 10(b) and 10(c) show results for a quasi-uniform set of $N = 2470$ nodes: nodes, numerical solution, and error function. The error is largest near the peak, indicating inaccuracy there.

We use the NodeLab algorithm [8, 16] to generate data-dependent nodes with same number $N = 2470$, which is denser in the peak zone, see Fig. 10(d). This adaptive node-distribution enhance the accuracy in the approximation. Figs. 10(e) and 10(f) show the approximated solution and point-wise absolute error with an expected global order $g = 5$. Improved in accuracy around the peak zone is obvious in comparison with standard RBF-FD. By setting convergence rate $g \in \{3, 5, 7\}$, Fig. 11 shows the g -th convergence rate can be obtained by both method, but the proposed adaptive method are more accurate by more than one order of magnitude.

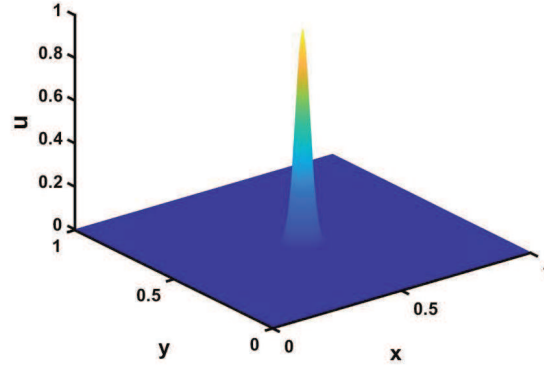


Figure 9: Example 2. A reference solution with $\alpha = 1000$ and $(x_c, y_c) = (0.5, 0.5)$.

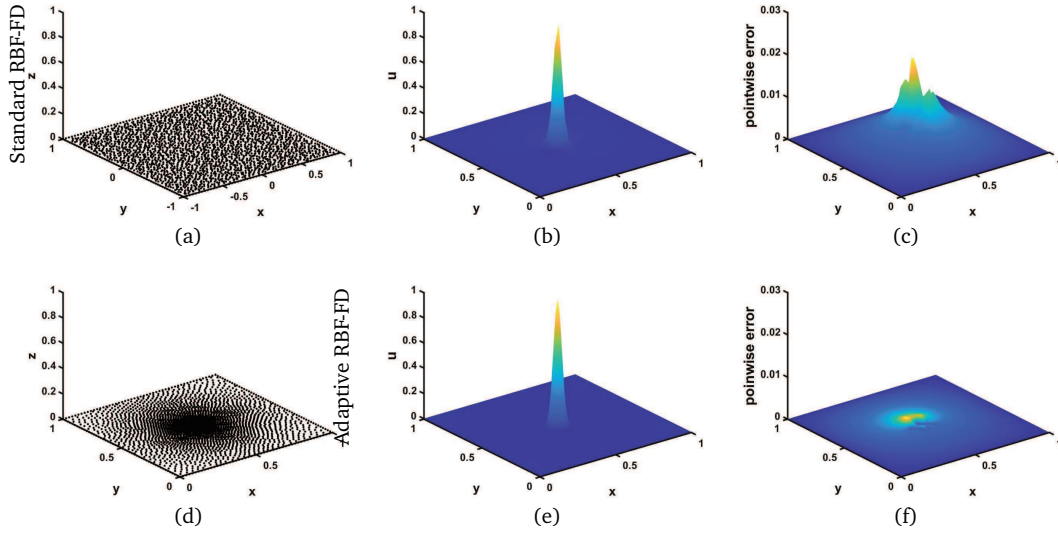


Figure 10: Example 2. Setting $N = 2470, g = 5$, results by adaptive RBF-FD approach under uniform node-distribution with $p = 6$ in first row and under nonuniform nodes with variable polynomial degree in second row, (a, d) node distribution, (b, e) approximated solution, and (c, f) pointwise absolute error.

4.3. Example 3. Solving the heat equation

We consider a time-dependent heat equation incorporated with compatible initial and boundary conditions

$$\begin{cases} u_t - \Delta u = f(x, y, t) & \text{for } (x, y, t) \in \Omega \times (0, T], \\ u(x, y, 0) = g_1(x, y) & \text{for } (x, y) \in \Omega, \\ u(x, y, t) = g_2(x, y, t) & \text{for } (x, y, t) \in \partial\Omega \times (0, T] \end{cases} \quad (4.1)$$

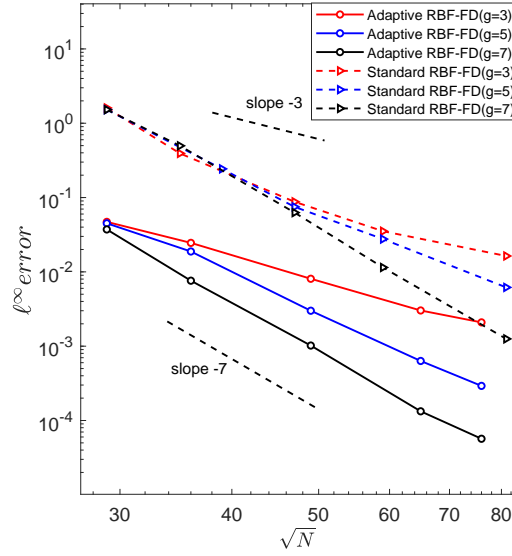


Figure 11: Example 2. Comparison of convergence of the standard RBF-FD and the Adaptive RBF-FD for a Poisson equation with delta-like exact solution.

in $\Omega = [-1, 1]^2$ with exact solution $u^*(x, y, t) = \exp(t) \cos(\pi x) \cos(\pi y)$. The initial condition and boundary conditions are generated from the exact solution

$$g_1(x, y) = \cos(\pi x) \cos(\pi y), \quad g_2(x, y, t) = \exp(t) \cos(\pi x) \cos(\pi y),$$

and so as the source term $f(x, y, t) = (1 + 2\pi^2)u^*(x, y, t)$.

We discretize the PDE (4.1) in time using the θ -method over M time steps with step size $\tau = T/M$

$$\frac{u^n - u^{n-1}}{\tau} = \theta(\Delta u^n + f^n) + (1 - \theta)(\Delta u^{n-1} + f^{n-1}).$$

We consider both the backward-Euler scheme with $\theta = 1$

$$u^n - \tau \Delta u^n = u^{n-1} + \tau f^n(x, y, t),$$

and Crank-Nicolson scheme with $\theta = 1/2$

$$u^n - \frac{\tau}{2} \Delta u^n = u^{n-1} + \frac{\tau}{2} \Delta u^{n-1} + \frac{\tau}{2} \Delta (f^n + f^{n-1}).$$

The nonuniform discrete points in the domain are generated by the same method as in Example 1. The user-defined global convergence rate set to $g = \{3, 5, 7\}$ and PHS kernel is $\phi(r) = r^{2m-1}$, $m = 3$. Fig. 12 illustrates the convergence behavior of the solutions by the $\ell^\infty(\Omega, T)$ error. Fig. 12(a) plots the space convergence results with respect to \sqrt{N} by $\tau = 1E - 7$, $T = 1E - 6$, discrete points in domain nodes $N \in$

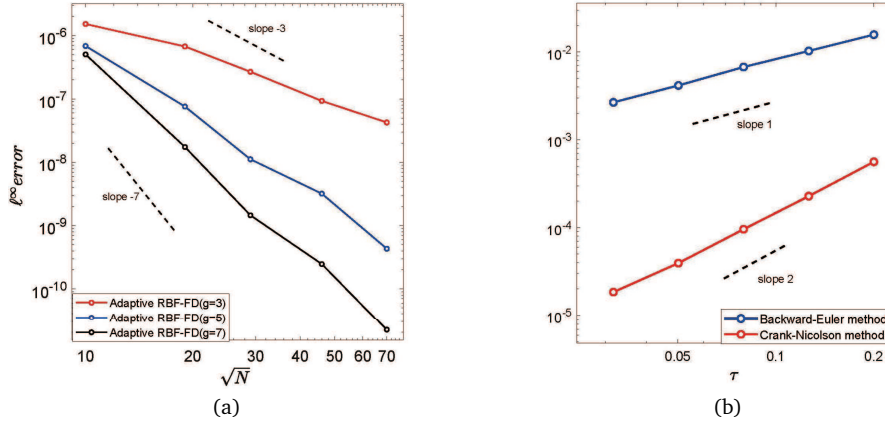


Figure 12: Example 3. The convergence results of adaptive RBFFD (PHS+Poly) for solving the heat equation (4.1): (a) space convergence by $\tau = 1E - 7, T = 1E - 6, N \in [10^2, 70^2]$; (b) time convergence by $\tau \in [0.02, 0.2], T = 1, N = 64^2$.

$[10^2, 70^2]$. The local stencil is set as $n_s = 100$. The result shows that the convergence rate g is obtained. Fig. 12(b) illustrates the time convergence results by $T = 1, N = 64^2$. It can be observed that the Crank-Nicolson method achieves second-order accuracy in time, while the backward-Euler method achieves first-order accuracy in time. These results suggest that the proposed adaptivity does not cause instability when solving parabolic equations.

4.4. Example 4. Application in elastic wave model

We apply the adaptive RBF-FD method to the following elastic wave Navier equation [31]:

$$\begin{cases} \mu \Delta u_1 + (\lambda + \mu)(\partial_x^2 u_1 + \partial_x \partial_y u_2) + \omega^2 \rho u_1 = f_1, \\ \mu \Delta u_2 + (\lambda + \mu)(\partial_x \partial_y u_1 + \partial_y^2 u_2) + \omega^2 \rho u_2 = f_2 \end{cases} \quad (4.2)$$

with ω being the angular frequency and λ, μ being Lamé constants. We assume the medium is homogeneous and set the density $\rho \equiv 1$ after normalization. The time-harmonic displacement vector $\mathbf{u} = (u_1, u_2)^T$ satisfies the lowest-order absorbing boundary condition

$$\mathbf{T}^{(n)}(\mathbf{u}) - i\eta \mathbf{u} = \mathbf{g} \quad \text{on } \gamma = \partial\Omega,$$

where the traction operator $\mathbf{T}^{(n)}$ is

$$\mathbf{T}^{(n)}(\mathbf{u}) = 2\mu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} + \lambda \mathbf{n} \nabla \cdot \mathbf{u} + \mu \mathbf{n} \times (\nabla \times \mathbf{u})$$

with \mathbf{n} being the unit normal vector and a real-valued matrix function η is defined by

$$\eta = \omega \rho (C_P \mathbf{n} \mathbf{n}^T + C_S \mathbf{s} \mathbf{s}^T), \quad C_P = \sqrt{\frac{\lambda + 2\mu}{\rho}}, \quad C_S = \sqrt{\frac{\mu}{\rho}}$$

with \mathbf{s} being a unit tangent vector. We study the problem in $\Omega = [0, 1]^2$, and nonuniform distribution nodes are obtained by using the transformation $z = (\sin(\pi z) + 1)/2$ on each coordinate. The components of the exact displacement field $\mathbf{u} = (u_1, u_2)^T$ is given as follows:

$$\begin{cases} u_1 = \alpha_S \left\{ \exp(-\alpha_S y) - \frac{2\kappa_R^2}{\kappa_R^2 + \alpha_S^2} \exp(-\alpha_P y) \right\} \exp(i\kappa_R x), \\ u_2 = i\kappa_R \left\{ \exp(-\alpha_S y) - \frac{2\alpha_P \alpha_S}{\kappa_R^2 + \alpha_S^2} \exp(-\alpha_P y) \right\} \exp(i\kappa_R x), \end{cases} \quad (4.3)$$

where $\alpha_P = \sqrt{\kappa_R^2 - \kappa_P^2}$, $\alpha_S = \sqrt{\kappa_R^2 - \kappa_S^2}$. We take the Lamé constants $\lambda = 2$, $\mu = 1$, angular frequency $\omega = 2\pi$, which decide the value of $\kappa_R = 2.14\pi$, $\kappa_P = \pi$, $\kappa_S = 2\pi$. We report the relative L^2 -error

$$E_\eta = \frac{\|\mathbf{u}^e - \mathbf{u}^n\|_2}{\|\mathbf{u}^e\|_2} = \left(\frac{\int_\Omega (\bar{\mathbf{u}}^e - \bar{\mathbf{u}}^n)^T (\mathbf{u}^e - \mathbf{u}^n) d\Omega}{\int_\Omega (\bar{\mathbf{u}}^e)^T (\mathbf{u}^e) d\Omega} \right)^{1/2}.$$

Setting the global convergence order $g \in \{3, 5, 7\}$, Fig. 13(a) compares the proposed adaptive PHS+Poly RBF-FD on nonuniform nodes and standard RBF-FD on uniform nodes. The adaptive method achieves higher accuracy while maintaining the desired convergence rate.

Angular frequency ω determines oscillatory behavior: higher ω causes more frequent oscillations. Fig. 13(b) plots error E_η versus ω for $g = 5$. Errors progressively worsen with increasing ω , but the adaptive RBF-FD is less sensitive. Fig. 14 plots the real and imaginary parts of u_1, u_2 at circle $r = 0.5$ for $N = 3600$. Both methods recover the analytic solutions well. For $g = 5$, standard RBF-FD has 8.2×10^5 nonzeros while adaptive RBF-FD has 6.5×10^5 , a reduction of 1.7×10^5 nonzeros. For $g = 7$, adaptive

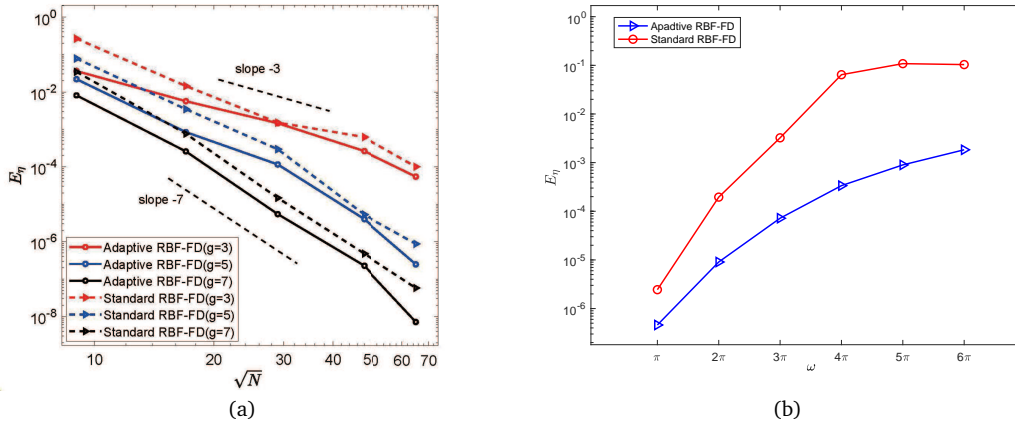


Figure 13: Example 4. (a) Comparison of convergence of the standard RBF-FD and the Adaptive RBF-FD for Navier equation. (b) The relative errors with respect the angular frequencies.

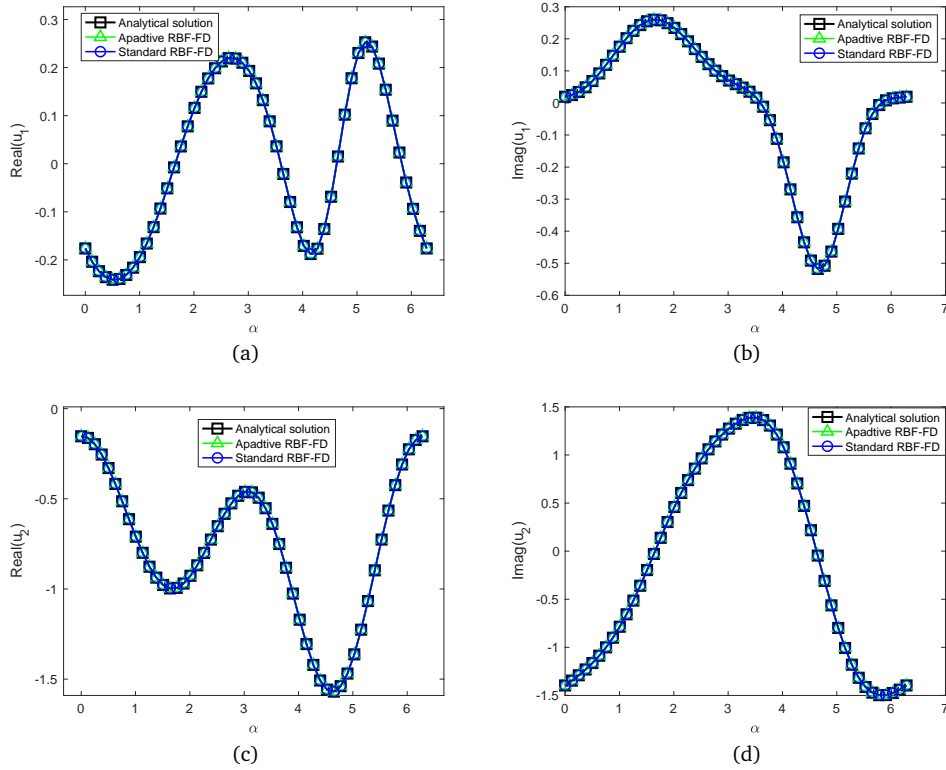


Figure 14: Example 4. Results by adaptive RBF-FD for the Navier equation with $\omega = \pi$ and $N = 3600$.

RBF-FD reduces nonzeros by 2.3×10^5 . The adaptive method saves more computation cost at higher accuracy. For realistic elastic wave problems, the adaptive RBF-FD could be an efficient alternative for practitioners.

5. Conclusions

We developed an adaptive PHS+Poly RBF-FD method well suited for high-order approximations over nonuniform nodes. For uniform nodes, adaptive RBF-FD simplifies to standard RBF-FD. By varying the polynomial degree at each point based on the desired global convergence order, we constructed the adaptive scheme. Our adaptive method provides several advantages: it achieves the specified global convergence order and high accuracy for nonuniform nodes while performing comparably to standard RBF-FD for uniform nodes. The differentiation matrix is sparser, suggesting lower computational cost that will increase for large-scale problems solved using sparse linear solvers. By adaptively choosing local stencil sizes and polynomial degrees based on node density, the proposed method saves the computational resources by balancing accuracy and efficiency. In other words, the proposed method is not aimed to improve accuracy. The method also applies to both steady and time-dependent problems, as demonstrated for

the heat equation and elastic wave problems. These characteristics make the adaptive RBF-FD method a powerful tool for solving PDEs using scattered data. Future work will explore applications to large-scale and 3D problems to fully leverage the computational efficiency gains of the adaptive method.

Acknowledgments

This work was funded by the Hong Kong Research Grant Council GRF (Grants 12301520, 12301021, 12300922) and by the National Science Foundation of China (Grant 12201449).

References

- [1] V. BAYONA, N. FLYER, AND B. FORNBERG, *On the role of polynomials in RBF-FD approximations: III. Behavior near domain boundaries*, J. Comput. Phys. 380 (2019), 378–399.
- [2] V. BAYONA, N. FLYER, B. FORNBERG, AND 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.
- [3] V. BAYONA, M. MOSCOSO, M. CARRETERO, AND M. KINDELAN, *RBF-FD formulas and convergence properties*, J. Comput. Phys. 229 (2010), 8281–8295.
- [4] G. CHANDHINI AND Y. SANYASIRAJU, *Local RBF-FD solutions for steady convection-diffusion problems*, Internat. J. Numer. Methods Engrg. 72 (2007), 352–378.
- [5] P. P. CHINCHAPATNAM, K. DJIDJELI, P. NAIR, AND M. TAN, *A compact RBF-FD based meshless method for the incompressible Navier-Stokes equations*, Proc. Inst. Mech. Eng. M 223 (2009), 275–290.
- [6] N. FLYER, B. FORNBERG, V. BAYONA, AND G. A. BARNETT, *On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy*, J. Comput. Phys. 321 (2016), 21–38.
- [7] N. FLYER, E. LEHTO, S. BLAISE, G. B. WRIGHT, AND A. ST-CYR, *A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere*, J. Comput. Phys. 231 (2012), 4078–4095.
- [8] B. FORNBERG AND N. FLYER, *Fast generation of 2-D node distributions for mesh-free PDE discretizations*, Comput. Math. Appl. 69 (2015), 531–544.
- [9] B. FORNBERG AND N. FLYER, *Solving PDEs with radial basis functions*, Acta Numer. 24 (2015), 215–258.
- [10] B. FORNBERG AND E. LEHTO, *Stabilization of RBF-generated finite difference methods for convective PDEs*, J. Comput. Phys. 230 (2011), 2270–2285.
- [11] M. KINDELAN, D. ÁLVAREZ, AND P. GONZALEZ-RODRIGUEZ, *Frequency optimized RBF-FD for wave equations*, J. Comput. Phys. 371 (2018), 564–580.
- [12] B. MARTIN, A. ELSHERBENI, G. E. FASSHAUER, AND M. HADI, *Improved FDTD method around dielectric and PEC interfaces using RBF-FD techniques*, in: Applied Computational Electromagnetics Society Symposium (ACES), 2018 International, IEEE, (2018), 1–2.
- [13] B. MARTIN AND B. FORNBERG, *Using radial basis function-generated finite differences (RBF-FD) to solve heat transfer equilibrium problems in domains with interfaces*, Eng. Anal. Bound. Elem. 79 (2017), 38–48.

- [14] B. MARTIN, B. FORNBERG, AND A. ST-CYR, *Seismic modeling with radial-basis-function-generated finite differences*, *Geophysics* 80 (2015), T137–T146.
- [15] P. MISHRA, S. NATH, G. FASSHAUER, AND M. SEN, *Frequency-domain meshless solver for acoustic wave equation using a stable radial basis-finite difference (RBF-FD) algorithm with hybrid kernels*, in: *SEG Technical Program Expanded Abstracts 2017*, Society of Exploration Geophysicists, (2017), 4022–4027.
- [16] P. K. MISHRA, *NodeLab – a Matlab package for meshfree node-generation and adaptive refinement*, (2019), <https://github.com/pankajkmishra/NodeLab>.
- [17] P. K. MISHRA, G. E. FASSHAUER, M. K. SEN, AND L. LING, *A stabilized radial basis-finite difference (RBF-FD) method with hybrid kernels*, *Comput. Math. Appl.* 77 (2018), 2354–2368.
- [18] W. F. MITCHELL, *A collection of 2D elliptic problems for testing adaptive grid refinement algorithms*, *Appl. Math. Comput.* 220 (2013), 350–364.
- [19] A. PETRAS, L. LING, AND S. J. RUUTH, *An RBF-FD closest point method for solving PDEs on surfaces*, *J. Comput. Phys.* 370 (2018), 43–57.
- [20] L. SANTOS, N. MANZANARES-FILHO, G. MENON, AND E. ABREU, *Comparing RBF-FD approximations based on stabilized Gaussians and on polyharmonic splines with polynomials*, *Internat. J. Numer. Methods Engrg.* 115 (2018), 462–500.
- [21] B. ŠARLER, *A radial basis function collocation approach in computational fluid dynamics*, *CMES - Comput. Model. Eng. Sci.* 7 (2005), 185–194.
- [22] B. ŠARLER AND R. VERTINK, *Meshfree explicit local radial basis function collocation method for diffusion problems*, *Comput. Math. Appl.* 51 (2006), 1269–1282.
- [23] R. SCHABACK, *A computational tool for comparing all linear PDE solvers*, *Adv. Comput. Math.* 41 (2014), 333–355.
- [24] R. SCHABACK, *Error Analysis of Nodal Meshless Methods*, in: *Meshfree Methods for Partial Differential Equations VIII. Lecture Notes in Computational Science and Engineering*, Vol. 115, Springer, 2017.
- [25] V. SHANKAR, *The overlapped radial basis function-finite difference (RBF-FD) method: A generalization of RBF-FD*, *J. Comput. Phys.* 342 (2017), 211–228.
- [26] V. SHANKAR, G. B. WRIGHT, R. M. KIRBY, AND A. L. FOGELSON, *A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction-diffusion equations on surfaces*, *J. Sci. Comput.* 63 (2015), 745–768.
- [27] J. SLAK AND G. KOSEC, *Refined RBF-FD solution of linear elasticity problem*, in: *3rd International Conference on Smart and Sustainable Technologies (SpliTech)*, IEEE, (2018), 1–6.
- [28] A. TOLSTYKH AND D. SHIROBOKOV, *On using radial basis functions in a “finite difference mode” with applications to elasticity problems*, *Comput. Mech.* 33 (2003), 68–79.
- [29] G. B. WRIGHT, *Radial Basis Function Interpolation: Numerical and Analytical Developments*, PhD Thesis, University of Colorado at Boulder, 2003.
- [30] G. B. WRIGHT AND B. FORNBERG, *Scattered node compact finite difference-type formulas generated from radial basis functions*, *J. Comput. Phys.* 212 (2006), 99–123.
- [31] L. YUAN AND Y. LIU, *A Trefftz-discontinuous Galerkin method for time-harmonic elastic wave problems*, *Comput. Appl. Math.* 38 (2019), 1–29.
- [32] S. ZHANG, *Meshless symplectic and multi-symplectic local RBF collocation methods for Hamiltonian PDEs*, *J. Sci. Comput.* 88 (2021), p. 90.