

A new higher-order RBF-FD scheme with optimal variable shape parameter for partial differential equation

Y. L. Ng, K. C. Ng & T. W. H. Sheu

To cite this article: Y. L. Ng, K. C. Ng & T. W. H. Sheu (2019) A new higher-order RBF-FD scheme with optimal variable shape parameter for partial differential equation, Numerical Heat Transfer, Part B: Fundamentals, 75:5, 289-311, DOI: [10.1080/10407790.2019.1627811](https://doi.org/10.1080/10407790.2019.1627811)

To link to this article: <https://doi.org/10.1080/10407790.2019.1627811>



Published online: 13 Jun 2019.



Submit your article to this journal [↗](#)



Article views: 53



View related articles [↗](#)




View Crossmark data [↗](#)



Citing articles: 1 View citing articles [↗](#)



A new higher-order RBF-FD scheme with optimal variable shape parameter for partial differential equation

Y. L. Ng^{a,d} , K. C. Ng^b, and T. W. H. Sheu^{a,c}

^aDepartment of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, Taiwan; ^bSchool of Engineering, Taylor's University, Selangor Darul Ehsan, Malaysia; ^cCenter of Advance Studies in Theoretical Sciences (CASTS), National Taiwan University, Taipei, Taiwan; ^dDepartment of Mechanical Engineering, Universiti Tenaga Nasional, Selangor Darul Ehsan, Malaysia

ABSTRACT

Radial basis functions (RBFs) with multiquadric (MQ) kernel have been commonly used to solve partial differential equation (PDE). The MQ kernel contains a user-defined shape parameter (ε), and the solution accuracy is strongly dependent on the value of this ε . In this study, the MQ-based RBF finite difference (RBF-FD) method is derived in a polynomial form. The optimal value of ε is computed such that the leading error term of the RBF-FD scheme is eliminated to improve the solution accuracy and to accelerate the rate of convergence. The optimal ε is computed by using finite difference (FD) and combined compact differencing (CCD) schemes. From the analyses, the optimal ε is found to vary throughout the domain. Therefore, by using the localized shape parameter, the computed PDE solution accuracy is higher as compared to the RBF-FD scheme which employs a constant value of ε . In general, the solution obtained by using the ε computed from CCD scheme is more accurate, but at a higher computational cost. Nevertheless, the cost-effectiveness study shows that when the number of iterative prediction of ε is limited to two, the present RBF-FD with ε by CCD scheme is as effective as the one using FD scheme.





ARTICLE HISTORY

Received 28 March 2019
Accepted 31 May 2019

1. Introduction

The method of radial basis function (RBF) is an efficient technique in solving multidimensional interpolation problem due to the ease of implementation and use of directionally-independent kernel. In spite of many advantages, a highly ill-conditioned dense matrix needs to be solved especially for the case that involves a large number of nodes. A more practical approach is to use the local method, in which only a fixed number of neighboring nodes is taken into account. Application of this method results in a sparse linear system with smaller conditioning number.

The infinitely smooth RBF kernels such as the Gaussian and multiquadric (MQ) kernels introduced by Hardy [1] are known to give a more accurate solution as compared to the other types of kernels. However, the quality of the computed solution is strongly dependent on the introduced tuning shape parameter, ε , in the infinitely smooth kernels. The shape parameter also gives rise to a singularity problem as $\varepsilon \rightarrow 0$, in which the RBF linear system tends to be ill-conditioned.

CONTACT K. C. Ng  ngkhaiching2000@yahoo.com  School of Engineering, Taylor's University, Taylor's Lakeside Campus, No. 1, Jalan Taylor's, 47500 Subang Jaya, Selangor Darul Ehsan, Malaysia; T.W.H. Sheu  twhsheu@ntu.edu.tw  Department of Engineering Science and Ocean Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, 10617, Taiwan

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/unhb.

When the value of ε is small enough, the solution obtained by the conventional linear solver (e.g. Gauss elimination) may fluctuate drastically.

Fornberg and Wright [2] employed contour-Padé algorithm to derive the RBF interpolating function for any value of ε . The studies by Kindelan et al. [3, 4] treated the RBF matrix as the perturbed singular matrix, and the inverse of the RBF matrix was computed in the form of Laurent expansion. The singularity point at $\varepsilon = 0$ can be removed by expressing the interpolating function in polynomial form; hence, the RBF interpolating function is valid for any value of ε . Both studies found that the RBF interpolating function is converged to a Lagrange polynomial when $\varepsilon \rightarrow 0$. The optimal value of the shape parameter is often found to lie within the unstable region. Hence, one is unable to obtain the optimal solution from the conventional matrix solver, even if the optimal value of the shape parameter is known. Moreover, the optimal shape parameter is found to be problem-dependent, in which the value is unknown beforehand.

The RBF method has been extended and used to solve partial differential equation [5, 6]. This method is commonly known as the RBF-FD method, as the discretization of the derivative term is similar to that of the conventional finite difference. The coefficient of each stencil point is found by approximating the derivative using the RBF function instead of the polynomial as employed in conventional finite difference schemes. The resulting discretization scheme becomes the function of the shape parameter ε . Hence, the solution quality is dependent on the chosen value of ε . Although the use of a constant value of ε could somehow produce accurate results as shown in some studies [7, 8], there is no guarantee that the randomly chosen shape parameter could always improve the solution accuracy.

Optimization of the shape parameter is always a topic of interest within the RBF research community. Several studies have been carried out in this regard. Huang et al. [9] used the arbitrary precision computation in their work to determine the correlation between the value of ε and the solution accuracy. In their study, they solved the RBF problem using 100-digit precision arithmetic to overcome the singularity problem caused by the round-off error in common 16-digit precision arithmetic when ε is small. Based on the obtained numerical results, they derived an error formulation based on the shape parameter and the grid spacing. Bayona et al. [10, 11] derived an error function as an infinite series, which is dependent on the local value of the function and its derivative at the node of interest. They found that the optimal value of ε can be calculated by minimizing the error function. As reported in their study in Bayona et al. [11], the optimal shape parameter does not exist in cases involving a large number of nodes (accuracy deteriorates significantly). On the other hand, in the work of Guo and Jung [12, 13], the optimized shape parameter was found by applying the Taylor series expansion on the discretization scheme. The higher-order derivative terms appeared in the optimal shape parameter were computed by using a polynomial reconstruction method.

In this article, we have developed a RBF-FD formulation based on the MQ kernel for solving partial differential equations. The RBF interpolating function is derived in polynomial form with a limited number of terms. The optimal value of the shape parameter is computed based on the Taylor series expansion of the discretized RBF-FD formula. The optimized shape parameter is nodally dependent and it is calculated such that the leading error term is eliminated to improve the solution accuracy and to accelerate the rate of convergence.

2. RBF-FD formulation

The method of RBF is an efficient technique in solving multidimensional interpolation problem due to the ease of implementation and use of directionally-independent kernel. In spite of many advantages, a highly ill-conditioned dense matrix needs to be solved especially for the case that involves a large number of nodes. A more practical approach is to use the local method, in which

only a fixed number of neighboring nodes is taken into account. Application of this method results in a sparse linear system with smaller conditioning number.

The standard RBF interpolation function can be constructed from N scattered nodes $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ with the function value of $u(\mathbf{x}_i)$ being known at each node. It can be written in the form of:

$$\mathbf{u}(\mathbf{x}) \approx s(\mathbf{x}) = \sum_{i=1}^N \lambda_i \varphi(\mathbf{x} - \mathbf{x}_i), \quad (1)$$

where $\varphi(\mathbf{x} - \mathbf{x}_i)$ is the chosen RBF kernel. λ_i is the interpolation coefficients to be determined by solving a linear system of N equations $\mathbf{A}\boldsymbol{\lambda} = \mathbf{b}$, where $\boldsymbol{\lambda} = [\lambda_1 \lambda_2 \dots \lambda_N]^T$, $\mathbf{b} = [u_1 \ u_2 \ \dots \ u_N]^T$, and the RBF matrix \mathbf{A} is:

$$\mathbf{A} = \begin{bmatrix} \varphi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \varphi(\|\mathbf{x}_1 - \mathbf{x}_2\|) & \cdots & \varphi(\|\mathbf{x}_1 - \mathbf{x}_N\|) \\ \varphi(\|\mathbf{x}_2 - \mathbf{x}_1\|) & \varphi(\|\mathbf{x}_2 - \mathbf{x}_2\|) & \cdots & \varphi(\|\mathbf{x}_2 - \mathbf{x}_N\|) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi(\|\mathbf{x}_N - \mathbf{x}_1\|) & \varphi(\|\mathbf{x}_N - \mathbf{x}_2\|) & \cdots & \varphi(\|\mathbf{x}_N - \mathbf{x}_N\|) \end{bmatrix}. \quad (2)$$

Eq. (1) can also be written in the form of:

$$\mathbf{u}(\mathbf{x}) \approx s(\mathbf{x}) = \sum_{i=1}^N \alpha_i(\mathbf{x} - \mathbf{x}_i) u(\mathbf{x}_i), \quad (3)$$

The finite difference type formula for a derivative at a node, say node \mathbf{x}_k , can be derived by applying the differential operator \mathcal{L} to Eq. (3).

$$\mathcal{L}\mathbf{u}(\mathbf{x}_k) \approx \mathcal{L}s(\mathbf{x}_k) = \sum_{i=1}^N \mathcal{L}\alpha_i(\mathbf{x}_k - \mathbf{x}_i) u(\mathbf{x}_i) = \sum_{i=1}^N w_i u(\mathbf{x}_i), \quad (4)$$

where, $w_i = \mathcal{L}\alpha_i(\mathbf{x}_k - \mathbf{x}_i)$ are the weighted coefficients which can be found by solving the linear system:

$$\mathcal{L}\varphi_j(\mathbf{x}_k) = \sum_{i=1}^N w_i \varphi_j(\mathbf{x}_i), \quad j = 1, 2, \dots, N. \quad (5)$$

In Eqs. (1) and (5), the chosen RBF kernel $\varphi(\mathbf{x})$ is known to affect the accuracy of the solution. The infinitely smooth RBF kernels shown in Table 1 have a user-defined shape parameter ε . When approximating a smooth function, application of a properly chosen value of ε can render a faster rate of convergence than that using the piecewise smooth kernels [14–16]. Among the infinitely smooth RBF kernels, MQ kernel introduced by Hardy [1] is known to give a more accurate solution compared to those obtained from the other kernels [17]. Hence, it is chosen as the kernel in our study.

The determination of the weighted coefficients w_i in Eq. (5) involves solving a linear system of equations $\mathbf{A}\mathbf{w} = \mathbf{f}$, where $A_{i,j} = \varphi_j(\mathbf{x}_i)$ and $f_j = \mathcal{L}\varphi_j(\mathbf{x}_k)$. However, the RBF interpolation matrix \mathbf{A} approaches singularity as $\varepsilon \rightarrow 0$. The solution obtained from a straightforward computation becomes oscillatory even with a small change of ε . In the following section, the RBF-FD scheme is derived in a polynomial form to avoid this oscillatory problem.

Table 1. Some commonly used infinitely smooth RBF kernels.

RBF kernel	$\varphi(r)$
Gaussian	$e^{-\varepsilon r^2}$
Multiquadric (MQ)	$(1 + \varepsilon r^2)^{1/2}$
Inverse Multiquadric (IMQ)	$(1 + \varepsilon r^2)^{-1/2}$
Inverse quadratic (IQ)	$(1 + \varepsilon r^2)^{-1}$

3. Improvement of the accuracy of RBF-FD scheme

3.1. RBF-FD in polynomial form

Several studies have been done to overcome the singularity problem encountered in the RBF interpolation method when $\varepsilon \rightarrow 0$. Fornberg and Wright [2] employed contour-Padé algorithm to circumvent the singularity problem and computed the RBF interpolation function using any value of shape parameter ε . The works by Kindelan et al. [3, 4] computed the Laurent series of the inverse of the RBF interpolation matrix \mathbf{A} analytically. Then, the singularity term can then be removed in the process of constructing an RBF interpolation function. The resulting RBF function $s(\mathbf{x})$ is a power series of the shape parameter ε and the polynomial functions $P_0(\mathbf{x}), P_1(\mathbf{x}), P_2(\mathbf{x}), \dots$, as shown in Eq. (6):

$$u(\mathbf{x}) \approx s(\mathbf{x}) = P_0(\mathbf{x}) + \varepsilon P_1(\mathbf{x}) + \varepsilon^2 P_2(\mathbf{x}) + \dots \tag{6}$$

In one dimensional problem, the RBF interpolation function $s(\mathbf{x})$ derived by using MQ kernel converges to the Lagrange polynomial $P_0(\mathbf{x})$ when $\varepsilon \rightarrow 0$. However, all approaches mentioned above are impractical as they involve heavy computation especially for the case involving a large number of nodes and a high-order singularity.

In this article, our intention is to develop a new finite-difference-like derivative scheme to solve the PDE problem. Hence, the number of nodes involved in the derivative formulation is pre-fixed. For a set of three equally spaced nodes schematically shown in Figure 1, the RBF matrix \mathbf{A} using MQ kernel is as shown in Eq. (7):

$$\mathbf{A} = \begin{bmatrix} 1 & \sqrt{1 + \varepsilon \left(\frac{\Delta x}{\Delta x}\right)^2} & \sqrt{1 + \varepsilon \left(\frac{\Delta x}{\Delta x}\right)^2} \\ \sqrt{1 + \varepsilon \left(\frac{\Delta x}{\Delta x}\right)^2} & 1 & \sqrt{1 + \varepsilon \left(\frac{2\Delta x}{\Delta x}\right)^2} \\ \sqrt{1 + \varepsilon \left(\frac{\Delta x}{\Delta x}\right)^2} & \sqrt{1 + \varepsilon \left(\frac{2\Delta x}{\Delta x}\right)^2} & 1 \end{bmatrix} = \begin{bmatrix} 1 & \sqrt{1 + \varepsilon} & \sqrt{1 + \varepsilon} \\ \sqrt{1 + \varepsilon} & 1 & \sqrt{1 + 4\varepsilon} \\ \sqrt{1 + \varepsilon} & \sqrt{1 + 4\varepsilon} & 1 \end{bmatrix}. \tag{7}$$

The terms $\sqrt{1 + \varepsilon}$ and $\sqrt{1 + 2\varepsilon}$ can be expanded in binomial series as follows:

$$\begin{aligned} \sqrt{1 + \varepsilon} &= 1 + \frac{1}{2}\varepsilon - \frac{1}{8}\varepsilon^2 + \frac{1}{16}\varepsilon^3 - \frac{5}{128}\varepsilon^4 + \dots, \\ \sqrt{1 + 4\varepsilon} &= 1 + \varepsilon - \frac{1}{2}\varepsilon^2 + \frac{1}{2}\varepsilon^3 - \frac{5}{8}\varepsilon^4 + \dots. \end{aligned} \tag{8}$$

By substituting Eq. (8) into Eq. (7), the matrix \mathbf{A} can be expanded as:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} 1 & \sqrt{1 + \varepsilon} & \sqrt{1 + \varepsilon} \\ \sqrt{1 + \varepsilon} & 1 & \sqrt{1 + 4\varepsilon} \\ \sqrt{1 + \varepsilon} & \sqrt{1 + 4\varepsilon} & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + \varepsilon \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & 2 \\ \frac{1}{2} & 2 & 0 \end{bmatrix} \\ &+ \varepsilon^2 \begin{bmatrix} 0 & \frac{-1}{8} & \frac{-1}{8} \\ \frac{-1}{8} & 0 & -2 \\ \frac{-1}{8} & -2 & 0 \end{bmatrix} + \varepsilon^3 \begin{bmatrix} 0 & \frac{1}{16} & \frac{1}{16} \\ \frac{1}{16} & 0 & 4 \\ \frac{1}{16} & 4 & 0 \end{bmatrix} + \varepsilon^4 \begin{bmatrix} 0 & \frac{-5}{128} & \frac{-5}{128} \\ \frac{-5}{128} & 0 & -10 \\ \frac{-5}{128} & -10 & 0 \end{bmatrix} + \dots \end{aligned} \tag{9}$$

The matrix \mathbf{A} in Eq. (9) can be considered as singular perturbed matrix for the small value of shape parameter ε . The Laurent series of the inverse matrix \mathbf{A}^{-1} can be expressed in terms of the

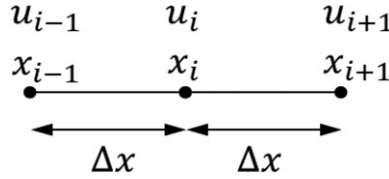


Figure 1. Nodal configuration for Eq. (7).

shape parameter by using the calculation procedure introduced by Kindelan et al. [3, 4] as:

$$\begin{aligned}
 \mathbf{A}^{-1} = & \frac{1}{\varepsilon^2} \begin{bmatrix} -1 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -1 & -1 \\ \frac{1}{2} & -1 & -1 \end{bmatrix} + \frac{1}{\varepsilon} \begin{bmatrix} -3 & \frac{5}{4} & \frac{5}{4} \\ \frac{5}{4} & -3 & -1 \\ \frac{5}{4} & -1 & -3 \end{bmatrix} + \begin{bmatrix} 0 & -\frac{1}{16} & -\frac{1}{16} \\ -\frac{1}{16} & 0 & \frac{1}{2} \\ -\frac{1}{16} & \frac{1}{2} & 0 \end{bmatrix} \\
 & + \varepsilon \begin{bmatrix} -1 & \frac{21}{32} & \frac{21}{32} \\ \frac{21}{32} & -1 & -3 \\ \frac{21}{32} & -3 & -1 \end{bmatrix} + \varepsilon^2 \begin{bmatrix} 3 & -\frac{485}{256} & -\frac{485}{256} \\ -\frac{485}{256} & \frac{3}{4} & \frac{7}{4} \\ -\frac{485}{256} & \frac{7}{4} & \frac{3}{4} \end{bmatrix} + \dots
 \end{aligned} \quad (10)$$

From Eq. (10), the terms $1/\varepsilon^2$ and $1/\varepsilon$ become singular when $\varepsilon \rightarrow 0$. However, these singularity components can be eliminated when the interpolation function is built. The interpolation coefficients λ in Eq. (1) can be obtained by multiplying \mathbf{A}^{-1} in Eq. (10) with the solution vector $\mathbf{b} = [u_1 \ u_2 \ u_3]^T$. The RBF interpolation function in its polynomial form can then be obtained using Eq. (1).

$$\begin{aligned}
 s(x) = & (1-\delta^2)u_i + \left(\frac{\delta}{2} + \frac{\delta^2}{2}\right)u_{i+1} + \left(-\frac{\delta}{2} + \frac{\delta^2}{2}\right)u_{i-1} \\
 & + \varepsilon(\Delta x)^2 \left[(\delta^4 - \delta^2)u_i + \left(\frac{\delta}{4} + \frac{\delta^2}{2} - \frac{\delta^3}{4} - \frac{\delta^4}{2}\right)u_{i+1} + \left(-\frac{\delta}{4} + \frac{\delta^2}{2} + \frac{\delta^3}{4} - \frac{\delta^4}{2}\right)u_{i-1} \right] + \dots
 \end{aligned} \quad (11)$$

The term δ in Eq. (11) is the normalized distance from point i defined as $\delta = (x - x_i)/\Delta x$ with a range of $0 \leq \delta \leq 1$. In Eq. (11), the higher-order terms such as ε^2 , ε^3 , ..., are truncated for simplicity purpose. Both solution accuracy and rate of convergence are not significantly affected by this truncation.

The derivatives can be formulated by applying the respective differential operator to the RBF interpolation function as shown in Eq. (4). For the second derivative term, Eq. (11) is differentiated twice, leading to Eq. (12):

$$s''(x) = \frac{-2u_i + u_{i+1} + u_{i-1}}{(\Delta x)^2} + \varepsilon \left[(-2 + 12\delta^2)u_i + \left(1 - \frac{3\delta}{2} - 6\delta^2\right)u_{i+1} + \left(1 + \frac{3\delta}{2} - 6\delta^2\right)u_{i-1} \right]. \quad (12)$$

By evaluating Eq. (12) at $x = x_i$ ($\delta = 0$), the centered type of differencing scheme is obtained as

$$s''(x_i) = \frac{-2u_i + u_{i+1} + u_{i-1}}{(\Delta x)^2} + \varepsilon[-2u_i + u_{i+1} + u_{i-1}]. \quad (13)$$

The first term on the right-hand side of Eq. (13) represents the conventional second-order central differencing scheme. Hence, for $\varepsilon = 0$, Eq. (13) is equivalent to the finite difference

scheme. The following section shows that there exists an optimal value of ε that can improve the accuracy of Eq. (13).

3.2. Optimal shape parameter with three equidistant nodes

The RBF-FD scheme used to approximate the second derivative shown in Eq. (13) contains a user-defined shape parameter ε that would affect the accuracy of the PDE solution. In the work of [12, 13], Guo and Jung applied the RBF method during the reconstruction process of their ENO scheme. The optimal value of the shape parameter is computed in such a way that it minimizes the leading error term of the RBF-FD scheme. The similar approach is applied to determine the optimal shape parameter based on the leading error terms of the present RBF-FD scheme in Eq. (13). Assuming that the function $s(x)$ is smooth, the Taylor series expansion of $s''(x_i)$ in Eq. (13) has the form of:

$$s''(x_i) = u'' + (\Delta x)^2 \left[\frac{1}{12} u^{(4)} + \varepsilon u'' \right] + (\Delta x)^4 \left[\frac{1}{360} u^{(6)} + \frac{1}{12} \varepsilon u^{(4)} \right] + O(\Delta x^6). \tag{14}$$

From Eq. (14), the value of ε can be chosen in the sense that $u^{(4)}/12 + \varepsilon u'' = 0$ to eliminate the $O(\Delta x^2)$ term. Therefore, the scheme in Eq. (13) can be improved to yield the accuracy of $O(\Delta x^4)$ if the shape parameter is computed as $\varepsilon = -u^{(4)}/12u''$.

The shape parameter of the second derivative expressed in the present work, that is, $\varepsilon = -u^{(4)}/12u''$, is different from that derived by Feng and Duan [18], that is, $\varepsilon = -u^{(4)}/15u''$. The slight difference is due to the variation of the expression of RBF-FD schemes. Nevertheless, both optimal shape parameters are still depending on derivative terms $u^{(4)}$ and u'' .

From the expression of the optimal ε for the scheme outlined in Eq. (13), we are aware that:

1. The optimal value of ε depends on the local derivative terms. This indicates that the optimal ε is not only problem-dependent, but it is also varying throughout the domain.
2. Obtaining the solution for u from RBF-FD by using the optimal ε is indeed a nonlinear process, as the optimal ε becomes the function of the derivative of u to be solved.

If the exact values of derivative terms $u^{(4)}$ and u'' are known, the solution with fourth-order accuracy can be obtained from Eq. (13) without introducing more nodal points. However, derivative terms are often unknown beforehand. Therefore, the problem under current investigation turns out to be a nonlinear problem.

3.3. 1D numerical experiment

An example of a nonhomogeneous boundary value problem in one-dimensional domain is given as:

$$\begin{cases} u'' = 2\pi^2 e^{\pi x} \cos(\pi x), & 0 < x < 1, \\ u = u(x) \text{ at } x = 0, x = 1, \end{cases} \tag{15}$$

where the source term $f(x)$ and boundary condition are derived from the exact solution $u(x) = e^{\pi x} \sin(\pi x)$. The domain is discretized into N segments, and the second derivative term u'' in Eq. (15) is approximated by using the polynomialized RBF-FD as shown in Eq. (13). The optimal value of shape parameter ε can be computed by $\varepsilon = -u^{(4)}/12u''$. In this numerical experiment, the derivative terms appeared in the optimal ε formulation are computed using the exact solution. From the result tabulated in Table 2, the present RBF-FD scheme with optimal ε is fourth order accurate.

The value of optimal ε throughout the domain with grid number $N = 40$ is shown in Figure 2. Obviously, the optimal value of ε varies significantly throughout the domain. To investigate the

Table 2. Result for Eq. (15) by using conventional finite difference scheme and present RBF-FD with the optimal value of the shape parameter. L_∞ denotes the infinite error norms of the solution obtained.

N	L_∞ , Finite difference	Present RBF-FD	
		L_∞ , Present RBF-FD	Order
10	1.041e-2	2.021e-4	–
20	2.704e-3	5.221e-6	5.27
40	6.854e-4	2.724e-7	4.26
80	1.717e-4	1.856e-8	3.87
160	2.716e-5	1.024e-9	4.18

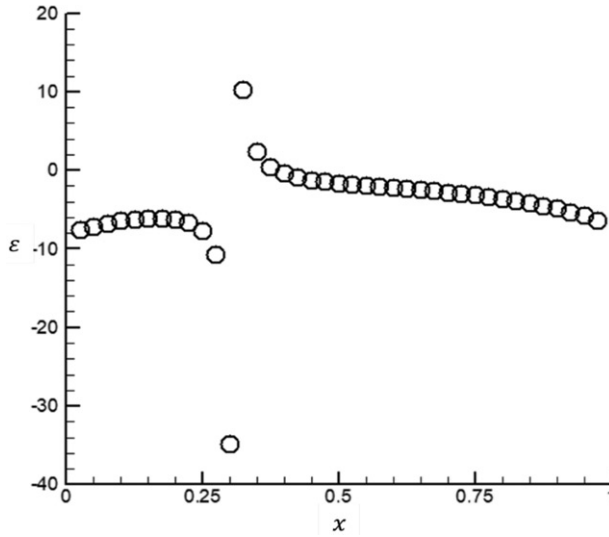


Figure 2. Distribution of the value of optimal ε throughout the domain for Eq. (15).

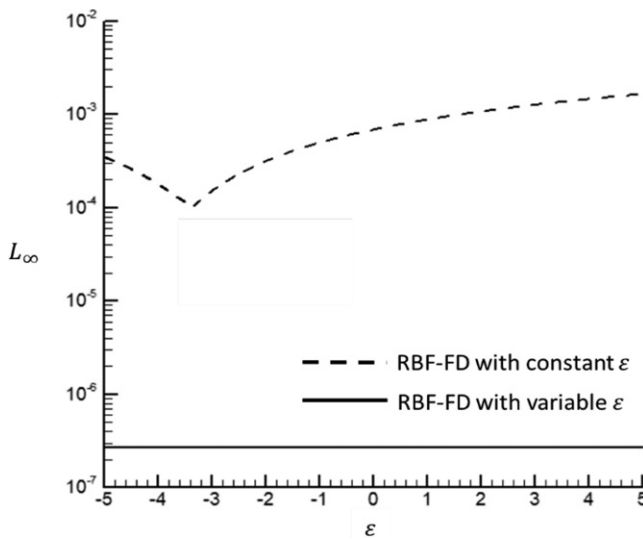


Figure 3. L_∞ error norm of the solution plotted with respect to ε for Eq. (15).

significance of the optimally varying shape parameter on the solution accuracy, Eq. (15) is solved by using the same RBF-FD scheme with different constant ε values for all nodes in the domain. From the L_∞ error norm versus ε plot as shown in Figure 3, the RBF-FD scheme with the optimally varying ε is more accurate than the one using the constant ε (x -axis in Figure 3). Although the error when using $\varepsilon \approx -3.2$ is the lowest, it is still significantly higher than the one that using the optimally varying ε .

3.4. Predicting optimal shape parameter

From the numerical example in Section 3.3, it is known that the accuracy and the rate of convergence can be greatly improved using the optimal value of ε . However, the exact value of optimal ε is a function of derivative terms that are often not known. The optimal value of ε needs to be predicted through different approaches.

In this study, the value of ε is predicted by using two different approaches: low-order finite difference and high-order combined compact differencing (CCD) schemes. The purpose of using two different approaches is to compare the accuracy of the solutions predicted from two different values of ε . The calculation of ε using two different approaches is explained as follows.

3.4.1. Predicting ε using low order finite difference scheme

Taking the optimal $\varepsilon = -u^{(4)}/12u''$ for Eq. (13) as an example, the value of ε at an interior node can be computed explicitly by using a low-order finite difference scheme as in Eq. (16) given below:

$$\varepsilon_{opt} \approx \frac{-u^{(4)}|_{FDM}}{12u''|_{FDM}} = \frac{-(u_{i+2}-4u_{i+1}+6u_i-4u_{i-1}+u_{i-2})}{12\Delta x^2(u_{i+1}-2u_i+u_{i-1})}. \tag{16}$$

In the above, the derivative terms $u^{(4)}|_{FDM}$ and $u''|_{FDM}$ are calculated using the second-order accurate central differencing scheme. The use of nodal points at $i-2$ and $i+2$ will not increase the complexity of the discretized PDE system as they only involve an explicit computation of ε . Therefore, the band width of the PDE matrix remains unchanged.

3.4.2. Predicting ε using high-order combined compact differencing scheme

CCD schemes [19–22] are popular approaches for computing the solution with high accuracy. The first- and second-order derivative terms are obtained simultaneously in compact stencil points. This is in contrast with the compact difference (CD) scheme [23], where the first- and second-order derivative terms need to be obtained separately in a sequential procedure. The computation of the value of ε in the present RBF-FD scheme requires the value of higher-order derivative terms. Hence, we applied the CCD scheme to calculate the values of the first- and second-order derivative terms simultaneously. The third- and fourth-order derivative terms can be explicitly calculated after the first- and second-order derivative terms are known.

In this article, only a problem with Dirichlet boundary condition and uniform grid spacing h is considered for the sake of description of the CCD scheme. The CCD schemes described in [19] for interior nodes are:

$$\frac{7}{16}(u'|_{i+1}+u'|_{i-1})+u'|_i-\frac{h}{16}(u''|_{i+1}-u''|_{i-1})=\frac{15}{16h}(u_{i+1}-u_{i-1}), \tag{17}$$

$$\frac{9}{8h}(u'|_{i+1}-u'|_{i-1})-\frac{1}{8}(u''|_{i+1}+u''|_{i-1})+u''|_i=\frac{3}{h^2}(u_{i+1}-2u_i+u_{i-1}). \tag{18}$$

The schemes in Eqs. (17) and (18) have $O(h^6)$ accuracy. Subjected to the Dirichlet boundary condition, four additional equations at the boundary nodes $i = 1$ and $i = N + 1$ proposed in [19]

are applied.

$$u'|_1 + 2u'|_2 - hu''|_2 = \frac{1}{h}(-3.5u_1 + 4u_2 - 0.5u_3), \quad (19)$$

$$hu''|_1 + 5hu''|_2 - 6u_x|_2 = \frac{1}{h}(9u_1 - 12u_2 + 3u_3), \quad (20)$$

$$u'|_{N+1} + 2u'|_N + hu''|_N = \frac{1}{h}(3.5u_{N+1} - 4u_N + 0.5u_{N-1}), \quad (21)$$

$$hu''|_{N+1} + 5hu''|_N + 6u'|_N = \frac{1}{h}(9u_{N+1} - 12u_N + 3u_{N-1}). \quad (22)$$

Note that the CCD schemes at the boundary nodes shown in Eqs. (19)–(22) have $O(h^4)$ accuracy. In the applied CCD schemes, the values of u' and u'' are obtained after solving the $2N + 2$ equations. Then the higher-order derivative terms such as u''' and $u^{(4)}$ at the interior nodes can be explicitly calculated by using Eqs. (23) and (24):

$$u'''|_i = -\frac{15}{4h^3}(u_{i+1} - u_{i-1}) + \frac{15}{4h^2}(u'|_{i+1} + u'|_{i-1}) - \frac{3}{4h}(u''|_{i+1} - u''|_{i-1}), \quad (23)$$

$$u^{(4)}|_i = -\frac{36}{h^4}(u_{i+1} - 2u_i + u_{i-1}) + \frac{21}{h^3}(u'|_{i+1} - u'|_{i-1}) - \frac{3}{h^2}(u''|_{i+1} + u''|_{i-1}). \quad (24)$$

4. Numerical algorithm

Predicting the value of ε in the present RBF-FD scheme requires the values of the derivative terms at every node. However, both FD and CCD schemes require the discrete values of $u(x)$ when computing the derivative terms. According to Zhang et al. [24], both level of accuracy and ROC of the proposed FD scheme can be improved by introducing a compact correction source term. The procedure involves solving the PDE with conventional FD scheme to obtain a guessed solution, followed by calculating the compact correction source term by using the guessed solution to improve the accuracy of the new solution.

In the present implementation, we also have applied a two-step approach. Firstly, an approximate solution of $u(x)$ is computed by using the conventional FD scheme. Then, the solution is improved using the optimal value of ε predicted from the approximate solution. Knowing the nonlinear nature of the present RBF-FD scheme, the optimal value of ε is recomputed using the refined solution of $u(x)$, and the discretized PDE is solved again using the new predicted value of ε . The procedures are repeated until the solution of $u(x)$ converges. The iterative procedures are summarized below as:

1. An approximate solution of $u(x)$ is computed by using the conventional FD scheme (or consider $\varepsilon = 0$ in present RBF-FD).
2. The values of the local derivative terms u' , u'' , u''' , $u^{(4)}$ at every node are predicted by virtue of FD/CCD scheme using the approximate solution obtained in (1).
3. Re-compute the new RBF-FD coefficients in the discretized equations by using the value of ε evaluated from the value of the local derivative terms obtained in (2).
4. Compute the new solution of $u(x)$ by using the new RBF-FD coefficients found in (3).
5. Continue the iterative procedure until the solution of $u(x)$ is converged.

5. 2D numerical experiments

In the following sections, the proposed numerical algorithm is applied to solve the problems in two-dimensional domains. In this article, the study is limited only to equally spaced nodes. We

will show that a significant improvement in accuracy can be achieved by using the optimal value of ε which is predicted locally from the involved derivative terms. In this section, all PDE matrices are solved by using Bi-CGStab iterative solver with incomplete LUT pre-conditioner. Iterative solver is preferred over direct solver as the solution obtained from the previous iteration can be used as an initial guess to accelerate the convergence.

5.1. Solution of Poisson equation

Consider the problem:

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -f(x, y), & \text{in } \Omega = (0, 1) \times (0, 1), \\ u = u(x, y), & \text{on } \partial\Omega, \end{cases} \tag{25}$$

The solution is approximated on a uniform grid as shown in Figure 4. The discretization of $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$ is obtained by combining the one-dimensional three stencil points scheme for predicting s'' (as in Eq. (13)). This scheme involves stencil points in x (i.e. $(i-1, j), (i, j), (i+1, j)$) and y -directions (i.e. $(i, j-1), (i, j), (i, j+1)$):

$$s''(x_i) = \frac{\partial^2 s}{\partial x^2}|_{i,j} = \frac{-2u_{i,j} + u_{i+1,j} + u_{i-1,j}}{(\Delta x)^2} + \varepsilon[-2u_{i,j} + u_{i+1,j} + u_{i-1,j}], \tag{26}$$

$$s''(y_i) = \frac{\partial^2 s}{\partial y^2}|_{i,j} = \frac{-2u_{i,j} + u_{i,j+1} + u_{i,j-1}}{(\Delta y)^2} + \varepsilon[-2u_{i,j} + u_{i,j+1} + u_{i,j-1}]. \tag{27}$$

For equally spaced grid $\Delta x = \Delta y = h$, the discretization of $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$ using the present RBF-FD scheme is the summation of Eqs. (26) and (27):

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2}|_{i,j} + \frac{\partial^2 u}{\partial y^2}|_{i,j} \approx \frac{\partial^2 s}{\partial x^2}|_{i,j} + \frac{\partial^2 s}{\partial y^2}|_{i,j} &= \frac{-4u_{i,j} + u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}}{h^2} \\ &+ \varepsilon(-4u_{i,j} + u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}). \end{aligned} \tag{28}$$

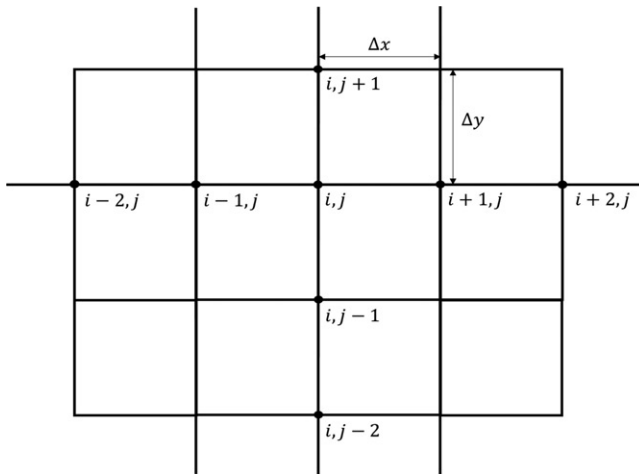


Figure 4. Local uniform grid in present RBF-FD discretization.

Upon conducting the modified equation analysis on Eq. (28) using Taylor’s series, we get

$$\begin{aligned} \frac{\partial^2 s}{\partial x^2}|_{ij} + \frac{\partial^2 s}{\partial y^2}|_{ij} &= \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + h^2 \left(\frac{1}{12} \left(\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4} \right) + \varepsilon \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \right) \\ &+ h^6 \left(\frac{1}{360} \left(\frac{\partial^6 u}{\partial x^6} + \frac{\partial^6 u}{\partial y^6} \right) + \varepsilon \frac{1}{12} \left(\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4} \right) \right) + O(h^6), \end{aligned} \tag{29}$$

Hence, the shape parameter expressed below can be used to eliminate the $O(h^2)$ error term so that the scheme in Eq. (28) becomes fourth order accurate.

$$\varepsilon_{opt} = \frac{-\left(\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4}\right)}{12\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)}. \tag{30}$$

5.1.1. 2D numerical experiment 1

Consider the problem:

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -f(x, y), \text{ in } \Omega = (0, 1) \times (0, 1), \\ u(x, y) = \exp\left(-\left(x - \frac{1}{4}\right)^2 - \left(y - \frac{1}{2}\right)^2\right) \sin(\pi x) \cos(2\pi y) \text{ on } \partial\Omega. \end{cases} \tag{31}$$

The source term $f(x, y)$ is derived from the exact solution $u(x, y) = \exp\left(-\left(x - \frac{1}{4}\right)^2 - \left(y - \frac{1}{2}\right)^2\right) \cos(2\pi y) \sin(\pi x)$. The same problem has been solved in [11, 25, 26].

The calculation begins with the initial value of $\varepsilon = 0$, thus simplified the Eq. (28) to conventional second order central differencing scheme. The optimal value of ε is computed from the initial solution u^0 obtained after solving the PDE in Eq. (31). The coefficients of the PDE matrix are updated with the new value of optimal ε , leading to the new refined solution u^1 . The iterative solution process continues until the solution u becomes converged, that is, when $\max||u^n - u^{n-1}|| < 10^{-6}$.

The infinite error norms of the solution obtained by using the present RBF-FD scheme are shown in Table 3. The range of the computed ε throughout the domain is shown as ε_{min} and

Table 3. Result for the Eq. (31). L_∞ denotes the infinite error norm of the solution obtained; (iters) denotes the number of iterations needed until convergence; ε_{min} and ε_{max} are the minimum and maximum values of the optimal ε .

N	$\varepsilon = \text{FD}$			$\varepsilon = \text{CCD}$		
	L_∞ (iters)	ε_{min}	ε_{max}	L_∞ (iters)	ε_{min}	ε_{max}
10	4.432e-3 (8)	1.55	5.78	2.070e-3 (17)	1.57	6.36
20	1.545e-4 (6)	-54.12	46.13	4.025e-5 (9)	-61.79	39.15
50	4.364e-6 (4)	-50.19	71.49	1.15e-6 (6)	-50.97	69.58
100	2.551e-7 (4)	-5.45e + 3	4.03e + 2	7.426e-8 (6)	-1.74e + 3	1.08e + 4
$\varepsilon = \text{exact solution}$						
N	L_∞	ε_{min}	ε_{max}			
10	5.619e-4	-1.90e + 3	5.82			
20	3.200e-5	-46.58	36.06			
30	8.331e-7	-51.50	68.90			
40	6.266e-8	-3.36e + 5	3.77e + 2			

ε_{max} . From the tabulated result, it is clear that the solution obtained by using the value of ε predicted through the CCD scheme is more accurate. The range of ε values computed by using the CCD scheme is also closer to that of the exact solution.

The plot of L_∞ error norm of the solution obtained with different values of ε for solving Eq. (31) at $N = 50$ is shown in Figure 5. The result obtained by using the conventional FD scheme is plotted in dash-dotted line for comparison purpose. The solutions obtained by using the value of

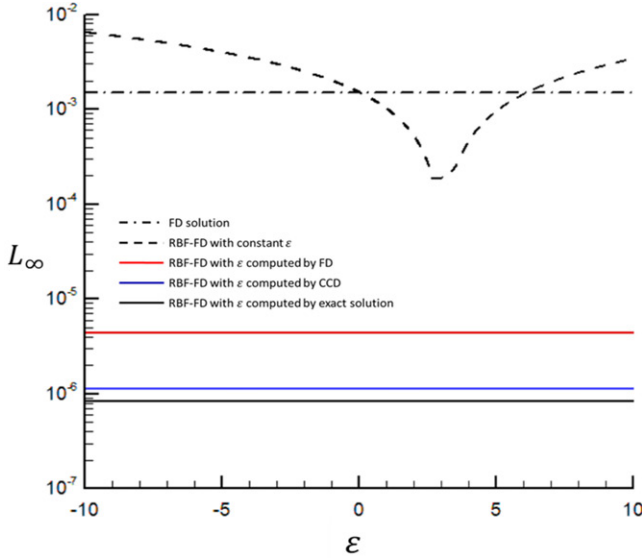


Figure 5. L_∞ error norms of the solution are plotted with respect to ε for the Eq. (31) at $N = 50$.

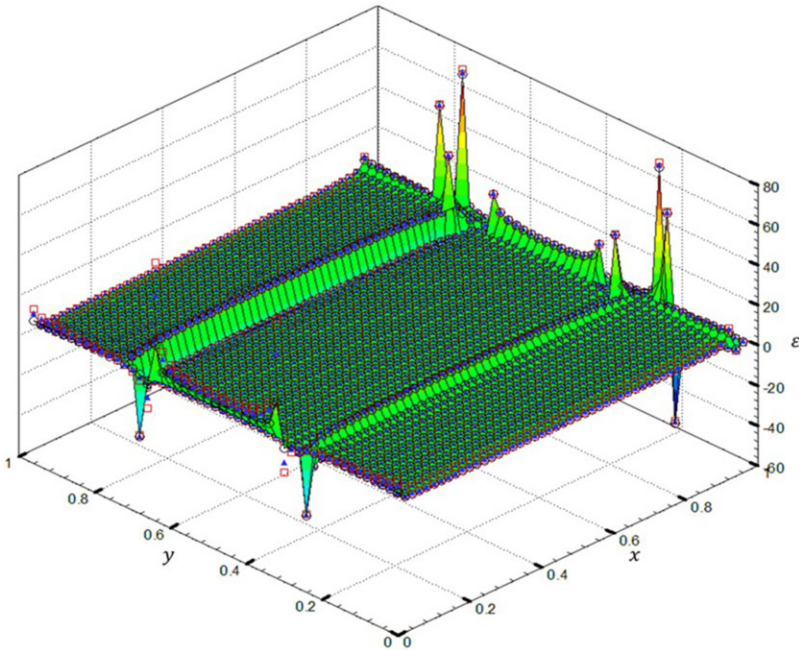


Figure 6. Distribution of the value of ε throughout the domain for the Eq. (31) at $N = 50$. Red square: ε computed by using the FD scheme. Blue delta: ε computed by using CCD scheme. Black circle: ε computed by using the exact solution.

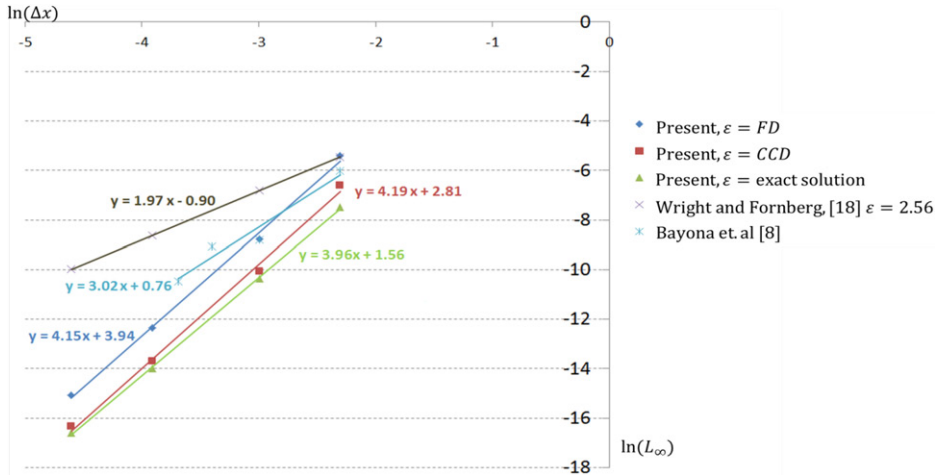


Figure 7. The rate of convergence of the present RBF-FD scheme for the Eq. (31) using the values of ε predicted by using different approaches. The results of [9] and [22] are included to compare with the present RBF-FD result.

variable ε predicted through the FD (red solid line in Figure 5) and the CCD (blue solid line in Figure 5) schemes have shown a very significant improvement in accuracy as compared to the conventional FD method and the RBF-FD scheme by using constant ε (dashed line in Figure 5).

Figure 6 shows the distributions of the values of ε computed by using the FD (red square in Figure 6) and the CCD (blue delta in Figure 6) schemes at $N = 50$. The value of ε computed from the exact solution is shown in black circle on the surface. In general, the distributions of the values of ε predicted by using either of the schemes agree well with the one computed from the exact solution (black circle in Figure 6). Note that there is a discontinuity on the value of ε across the small intervals at $y = 0.25$ and $y = 0.75$. Similar distribution pattern on the value of shape parameter is also observed in [11] by using their error minimization scheme in computing the optimal value of the shape parameter. Unfortunately, a direct comparison cannot be made between our computed shape parameter and the one reported in [11]. This is because the negative value of shape parameter is permitted in the present approach, and an imaginary number will be obtained if we convert our shape parameter ε to their shape parameter c in [11], where $c = 1/\sqrt{\varepsilon}$.

The rate of convergence of the solution obtained by using the present RBF-FD scheme is shown in Figure 7. We also include the results of Bayona et al. [11] generated from their non-constant shape parameter, and the result of Wright and Fornberg in [25] computed from the five-node RBF-FD scheme with $\varepsilon = 2.56$ (our ε is equivalent to their ε^2 , due to different ways of describing the MQ kernel) as this gives the best solution at $N = 10 - 100$ from their tabulated results. From Figure 7, it is found that the scheme in [25] has a ROC a of 2, which is similar to the conventional five-node central differencing scheme. Also, by using the five-node scheme, the scheme in [11] and our RBF-FD scheme improve the ROC to 3 and 4, respectively. This shows that the use of a variable shape parameter is able to eliminate the leading error term in the discretized equation and increase the ROC.

5.1.2. Computational cost evaluation

From the results above, it is obvious that the solution of the present RBF-FD scheme is more accurate than that of the conventional FD scheme at the expense of a higher computational cost. In the present RBF-FD scheme, the optimal value of ε and the coefficients of the PDE matrix are re-computed at every iterative level. The cost-effectiveness of the present RBF-FD scheme is

investigated by comparing the computing time required to reach a predetermined accuracy level against that of the conventional FD scheme.

The computational times of the conventional FD and the present RBF-FD schemes are reported in Table 4 for different grid resolutions. Figure 8 plots the L_∞ error norm against the computational time. From this plot, it is apparent that the present RBF-FD scheme (using ε predicted by either FD or CCD schemes) is more cost-effective than the conventional FD scheme, especially when the grid resolution increases.

Under the same computational time, the RBF-FD scheme with ε predicted by using the FD scheme is more accurate than that by using the CCD scheme. It is noted that CCD scheme requires additional computational effort (higher number of iterations, see Table 3). Also, due to the implicit procedure of CCD scheme, additional matrices should be solved. Hence, albeit the fact that the present RBF-FD scheme with ε predicted by using the CCD scheme gives more accurate solution under the same grid resolution, its computational efficiency is somewhat inferior to that of using the FD scheme.

The effect of iteration number on the solution accuracy is shown in Figure 9. One can see that the enhancement in solution accuracy is very apparent at the 2nd iteration level. Beyond that, the solution accuracy is almost similar. Hence, it is interesting to check if the cost-effectiveness of the

Table 4. L_∞ error norm and the CPU time in solving Eq. (31) by using different schemes at different grid resolutions N .

N	RBF – FD, $\varepsilon =$ FD		RBF – FD, $\varepsilon =$ CCD	
	L_∞	CPU time, T (s)	L_∞ (iters)	CPU time, T (s)
10	4.432e-3	7.40E-02	2.070e-3	4.39E-01
20	1.545e-4	3.42E-01	4.025e-5	1.25E + 00
50	4.364e-6	2.36E + 00	1.150e-6	9.29E + 00
100	2.551e-7	1.19E + 01	7.426e-8	6.67E + 01

N	FD scheme	
	L_∞	CPU time, T (s)
10	5.619e-4	9.00e-03
20	3.200e-5	5.90E-02
50	8.331e-7	6.49e-01
100	6.266e-8	3.41e + 00

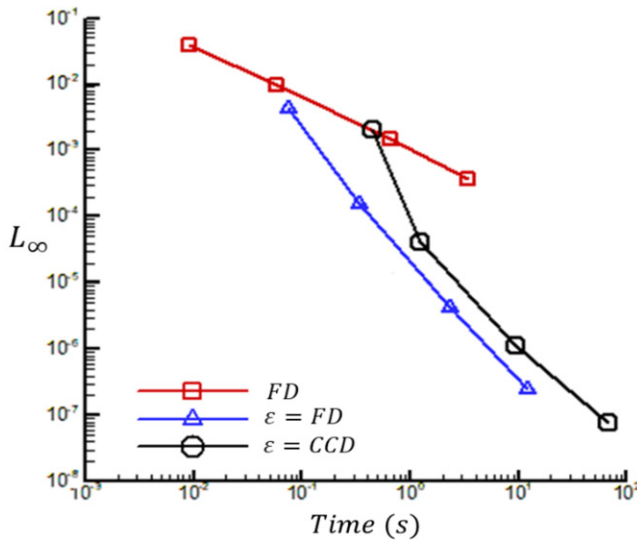


Figure 8. L_∞ error norm versus computational time for different schemes applied to solve Eq. (31).

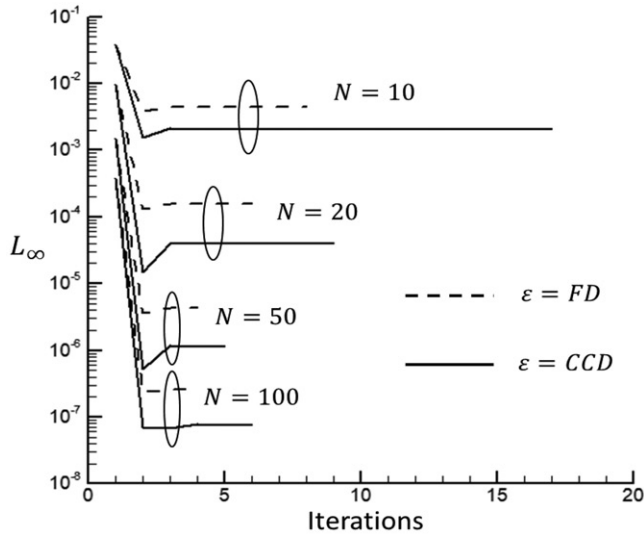


Figure 9. L_∞ error norm versus computational time for different schemes applied to solve Eq. (31).

Table 5. L_∞ error norm and the CPU time in solving Eq. (31) by using the present RBF-FD scheme with only two iterations for different grid resolutions N .

N	RBF - FD, $\varepsilon = FD$		RBF - FD, $\varepsilon = CCD$	
	L_∞	CPU time, T (s)	L_∞ (iters)	CPU time, T (s)
10	3.901E-03	1.09E-02	1.530E-02	3.40E-02
20	1.310E-04	1.90E-02	1.512E-05	1.86E-01
50	3.682E-06	1.11E+00	5.318E-07	2.11E+00
100	2.304E-07	5.72E+00	6.921E-08	1.43E+01

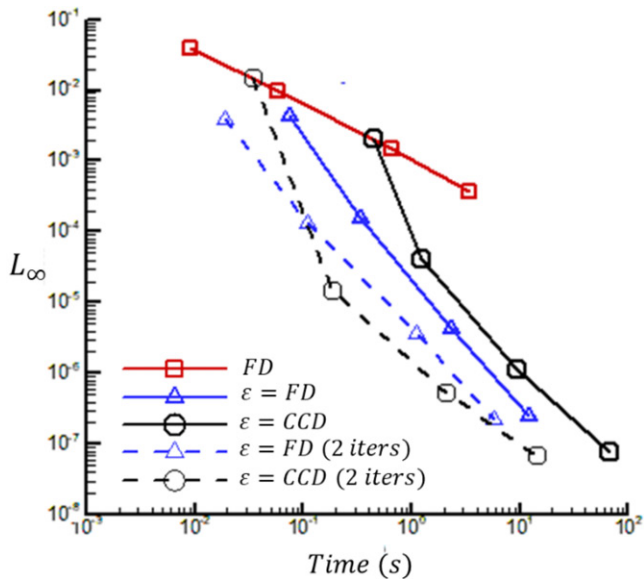


Figure 10. L_∞ error norm versus computational time for different schemes applied to solve Eq. (31). Dashed lines indicate the solution at the second iteration.

present RBF-FD scheme could be further improved when the number of iterations is limited to two.

Table 5 shows the L_∞ error norm and the computational time when the number of iterations is limited to two. The results in Figure 9 are overlaid on the plot in Figure 10. It is found that the cost-effectiveness of the present RBF-FD scheme (ε predicted from CCD) is more cost effective at a certain point. However, as the grid size is further refined, the accuracy per cost of the proposed RBF-FD scheme is independent of the prediction method of ε .

5.1.3. 2D numerical experiment 2

Consider the boundary value problem below:

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -f = -5\pi^2 \sin(\pi x) \cos(2\pi y), & \text{in } \Omega = (0, 1) \times (0, 1), \\ u(x, y) = \sin(\pi x)\cos(2\pi y) & \text{on } \partial\Omega. \end{cases} \tag{32}$$

The problem of Eq. (32) has an exact solution of $u(x, y) = \sin(\pi x)\cos(2\pi y)$. For this problem, one can easily find that the term $-\left(\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4}\right) / \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$ is $3.4\pi^2$. This leads to the constant value of ε_{opt} , that is, 2.7964 for the present RBF-FD scheme. However, the value of the denominator $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$ is zero when $y = 0.25$ and $y = 0.75$. Hence, this numerical experiment is to test the accuracy and the robustness of the present scheme in computing the constant value of ε_{opt} .

The problem in Eq. (32) is solved by using the RBF-FD scheme proposed by Feng and Duan [18]. In their work [18], the expression of the optimal shape parameter for the Poisson equation is modified by substituting the PDE (Poisson equation) into the equation of shape parameter, thus replacing the fourth order derivative terms with the mixed derivative term $\partial^4 u / \partial x^2 \partial y^2$ and the second derivative of source term f . The derived optimal shape parameter can be written as:

$$\varepsilon = \frac{-4}{60f(x, y)} \left[-2 \frac{\partial^2 u}{\partial x^2 \partial y^2} - \frac{\partial^2 f}{\partial x^2} - \frac{\partial^2 f}{\partial y^2} \right] \tag{33}$$

Note that the optimal shape parameter above is expressed as ε instead of ε^2 (as used in [18]) in order to prevent confusion when comparison is performed against the optimal shape parameter derived using the present RBF-FD scheme. The substitution of the exact u and f into Eq. (33) would yield a constant value of $\varepsilon = 17\pi^2/75 = 2.2371$.

For the problem in Eq. (32), we compare the L_1 error of the solution obtained by using the present RBF-FD scheme (with 2 iterations) and the RBF-FD scheme of Feng and Duan [18]. The results are tabulated in Table 6. In general, the accuracy and the rate of convergence of the present RBF-FD scheme is on par with those of Feng and Duan [18]. In fact, the RBF-FD scheme with ε computed using the CCD scheme is slightly more accurate.

The values of ε obtained from both schemes are shown in Figure 11. As observed, the value of ε is almost constant throughout the domain, except along the lines of singularity, that is, $y = 0.25$

Table 6. L_1 error norm of the solution of the problem in Eq. (32) by using the present RBF-FD scheme and that reported in [18] for different grid size h .

h	L_1 error		
	Feng and Duan [18]	RBF – FD, $\varepsilon = \text{FD}$	RBF – FD, $\varepsilon = \text{CCD}$
1/5	4.900E-03	3.080E-02	1.446E-02
1/10	4.100E-04	9.979E-04	4.376E-04
1/20	2.920E-05	2.906E-05	9.551E-06
1/40	1.933E-06	1.760E-06	4.701E-07
1/80	1.242E-07	1.171E-07	3.004E-08

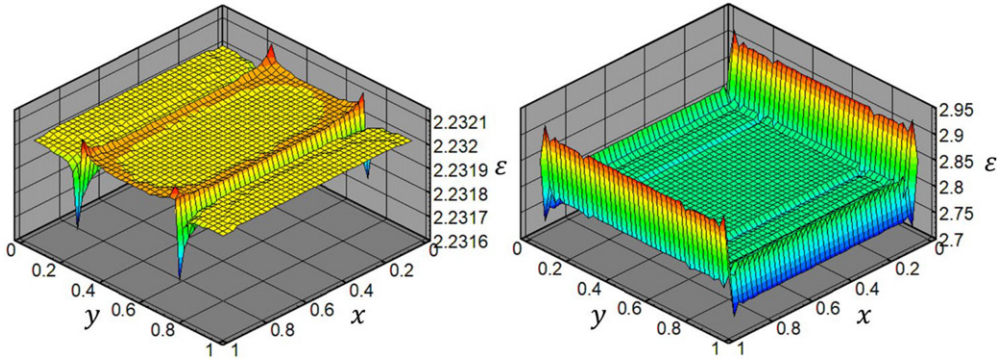


Figure 11. Plot of the distribution of ε value for grid $N = 40$ for the problem in Eq. (32). Left: ε for Feng and Duan [16]. Right: ε for present RBF-FD computed by using CCD scheme.

and $y = 0.75$. The ε value computed using Feng and Duan [18] has minor fluctuation along the singular lines. Meanwhile, the fluctuation of ε predicted using the present RBF-FD scheme is quite apparent near the boundaries. Nevertheless, the accuracy of the present RBF-FD scheme is not significantly affected.

Although Eq. (33) can produce a smoother value of ε near the boundaries, the calculation of Eq. (33) involves the discretization of derivative terms $\partial^2 f / \partial x^2$ and $\partial^2 f / \partial y^2$. In other words, the values of f must be known beforehand at the boundaries. It is also interesting to note that there is a significant difference between the present RBF-FD scheme and the one proposed by Feng and Duan [18]. The latter does not converge to the conventional finite differencing scheme when $\varepsilon \rightarrow 0$; however, the entire discretized term $s''(x)$ reduces to 0.

5.2. Solution of steady convection-diffusion equation

Consider the problem

$$\begin{cases} a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} - k \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = f(x, y), & \text{in } \Omega = (0, 1) \times (0, 1), \\ u = u(x, y), & \text{on } \partial\Omega, \end{cases} \quad (34)$$

Here, the coefficients a and b in Eq. (34) are the transport velocity in x and y directions, respectively. The diffusive coefficient is denoted as k .

The discretization of the diffusion term $\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2$ is already described in Section 5.1, in which the central differencing type of method is applied. However, the discretization of convective terms must take the flow direction of a and b into account. In the present study, the RBF-FD scheme for convective terms is derived by utilizing the upstream nodes. By using $\partial u / \partial x$ and positive value of a as an example, the upwind discretization can be obtained by constructing the RBF-FD equation using the local node (i, j) and the upstream nodes $(i - 1, j)$ and $(i - 2, j)$:

$$\frac{\partial u}{\partial x} \Big|_{i,j} \approx \frac{\partial s}{\partial x} \Big|_{i,j} = \frac{0.5u_{i-2,j} - 2u_{i-1,j} + 1.5u_{i,j}}{\Delta x} + \varepsilon(\Delta x)[0.5u_{i-2,j} - 2u_{i-1,j} + 1.5u_{i,j}], \quad (35)$$

The Taylor series expansion of Eq. (35) is:

$$\frac{\partial s}{\partial x} \Big|_{i,j} = \frac{\partial u}{\partial x} + (\Delta x)^2 \left[\frac{-1}{3} \frac{\partial^3 u}{\partial x^3} - \varepsilon \frac{\partial u}{\partial x} \right] + (\Delta x)^3 \left[\frac{1}{4} \frac{\partial^4 u}{\partial x^4} \right] + (\Delta x)^4 \left[\frac{-7}{60} \frac{\partial^5 u}{\partial x^5} + \varepsilon \frac{1}{3} \frac{\partial^3 u}{\partial x^3} \right] + O(\Delta x^5). \quad (36)$$

From Eq. (36), it is obvious that if $\varepsilon = -\frac{\partial^3 u}{\partial x^3} / \left(3 \frac{\partial u}{\partial x}\right)$, the order of accuracy of Eq. (35) would be $O(\Delta x^3)$. The order of accuracy can be further improved to $O(\Delta x^4)$ by taking into account the term $(\Delta x)^3 \left[\frac{1}{4} \frac{\partial^4 u}{\partial x^4}\right]$ when computing the optimal value of ε :

$$\varepsilon = \left(3(\Delta x) \frac{\partial^4 u}{\partial x^4} - 4 \frac{\partial^3 u}{\partial x^3}\right) / \left(12 \frac{\partial u}{\partial x}\right). \tag{37}$$

The fourth order derivative term $\partial^4 u / \partial x^4$ is also computed and used during the discretization of the second derivative term $\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2$. Therefore, when solving Eq. (34), the additional computational cost incurred while improving the order of accuracy of Eq. (35) from $O(\Delta x^3)$ to $O(\Delta x^4)$ is negligible.

By using the same procedure, the discretization of $\partial u / \partial y$ when b is positive can be obtained from nodes (i, j) , $(i, j - 1)$, and $(i, j - 2)$:

$$\frac{\partial u}{\partial y} \Big|_{i,j} \approx \frac{\partial s}{\partial y} \Big|_{i,j} = \frac{0.5u_{i,j-2} - 2u_{i,j-1} + 1.5u_{i,j}}{\Delta y} + \varepsilon(\Delta y)[0.5u_{i,j-2} - 2u_{i,j-1} + 1.5u_{i,j}], \tag{38}$$

The optimal shape parameter expressed below would provide $O(\Delta y^4)$ accuracy of Eq. (38):

$$\varepsilon = \left(3(\Delta y) \frac{\partial^4 u}{\partial y^4} - 4 \frac{\partial^3 u}{\partial y^3}\right) / \left(12 \frac{\partial u}{\partial y}\right). \tag{39}$$

By using the discretization schemes in Eqs. (28), (35) and (38), the PDE in Eq. (34) is discretized as:

$$\begin{aligned} a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} - k \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) &= a \left[\frac{0.5u_{i-2,j} - 2u_{i-1,j} + 1.5u_{i,j}}{\Delta x} \right] + b \left[\frac{0.5u_{i,j-2} - 2u_{i,j-1} + 1.5u_{i,j}}{\Delta x} \right] \\ &\quad - k \left[\frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{(\Delta x)^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{(\Delta y)^2} \right] \\ &\quad + \varepsilon_{du/dx}(a\Delta x)[-0.5u_{i-2,j} + 2u_{i-1,j} - 1.5u_{i,j}] \\ &\quad + \varepsilon_{du/dy}(b\Delta y)[-0.5u_{i,j-2} + 2u_{i,j-1} - 1.5u_{i,j}] \\ &\quad - \varepsilon_{\Delta u}(k)[u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}]. \end{aligned} \tag{40}$$

In Eq. (40), $\varepsilon_{du/dx}$, $\varepsilon_{du/dy}$, and $\varepsilon_{\Delta u}$ are the shape parameters introduced to optimize the discretization of $\partial u / \partial x$, $\partial u / \partial y$, and $\partial^2 u / \partial x^2 + \partial^2 u / \partial y^2$, respectively. The value of $\varepsilon_{\Delta u}$ is optimized by using Eq. (30) outlined in Section 5.1. The shape parameter of the first derivative terms $\partial u / \partial x$, $\partial u / \partial y$ is optimized using Eqs. (37) and (39). Hence, there are three different values of ε at the same node.

5.2.1. 2D numerical experiment 3

$$\begin{cases} -2 \frac{\partial u}{\partial x} + 2 \frac{\partial u}{\partial y} - \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = \frac{8e^{2(1-x)} + 8e^{2y}}{e-1}, & \text{in } \Omega = (0, 1) \times (0, 1), \\ u = \frac{e^{2(1-x)} + e^{2y} - 2}{e-1}, & \text{on } \partial\Omega. \end{cases} \tag{41}$$

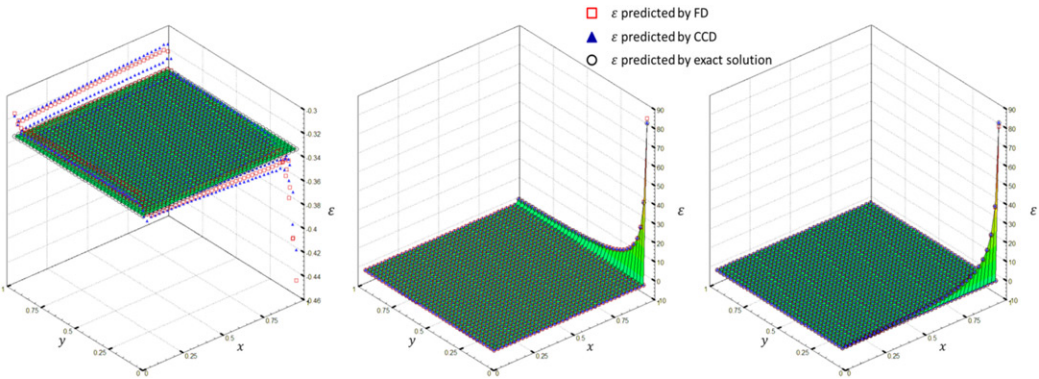
The exact solution $u(x, y)$ of Eq. (41) above is given as $u = (e^{2(1-x)} + e^{2y} - 2) / (e - 1)$. Note that

Table 7. L_∞ error norms of the predicted solution and the number of iterations needed to reach convergent solution for the Eq. (41).

N	$\varepsilon = \text{FD (iters)}$	$\varepsilon = \text{CCD (iters)}$	$\varepsilon = \text{exact solution}$
10	1.522e-3 (9)	1.736e-3 (18)	1.701e-3
20	2.245e-4 (8)	2.732e-4 (15)	2.364e-4
30	6.947e-5 (7)	7.239e-5 (13)	7.232e-5
40	2.991e-5 (6)	3.098e-5 (11)	3.097e-5

Table 8. Minimum and maximum values of the optimal ε predicted by using different approaches in solving Eq. (41). The discretized equations for $\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$, $\partial u/\partial x$ and $\partial u/\partial y$ have their own ε_{\min} and ε_{\max} values.

	N	$\varepsilon_{\min}(\varepsilon_{\max})$		
		$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$	$\frac{\partial u}{\partial x}$	$\frac{\partial u}{\partial y}$
$\varepsilon = \text{FD}$	10	-0.51 (-0.26)	-1.42 (27.09)	-1.27 (20.88)
	20	-0.46 (-0.23)	-1.38 (45.60)	-1.30 (40.74)
	30	-0.45 (-0.31)	-1.36 (62.25)	-1.31 (60.71)
	40	-0.44 (-0.31)	-1.36 (85.09)	-1.32 (80.70)
$\varepsilon = \text{CCD}$	10	-0.46 (-0.26)	-1.55 (23.34)	-1.16 (23.31)
	20	-0.43 (-0.29)	-1.45 (42.98)	-1.24 (42.88)
	30	-0.42 (-0.30)	-1.41 (62.87)	-1.27 (62.79)
	40	-0.42 (-0.31)	-1.39 (82.82)	-1.29 (82.75)
$\varepsilon = \text{exact solution}$	10	-0.33 (-0.33)	-1.53 (23.04)	-1.13 (23.04)
	20	-0.33 (-0.33)	-1.43 (42.82)	-1.23 (42.82)
	30	-0.33 (-0.33)	-1.40 (62.76)	-1.27 (62.76)
	40	-0.33 (-0.33)	-1.38 (82.74)	-1.28 (82.74)


Figure 12. Distribution of the value of ε throughout the domain for Eq. (41). Left plot: ε for $\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$. Center: ε for $\partial u/\partial x$. Right: ε for $\partial u/\partial y$.

the transport velocity in x -direction is negative for the problem in Eq. (41), hence, the term $\partial u/\partial x$ is discretized by using the nodes (i, j) , $(i + 1, j)$, and $(i + 2, j)$ (see Figure 4).

The L_∞ error norms of the solution for Eq. (41) obtained by using the present RBF-FD scheme are tabulated in Table 7. The accuracy of the solution can be obtained through the use of ε computed by using either approach, as the solution accuracies are on par (at the same order of magnitude) with that using the value of ε computed from the exact solution.

The minimum and maximum values of ε applied to $\partial u/\partial x$, $\partial u/\partial y$, $\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$ for Eq. (41) are tabulated in Table 8. Meanwhile, the distributions of the values of ε across the domain at $N = 40$ are plotted in Figures 12 and 13. Seemingly, the values predicted by using FD (red square in Figure 12) or CCD (blue delta in Figure 12) scheme agree well with that computed from the exact solution, except for the ε applied to the discretization of $\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$, where the value of ε predicted by either numerical approach exhibits a larger discrepancy near

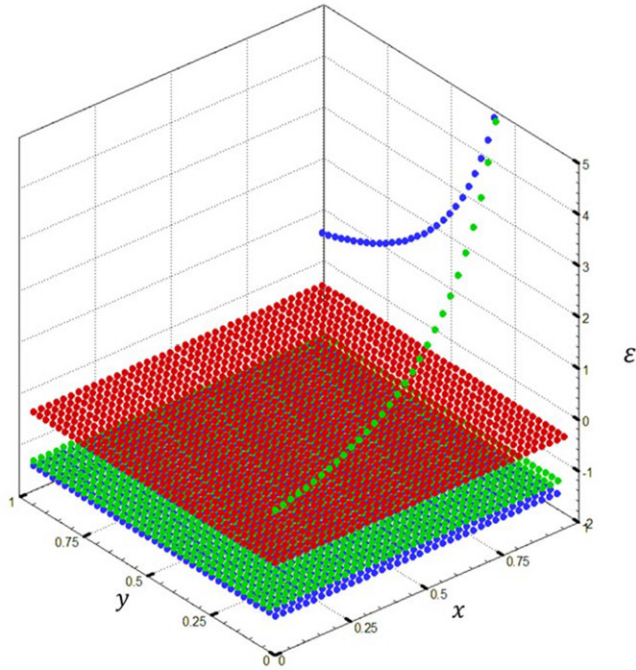


Figure 13. Comparing the values of ε applied to the discretized terms $\partial^2u/\partial x^2 + \partial^2u/\partial y^2$ (red circle), $\partial u/\partial x$ (blue circle), and $\partial u/\partial y$ (green circle) in Eq. (41). The points with the value of $\varepsilon > 5$ are not shown in the figure.

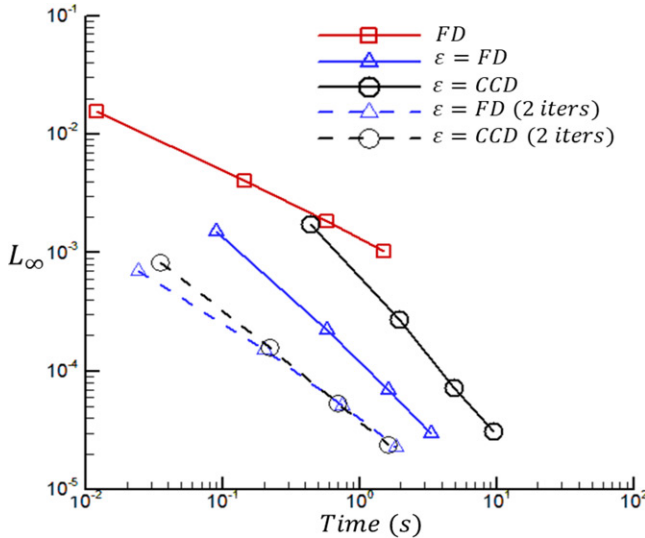


Figure 14. L_∞ error norm versus computational time for different schemes applied to solve Eq. (41). Dashed lines indicate the solutions at the second iteration.

the domain boundaries. From Figure 12, the optimal values of ε for discretization of the terms $\partial u/\partial x$ and $\partial u/\partial y$ show a sudden increase at boundaries $x = 1$ and $y = 0$, respectively. We have to stress that this occurs due to the nature of the optimal ε for Eq. (41) instead of the computational error.

The cost-effectiveness of the present scheme is further improved when the number of iterations is limited to two. From Figure 14, it is clear that the solution of the present RBF-FD

Table 9. L_∞ error norm and the CPU time in solving Eq. (41) by using the present RBF-FD scheme with only two iterations under different values of N .

N	RBF – FD, $\varepsilon = \text{FD}$		RBF – FD, $\varepsilon = \text{CCD}$	
	L_∞	CPU time, T (s)	L_∞ (iters)	CPU time, T (s)
10	6.979e-4	2.40e-2	8.244e-4	3.50e-2
20	1.511e-4	2.02e-1	1.584e-4	2.20e-1
30	5.147e-5	7.41e-1	5.303e-5	6.89e-1
40	2.310e-5	1.83e+0	2.367e-5	1.62e+0

Table 10. L_∞ error norms of the solution obtained for Eq. (41) using the optimal ε in Eq. (42) at different grid resolutions N .

N	RBF – FD, $\varepsilon = \text{FD}$	RBF – FD, $\varepsilon = \text{CCD}$
10	1.072e-3	1.076e-3
20	1.776e-4	1.405e-4
30	5.705e-5	4.821e-5
40	2.497e-5	2.174e-5

scheme for $N = 10$ is as accurate as the solution obtained by using the conventional FD scheme for $N = 40$. However, in order to achieve the same level of solution accuracy, the computation time of the conventional FD scheme is approximately 30 times higher than that of the present RBF-FD scheme with two iterations. Again, the cost-effectiveness of the RBF-FD scheme is independent of the prediction method for calculating ε as the grid resolution increases. The error norm and the computation time required for solving Eq. (41) is reported in Table 9.

5.3. Optimizing ε of the whole PDE

In Section 5.2, the value of ε is optimized for the derivative terms in Eq. (41), and the results are obtained using three optimal ε values at the same node. Alternatively, the value of ε could also be optimized to eliminate the leading error term of the modified partial differential equation. This could result in a more generic way of deriving the optimal ε , instead of using three different values of ε as outlined in Section 5.2.

By applying the Taylor series expansion, the optimal shape parameter for the discretized convection-diffusion Eq. (34) can be found as

$$\varepsilon = \frac{1}{12 \left(a(\Delta x)^2 \frac{\partial u}{\partial x} + b(\Delta y)^2 \frac{\partial u}{\partial y} + k \left((\Delta x)^2 \frac{\partial^2 u}{\partial x^2} + (\Delta y)^2 \frac{\partial^2 u}{\partial y^2} \right) \right)} \left[a \left(-4(\Delta x)^2 \frac{\partial^3 u}{\partial x^3} + 3(\Delta x)^3 \frac{\partial^4 u}{\partial x^4} \right) + b \left(-4(\Delta y)^2 \frac{\partial^3 u}{\partial y^3} + 3(\Delta y)^3 \frac{\partial^4 u}{\partial y^4} \right) - k \left((\Delta x)^2 \frac{\partial^4 u}{\partial x^4} + (\Delta y)^2 \frac{\partial^4 u}{\partial y^4} \right) \right]. \quad (42)$$

Note that the optimal ε in Eq. (42) above is also a function of transport velocities a, b and diffusive coefficient k . When $a = b = 0$, Eq. (34) is reduced to the Poisson equation in Eq. (31). The optimal shape parameter in Eq. (42) is also simplified to Eq. (30).

By employing the optimal shape parameter in Eq. (42), Eq. (41) is solved again by using the present RBF-FD scheme. The number of iterations is limited to two. The L_∞ error norms of the numerical solutions are tabulated in Table 10. As compare with the result obtained from segregated ε (reported in Table 9), there is no significant improvement in terms of the solution accuracy. Hence, regardless of whether the shape parameter is optimized for the PDE under

investigation, or is optimized for each individual term appeared in the PDE, the solution accuracy is not severely affected.

6. Conclusion

In this article, a new approach to solve the PDE problem with MQ-based RBF-FD scheme using the optimal value of the shape parameter has been proposed. In the present RBF-FD scheme, the optimal shape parameter that eliminates the leading error term depends on the local solution derivatives. In other words, the optimal value of the shape parameter is indeed varying throughout the domain. The RBF-FD scheme using the constant value of the shape parameter is found to be more accurate than the conventional FD scheme within a limited range of shape parameter. However, the proposed RBF-FD scheme employing the optimal shape parameter field has been found to be more accurate and effective. In general, the present RBF-FD scheme that employs the shape parameter computed by using the CCD scheme is more accurate than that predicted using the FD scheme. Even though the computational cost of CCD scheme is higher than that of the explicit FD scheme, the cost-effectiveness of the RBF-FD scheme is independent of the prediction methods employed in determining the shape parameter when the number of iterations for calculating ε is limited to 2. Meanwhile, by considering the same grid resolution, the usage of a more accurate scheme in predicting ε would yield a more accurate solution. As a result, it is essential to predict the derivative terms in the optimal shape parameter by using a more accurate scheme.

Funding

This work has been supported by the Ministry of Science and Technology (MOST) of the Republic of China (R.O.C) under the grant MOST105-2221-E-002-066.

ORCID

Y. L. Ng  <http://orcid.org/0000-0002-6836-4076>

References

- [1] R. L. Hardy, "Multiquadric equations of topography and other irregular surfaces," *J. Geophys. Res.*, vol. 76, no. 8, pp. 1905–1915, 1971. DOI: [10.1029/JB076i008p01905](https://doi.org/10.1029/JB076i008p01905).
- [2] B. Fornberg, and G. Wright, "Stable computation of multiquadric interpolants for all values of the shape parameter," *Comput. Math. Appl.*, vol. 48, no. 5–6, pp. 853–867, 2004. DOI: [10.1016/j.camwa.2003.08.010](https://doi.org/10.1016/j.camwa.2003.08.010).
- [3] P. González-Rodríguez, M. Moscoso, and M. Kindelan, "Laurent Expansion of the Inverse of Perturbed, Singular Matrices," *J. Comput. Phys.*, vol. 299, pp. 307–319, 2015. DOI: [10.1016/j.jcp.2015.07.006](https://doi.org/10.1016/j.jcp.2015.07.006).
- [4] M. Kindelan, M. Moscoso, and P. González-Rodríguez, "Radial Basis Function Interpolation in the Limit of Increasingly Flat Basis Functions," *J. Comput. Phys.*, vol. 307, pp. 225–242, 2016. DOI: [10.1016/j.jcp.2015.12.015](https://doi.org/10.1016/j.jcp.2015.12.015).
- [5] G. Chandhini and Y. Sanyasiraju, "Local RBF-FD solutions for steady convection-diffusion problems," *Int. J. Numer. Methods Eng.*, vol. 72, no. 3, pp. 352–378, 2007. DOI: [10.1002/nme.2024](https://doi.org/10.1002/nme.2024).
- [6] S. A. Sarra, "Adaptive radial basis function methods for time dependent partial differential equations," *Appl. Numer. Math.*, vol. 54, no. 1, pp. 79–94, 2005. DOI: [10.1016/j.apnum.2004.07.004](https://doi.org/10.1016/j.apnum.2004.07.004).
- [7] Z. H. Wang, Z. Huang, W. Zhang, and G. Xi, "A meshless local radial basis function method for two-dimensional incompressible Navier-Stokes equations," *Numer. Heat Transf. Part B Fundam.*, vol. 67, no. 4, pp. 320–337, 2015. DOI: [10.1080/10407790.2014.955779](https://doi.org/10.1080/10407790.2014.955779).
- [8] N. Li, Z. Tan, and X. Feng, "Novel two-level discretization method for high dimensional semilinear elliptic problems base on RBF-FD scheme," *Numer. Heat Transf. Part B Fundam.*, vol. 72, no. 5, pp. 349–360, 2017. DOI: [10.1080/10407790.2017.1409511](https://doi.org/10.1080/10407790.2017.1409511).

- [9] C. Huang, C. Lee, and A. H. Cheng, “Error estimate, optimal shape factor, and high precision computation of multiquadric collocation method,” *Eng. Anal. Bound Elem.*, vol. 31, no. 7, pp. 614–623, 2007. DOI: [10.1016/j.enganabound.2006.11.011](https://doi.org/10.1016/j.enganabound.2006.11.011).
- [10] V. Bayona, M. Moscoso, M. Carretero, and M. Kindelan, “RBF-FD formulas and convergence properties,” *J. Comput. Phys.*, vol. 229, no. 22, pp. 8281–8295, 2010. DOI: [10.1016/j.jcp.2010.07.008](https://doi.org/10.1016/j.jcp.2010.07.008).
- [11] V. Bayona, M. Moscoso, and M. Kindelan, “Optimal variable shape parameter for multiquadric based RBF-FD method,” *J. Comput. Phys.*, vol. 231, no. 6, pp. 2466–2481, 2012. DOI: [10.1016/j.jcp.2011.11.036](https://doi.org/10.1016/j.jcp.2011.11.036).
- [12] J. Guo and J.-H. Jung, “A RBF-WENO Finite volume method for hyperbolic conservation laws with the monotone polynomial interpolation method,” *Appl. Numer. Math.*, vol. 112, pp. 27–50, 2017. DOI: [10.1016/j.apnum.2016.10.003](https://doi.org/10.1016/j.apnum.2016.10.003).
- [13] J. Guo, and J.-H. Jung, “Radial basis function ENO and WENO finite difference methods based on the optimization of shape parameters,” *J. Sci. Comput.*, vol. 70, no. 2, pp. 551–575, 2017. DOI: [10.1007/s10915-016-0257-y](https://doi.org/10.1007/s10915-016-0257-y).
- [14] M. Buhmann and N. Dyn, “Spectral convergence of multiquadric interpolation,” *Proc. Edinburg Math. Soc.*, vol. 36, no. 2, pp. 319–333, 1993. DOI: [10.1017/S0013091500018411](https://doi.org/10.1017/S0013091500018411).
- [15] R. B. Platte, “How fast do radial basis function interpolants of analytic functions converge?,” *IMA J. Numer. Anal.*, vol. 31, no. 4, pp. 1578–1597, 2011. DOI: [10.1093/imanum/drq020](https://doi.org/10.1093/imanum/drq020).
- [16] E. Larsson and B. Fornberg, “Theoretical and computational aspects of multivariate interpolation with increasingly flat radial basis functions,” *Comput. Math. Appl.*, vol. 49, no. 1, pp. 103–130, 2005. DOI: [10.1016/j.camwa.2005.01.010](https://doi.org/10.1016/j.camwa.2005.01.010).
- [17] Frake, R. A Critical Comparison of Some Methods for Interpolation of Scattered Data. Nav Postgrad Sch Tech Rep 53-79-003 1979.
- [18] R. Feng and J. Duan, “High accurate finite differences based on RBF equations,” *J. Sci. Comput.*, vol. 76, no. 3, pp. 1785–1812, 2018. DOI: [10.1007/s10915-018-0684-z](https://doi.org/10.1007/s10915-018-0684-z).
- [19] P. C. Chu and C. W. Fan, “A three-point combined compact difference scheme,” *J. Comput. Phys.*, vol. 140, no. 2, pp. 370–399, 1998. DOI: [10.1006/jcph.1998.5899](https://doi.org/10.1006/jcph.1998.5899).
- [20] T. K. Sengupta, V. Lakshmanan, and V. Vijay, “A new combined stable and dispersion relation preserving compact scheme for non-periodic problems,” *J. Comput. Phys.*, vol. 228, no. 8, pp. 3048–3071, 2009. DOI: [10.1016/j.jcp.2009.01.003](https://doi.org/10.1016/j.jcp.2009.01.003).
- [21] C. H. Yu, Y. G. Bhumkar, and T. Sheu, “Dispersion relation preserving combined compact difference schemes for flow problems,” *J. Sci. Comput.*, vol. 62, no. 2, pp. 482–516, 2015. DOI: [10.1007/s10915-014-9864-7](https://doi.org/10.1007/s10915-014-9864-7).
- [22] C. Chang, C. Yu, and T. Sheu, “Long-time asymptotic solution structure of Camassa-Holm equation subject to an initial condition with non-zero reflection coefficient of the scattering data,” *J. Math. Phys.*, vol. 57, no. 10, pp. 103508, 2016. DOI: [10.1063/1.4966112](https://doi.org/10.1063/1.4966112).
- [23] S. K. Lele, “Compact finite difference schemes with spectral-like resolution,” *J. Comput. Phys.*, vol. 103, no. 1, pp. 16–42, 1992. DOI: [10.1016/0021-9991\(92\)90324-R](https://doi.org/10.1016/0021-9991(92)90324-R).
- [24] K. Zhang, L. Wang, and Y. Zhang, “Improved finite difference method with a compact correction term for solving Poisson’s equations,” *Numer. Heat Transf. Part B Fundam.*, vol. 70, no. 5, pp. 393–405, 2016. DOI: [10.1080/10407790.2016.1215715](https://doi.org/10.1080/10407790.2016.1215715).
- [25] G. B. Wright and B. Fornberg, “Scattered node compact finite difference-type formulas generated from radial basis functions,” *J. Comput. Phys.*, vol. 212, no. 1, pp. 99–123, 2006. DOI: [10.1016/j.jcp.2005.05.030](https://doi.org/10.1016/j.jcp.2005.05.030).
- [26] V. Bayona, M. Moscoso, and M. Kindelan, “Optimal constant shape parameter for multiquadric based RBF-FD method,” *J. Comput. Phys.*, vol. 230, no. 19, pp. 7384–7399, 2011. DOI: [10.1016/j.jcp.2011.06.005](https://doi.org/10.1016/j.jcp.2011.06.005).