A new adaptive-node refinement algorithm based on multiquadric method for the nearly singular problems

masoumeh koushki[1], ehsan jabbari[2], and mahdi ahmadinia[2]

[1]Lorestan University Faculty of Engineering
[2]Qom University of Medical Sciences and Health Services

July 7, 2020

Abstract

In this paper, a new adaptive-node refinement algorithm for multiquadric (MQ) method applied to nearly singular PDEs, has been presented. Besides, for the first time, the concept of the gradient has been employed in the refinement index. Regions with high gradients are identified based on the average value of the proposed function solution. The solution is approximated using a first-order derivative of interpolation with MQ. In the framework of the adaptive algorithm, the average of the proposed function was used as an indicator that determines where the point distribution can be refined and nodes can be added or removed based on this indicator. Different applications of the proposed adaptive algorithm are investigated through numerical examples. It has been revealed that the proposed algorithm is able to identify the singularities both in the domain and near boundaries and the numerical results of the MQ method confirm the accuracy and efficiency of the algorithm. The main advantage of this algorithm is that in the first steps, the regions with high gradients are identified correctly and the convergence speed of the algorithm increases continuously.

Introduction

Radial basis functions (RBF) are used as interpolation functions [1]. RBF has been used in various fields such as neural network, image processing and, partial differential equation approximation (PDEs) [2- 6]. RBFs merely depend on the distance of the entry points from the center x_j and the form$\varphi\left(|x-x_j|\right)$. The effective parameter in the accuracy of the RBF method is the shape parameter. If this parameter is used in RBF, $\varphi\left(r\right)$ is replaced by $\varphi(r,c)$. Some of the most commonly used RBFs are listed in Table 1 [6].

The main feature of the RBFs is that it does not need any mesh. The Euclidian distance between every two points is the only geometric feature used in the approximation of the RBF. The distances are easily calculated in any number of spatial dimensions so that complexity does not occur in the higher dimensions of geometry. The points used for definition of the RBFs are called centers that can be distributed arbitrary based on the given geometry. The smooth RBF functions have high convergence and the non-smooth functions converge slowly [7,8]. Therefore, it is necessary to use adaptive methods in these functions. In these methods, the position of the high gradients in the function is identified [9].

In an adaptive method, the points are distributed based on the gradient. In locations with high gradients, more nodes are considered and other locations require fewer nodes. The location of the nodes plays a key role in the stability of the interpolation matrix, which shows its effect on the Gibbs and Runge phenomenon [10-12].

Considerable research has been done on combining adaptive methods with RBF. Greedy algorithm in adaptive method was applied by Schaback and Wendland [13] and Hon et al. [14]. The results showed that

this method results leads to linear convergence of the interpolation matrix. Han [15] provided an indicator in the adaptive MQ scheme based on the weak formulation of the governing equation. Based on this indicator, locations with high gradients were identified then nodes were added. Ling and Trummer [16] Investigated the Hon's indicator and modified it to solve transformed boundary value problems. Sarra [17] proposed an adaptation method. In this method, the nodes are modified based on the point redistribution algorithm located along an arc. The method yields good results for the one-dimensional Burger's and the nearly singular time-dependent advection equation. Wu [18,19] has offered an adaptation scheme to solve the time-dependent problems of the PDE. One-dimensional problems have discussed in this research but an upgrade to higher dimensions presented. Bozzini et al. [20] presented an adaptive scheme based on the iterative algorithm. They used the combination of B-spline and scaled MQ. Behrens and Iske [21] Presented an adaptive algorithm based on RBF. The error indicator was used to identify locations that require higher accuracy and Behrens et al. [22] applied this algorithm to linear evolutionary PDEs and obtained accurate results. Gomez et al. [23] presented a RBF dynamic domain decomposition algorithm. This algorithm was applied in a large PDEs with high gradient. Driscoll and Herudono [9] developed an adaptive method based on the residual subsampling technique. Based on this technique, domain nodes are modified in locations with high indicator. This algorithm was applied to boundary value problems and initial boundary value problems with localized features. Libre et al. [25, 26] presented an adaptive scheme based on wavelet for nearly singular problems. The proposed method applied in one and two problems with internal and boundary near singularities and obtained accurate results. Shanazari and Hosami Presented adaptive mesh technique for irregular domains. Nodes are generated by the co-distribution technique in the rectangular domain, then mapped to the physical domain. The method was used to solve the PDE problems with the RBF and provided accurate results [27]. A dynamic algorithm based on this plate spline radial basis function presented by Esmaeilbeigi and Hosseini. The algorithm identifies locations with high gradient and refinement nodes perform in these locations. This method was applied for nearly singular problems in large domains and obtained appropriate results [28]. A direct multi-objective algorithm for optimizing node distribution was proposed by Roque et al. This algorithm was used to solve composite plate problems by collocation method [29]. Kamranian et al. developed a new adaptive moving least-squares method. An error indicator specifies the location of the refinement nodes. This method was used to solve boundary value problems using the Local Petrov-Galerkin method [30]. Cavoretto and Rossi Presented an adaptation-refinement node algorithm based on the RB-PUM method. The location of the node refinement is determined by the error indicator and the algorithm is repeated until the error is lower than the tolerance [31].

In nearly singular problems, the node spacing must be very fine to detect near-singular region. Increasing the points in the total domain occurs instability and ill-condition of the interpolation matrix and running the method will require a lot of memory to store the data. Therefore, regional refinement of points and the presentation of adaptive-node refinement algorithms in these problems are essential.

In many adaptive methods, RBFs with shape parameters have been used. Determining an optimal shape parameter is still an important challenge. In this study, the genetic optimization algorithm has been used to find the optimal value of the shape parameter.

In this research, nearly singular PDE problems will be investigated whose exact solution will have regions with high gradients. Specially, we introduce a proposed function that has been constructed based on the MQ solution. The proposed function identifies the regions of the domain to which the node must be added.

The layout of the paper is as follows: In section 2, a review of the collocation method for the boundary value problems is performed. In section 3 the main steps of the proposed algorithm are presented. Thereafter, the genetic optimization algorithm will be introduced to find the optimal value of the shape parameter in section 4. The results of numerical experiments are discussed in section 5. Finally, conclusions are given in section 6.

RBF approximation for boundary value problems

In this section, the asymmetric RBF collocation method for the following boundary value problem is presented.

$Hu = f \ in \ \Omega \subset R^d, \quad Bu = s \quad on \quad \partial\Omega.(1)$

where H is an arbitrary linear differential operator,$\partial\Omega$ denotes the boundary of the domain Ω, and B is a boundary operator, which can be Dirichlet, Neumann or a mixed type. The functions $f, s : R^d \longrightarrow R$ are provided.

In Kansa approach [5], the domain is discretized by N collocation nodes $\{x_k\}_{k=1}^{N} \subset \Omega$, which can be divided into interior $\{x_k\}_{k=1}^{N_I}$nodes and$\{x_k\}_{k=N_{I+1}}^{N} \in \partial\Omega$boundary nodes. In order to obtain the approximate solution$\tilde{u}\ (x)$ to the exact solution u(x) of the boundary value problem defined by (1), the radial approximation $\tilde{u}\ (x)$ is given by

$\tilde{u}\ (x) = \sum_{j=1}^{N} \alpha_j \phi\left(\|x - x_j\|\right) + p\left(x\right), \quad x \in R^d (2)$

where α_j are the unknown coefficients to be determined. Here$\phi\left(\|.\|\right)$, in which$\|.\|$ is the Euclidean norm, is an arbitrary RBF. The degree of polynomial p(x) is determined by the selected RBF. The polynomial p(x) is not necessary for positive definite RBF but for guaranteeing singularity in semi-positive RBF is necessary [35].

If *p(x)=0* , one obtains:

$\tilde{u}\ (x) = \sum_{j=1}^{N} \alpha_j \phi\left(\|x - x_j\|\right), \quad x \in R^d\ (3)$

Substituting (3) in (1):

$\sum_{j=1}^{N} \alpha_j \mathrm{H}\ \left(\|x_k - x_j\|\right) = f\left(x_k\right), \quad k = 1, 2, \ldots., N_i (4)$

$\sum_{j=1}^{N} \alpha_j \mathrm{B}\ \left(\|x_k - x_j\|\right) = s\left(x_k\right), \quad k = N_i\ + 1, \ldots.., N (5)$

which must be solved for the unknowns$\left\{\alpha_{j=1}^{N}\right\}$. The prevalent RBFs are shown in Table 1, with $r = \|x_k - x_j\|$. In this paper, MQ will be studied.

The derivative of Eq. (3) at each center $x = x_k$ is given by:

$\tilde{u}'_{f,x}\left(x_k\right) = \sum_{j=1}^{N} \alpha_j \frac{(x_k - x_j)c^2}{\sqrt{1+c^2(x_k - x_j)^2}} (6)$

Adaptive-node refinement algorithm

At first, the proposed function for detecting the regions to be refined, will be introduced. then in marked locations, the refinement scheme will be described.

3.1. The proposed function

The proposed function that is constructed based on the first-order derivative of interpolation using MQ is defined by:

$f\left(x, y\right) = abs\left(\tilde{u}'_x\left(x, y\right)\right) + abs\left(\tilde{u}'_y(x, y)\right) (7)$

or

$f\left(x, y\right) = \sqrt{\tilde{u}'^{\,2}_x\left(x, y\right) + \tilde{u}'^{\,2}_y(x, y)}\ (8)$

An interpolation $\tilde{u}(x,y)$ based on(x_k, y_k) , $k = 1, 2, \ldots N$, will be constructed. Then the derivative of $\tilde{u}(x,y)$ is computed in each direction x and y and substituted in equation (8). Based on the value of $f(x,y)$, the average indicator η is defined as:

$\eta = \frac{\sum_{k=1}^{N} f(x(k),y(k))}{N}$ (9)

If the value of $f(x,y)$ is smaller than η then$\tilde{u}(x,y)$ can be represented by a linear function. On the other hand, a high η commonly indicates that the numerical approximation around $\tilde{u}(x,y)$ has a sharp variation, which generally corresponds to regions with a high gradient. Based on the average indicator η , the regions that should be refined will be determined and in each algorithm loop, the nodes that are less of the indicator will be removed. The advantage of the proposed function is that it simply and quickly identifies regions with high gradients and makes the adaptive-refinement algorithm more efficient.

3.2. The structure of adding node

Based on the average indicator η , the regions to be refined will be marked. Nodes are added to the marked regions. This procedure increases the density of points in regions with high gradients. For nodes located in high gradient regions, three nodes are added at each step to their neighborhood: (up, right and up-right) or (down, left and down-left). It should be noted that the location of neighboring points with other patterns such as: adding points to the top and bottom of the marked point, adding the point to the left and right of the marked point, adding six points in each step or random selection of points around the marked point Tests have been performed, but these patterns have in inaccurate results or ill-conditioned interpolation matrix.

The distance between the points and their neighbor points is calculated based on the Median of Euclidean distance between them. In addition to the median distance, the mean distance also was examined. In the algorithm, the median one behaves better than the average one.

The following relationships are used to build neighbor nodes. The coordinates of the marked points is supposed to be(x_h, y_h). The Median of Euclidean distance of these points is determined by (MD):

$D = \sqrt{\left(x_h\left(k_h+1\right) - x_h\left(k_h\right)\right)^2 + \left(y_h\left(k_h+1\right) - y_h\left(k_h\right)\right)^2}, \qquad MD = median\left(D\right),\ k_h = 1, 2, \ldots, N_h$ (10)

The neighbor points are defined as:

$x_h^{\text{top}} = x_h \quad , \quad y_h^{\text{top}} = y_h + \frac{1}{2}\text{MD}$(11)

$x_h^{\text{right}} = x_h + \frac{1}{2}MD \quad , \quad y_h^{\text{right}} = y_h$(12)

$x_h^{top-right} = x_h + \frac{1}{2}MD \quad , \quad y_h^{top-right} = y_h + \frac{1}{2}\text{MD}$(13)

$x_h^{\text{down}} = x_h \quad , \quad y_h^{\text{down}} = y_h - \frac{1}{2}\text{MD}$(14)

$x_h^{\text{left}} = x_h - \frac{1}{2}MD \quad , \quad y_h^{\text{left}} = y_h$(15)

$x_h^{down-left} = x_h - \frac{1}{2}MD \quad , \quad y_h^{down-left} = y_h - \frac{1}{2}\text{MD}$(16)

3.3. General algorithm

In this section, the general algorithm is constructed from the combination of the proposed function and the refinement scheme then the initial boundary values problems will be solved using the MQ solution.

A short description of the algorithm is given by:

- Identifying the high gradient regions
- Refining nodes

- Creating the approximation function using MQ
- Calculating error in test points

Iterating the above steps until convergence

In this algorithm, convergence occurs when the root mean square error (E) is less than a given value δ.

$$E=\sqrt{\frac{1}{N_t}\sum_{j=1}^{N_t}\left(\tilde{u}_j^{S+1}-\tilde{u}_j^{S}\right)^2},\quad S=1,2,3,\ldots(17)$$

Where S and N_t represent step number and number of evaluation points, respectively. $\tilde{u}_j^{S+1}$ and$\tilde{u}_j^{S}$ represent the numerical solution in step*S+1* and *S,* respectively.

Algorithm 1. General Algorithm Steps

Definition of variables: the computational domain (Ω), the number of points in each direction (x, y) of the computational domain (d), coordinates of points with high gradient ($x_h,\ y_h$), the distance between points with high gradient (MD) and δ to be a given real number. Then the steps of the algorithm are as follows:

1. N: Selecting $d\times d$ grid of equally spaced points in the computational domain Ω;
2. c : Determining the optimal shape parameter;
3. u_j^s: Creating MQ solution with N center points and$c$;
4. Solving PDE using u_j^s in evaluation points (N_t) ;
5. *d=d+1;*
6. $N=d\times d$;
7. u_j^{s+1}*:* Creating MQ solution for N center points and the optimal shape parameter;
8. $f\left(x(k),y(k)\right)=\sqrt{\left(u'^{s+1}_{\mathrm{xj}}\left(x(k),y(k)\right)\right)^2+\left(u'^{s+1}_{\mathrm{yj}}\left(x(k),y(k)\right)\right)^2},\quad k=1,2,\ldots,N$
9. $\eta=\frac{\sum_{k=1}^{N}f(x(k),y(k))}{N}$;
10. If $f\left(x(k),y(k)\right)>\eta$ then (x,y) to be substituted in (x_h,y_h);
11. Solving PDE using FMQ2 in evaluation points(N_t);
12. Calculating$E=\sqrt{\frac{1}{N_t}\sum_{j=1}^{N_t}\left(u_j^{s+1}\left(x(j),y(j)\right)-u_j^{s}\left(x(j),y(j)\right)\right)^2}$;**While** $E>\delta$ **do**

- *d=d+1* ;
- $N=d\times d$;
- $D_k=\sqrt{\left(x_h\left(k_h+1\right)-x_h\left(k_h\right)\right)^2+\left(y_h\left(k_h+1\right)-y_h\left(k_h\right)\right)^2},\quad MD=median\left(D_k\right),\ \backslash nk_h=1,2,\ldots,N_h$;
- $x_h^{\mathrm{right}}=x_h+\frac{1}{2}MD\quad,\ y_h^{\mathrm{right}}=y_h$;
- $x_h^{\mathrm{top}}=x_h\quad,\ y_h^{\mathrm{top}}=y_h+\frac{1}{2}MD$;
- $x_h^{up-right}=x_h+\frac{1}{2}MD\quad,\ y_h^{up-right}=y_h+\frac{1}{2}\mathrm{MD}$
- $N=N\cup(x_h,y_h)\cup\left(x_h^{\mathrm{right}},y_h^{\mathrm{right}}\right)\cup\left(x_h^{\mathrm{top}},\ y_h^{\mathrm{top}}\right)\cup\left(x_h^{up-right},y_h^{up-right}\right)$;
- $\tilde{u}_j^s\left(x,y\right)$: Creating MQ solution using$N$ center points and the optimal shape parameter;
- $f\left(x(k),y(k)\right)=\sqrt{\left(\tilde{u}'^{s}_{\mathrm{jx}}\left(x(k),y(k)\right)\right)^2+\left(\tilde{u}'^{s}_{\mathrm{jy}}\left(x(k),y(k)\right)\right)^2},\quad k=1,\ 2,\ \ldots,N$;
- If $f\left(x(k),y(k)\right)>\eta$ then (x,y) to be substituted in (x_h,y_h);
- $\tilde{u}_j^S$: Solution of PDE using $\tilde{u}(x,y)$ in evaluation points (N_t);
- *d=d+1;*
- $N=d\times d$;
- $D_k=\sqrt{\left(x_h\left(N_h+1\right)-x_h\left(N_h\right)\right)^2+\left(y_h\left(N_h+1\right)-y_h\left(N_h\right)\right)^2},\quad MD=median\left(D_k\right),\ \backslash nk_h=1,2,\ldots,N_h$;
- $x_h^{\mathrm{left}}=x_h-\frac{1}{2}MD\quad,\ y_h^{\mathrm{left}}=y_h$;
- $x_h^{\mathrm{down}}=x_h\quad,\ y_h^{\mathrm{down}}=y_h-\frac{1}{2}MD$;
- $x_h^{down-left}=x_h-\frac{1}{2}MD\quad,\ y_h^{down-left}=y_h-\frac{1}{2}MD$;

- $N = N \cup (x_h, y_h) \cup \left(x_h^{\text{left}}, y_h^{\text{left}}\right) \cup \left(x_h^{\text{down}},\ y_h^{\text{down}}\right) \cup \left(x_h^{down-left}, y_h^{down-left}\right)$;
- $\tilde{u}_j^{s+1}(x, y)$: Creating MQ solution with N center points and the optimal shape parameter;
- $f(x(k), y(k)) = \sqrt{\left(\tilde{u}'^{s+1}_{\text{xj}}(x(k), y(k))\right)^2 + \left(\tilde{u}'^{s+1}_{\text{yj}}(x(k), y(k))\right)^2}, \quad k = 1, 2, \ldots, N$;
- If $f(x(k), y(k)) > \eta$ then (x, y) to be substituted in (x_h, y_h);
- $\tilde{u}_j^{S+1}$ = Solution of PDE with $\tilde{u}(x, y)$in evaluation points (N_t);
- $E = \sqrt{\frac{1}{N_t}\sum_{j=1}^{N_t}\left(\tilde{u}_j^{S+1} - \tilde{u}_j^{S}\right)^2}, \quad S = 1, 2, 3, \ldots$
- **End while**

Finally, the algorithm provides a non-uniform distribution of points. In this distribution, the density of points in regions with high gradients is higher and in other regions less (Fig.1). It should be noted that at any step, a unit is added to d . This addition has led to a more accurate performance of the algorithm.

Finding the optimal shape parameter using the genetic algorithm

The shape parameter is one of the most important variables in the accuracy and stability of the MQ method. Researchers have presented several experimental formulas for this parameter [32-38]. So far, no general theory has been developed for this parameter. Researchers have optimized this parameter over the past decades with a variety of classical and modern optimization methods [39-46]. One of these new optimization methods is the genetic algorithm which researchers have used to optimize this parameter and achieve accurate results [47-50]. The main advantage of the genetic algorithm over classical algorithms is that no information from the variable can optimize the optimization variable. This feature is very useful for a shape parameter that lacks an acceptable theory. Another feature of this algorithm is that it can find the absolute optimal and avoid being trapped in a local optimal solution. The pseudo code of the general GA is as follows:

Algorithm 1 . The pseudo code of the general GA [50]

1. Generating an initial population;
2. Evaluating fitness of individuals in the population;

While population has converged **do**

- Selecting parents from the population;
- Recombining (mate (crossover and mutation operators)) parents to produce children;
- Evaluating fitness of the children;
- Replacing some or all of the population by the children;

end while

In this research, root mean square error (RMS) is selected as the fitness function of the GA, Eq. (19). The detailed characteristics of GA in the present research are shown in Table 2 [50].

Numerical results

In this section, the efficiency of the described algorithm will be investigated to solve the nearly singular problems. The examples have internal and boundary singular points. To measure the accuracy of the results, two types of norm error were used. the L_∞ and RMSE defined as

$L_\infty = max\left|u(x_i) - \tilde{u}(x_i)\right|$, (18)

$RMSE = \sqrt{\frac{1}{N_t}\sum_{i=1}^{N_t}\left(u(x_i) - \tilde{u}(x_i)\right)^2}$(19)

where $u(x)$ is the exact solution, $\tilde{u}(x)$ is the approximate solution and N_t is the total number of evaluation points. In the singular regions, the node density is controlled by δ. The delta is determined based on the value of demanded accuracy. In all tables, N and c are the center points and shape parameter, respectively.

Example 1. The following Poisson equation with Dirichlet boundary conditions has been considered in the rectangle domain$\Omega = [\mathbf{0}, \mathbf{1}]$

$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (x, y) \in \Omega$(20)

$u\left(x, y\right) = s\left(x, y\right), \quad (x, y) \in \partial\Omega$(21)

where f and s are determined based on the exact solution [25]:

$u\left(x, y\right) = \exp\left(-200x^2\right) + \exp\left(-200y^2\right)$ (22)

The adaptive-refinement node algorithm was implemented using the MQ solution. The variable δ is assumed 1e-3. Nodes distribution in different steps of the algorithm are shown in Fig. 2.

The exact solution shows that there are high gradients around the axes x=0 and y=0. Outside the region, the gradient is very low and the function is smooth. Because of this high gradient, an exact numerical approximation is not easy therefore the density of points in these regions must be increased by the adaptive-node correction algorithm.

The initial and adaptive distributions are shown in Fig. 2. The number of center points starts with 19×19=361 and gradually grows to N=742 nodes then the algorithm results in N=1281 nodes. As the figure shows, the points can track high gradients quite efficiently.

The points distribution was used to solve Eq. (20) using the MQ solution. The number of center points, the optimal shape parameter, RMS error and L_∞ at each function are given in Table 3. A uniform distribution containingN_t=2500$(= 50 \times 50)$ points evenly spaced in the domain is used to evaluate the L_∞ and RMSE error norms in the test function. Applying the proposed adaptive algorithm resulted in N=1281 nodes and RMS=6.936e-5 after three steps of adaptation, while Libre et al. [25] Achieved lower accuracy using a wavelet scheme with N=1823 nodes and after four steps of adaptation. Munoz-Gomez et al. [23] Achieved this accuracy after 12 iterations and with 2646 points The numerical results with uniform and adaptive distributions are shown in Fig. 3.

Example 2. The Poisson equation with an internal pointwise near singularity in a rectangle domain$\Omega = [0, 1]^2$is

$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (x, y) \in \Omega$ (23)

$u\left(x, y\right) = s\left(x, y\right), \quad (x, y) \in \partial\Omega$ (24)

where f and s are determined based on the exact solution [25]:

$u\left(x, y\right) = \frac{\left(x^3+y^3\right)}{2} + exp\left(\frac{-(x-0.5)^2}{0.05^2} - \frac{(y-0.5)^2}{0.05^2}\right)$ (25)

The exact solution and the node distributions at each step of the algorithm are shown in Fig. 4. The steps of the algorithm show that the density of the nodes around the center of the domain and the axes x=0 and axes y=0 increases and the figure of the exact solution confirms that these regions have higher gradients than other regions. As may be seen the number of points has been increased to 853 in the first refinement and again has been decreased to 743 in the second refinement procedure which means that the number of points may have some oscillations to reach a stable condition for a specific error value.

Applying the proposed adaptive algorithm with δ=1e-4 is resulted in N=743 points and RMS error less than 4.7E-5. This accuracy was achieved by Libre et al. (2009) with 1823 center points using the wavelet adaptive algorithm. The RMS error and L_∞ error norm are summarized in Table 4. A uniform distribution

containingN_t=2500($=50\times50$) points evenly spaced in the domain is used to evaluate the L_∞ and RMSE error norms in the test function.

The results of this example show that the proposed algorithm is able to detect singularities in different regions of the domain and it also does the node refinement correctly while some adaptation algorithms [15,16] are only able to detect singularities in boundary regions.

Example 3. A 2D modified Helmholtz equation defined in $\Omega=[-1,1]$as

$$u_{xx}+u_{yy}-u(x,y)=f(x,y),\qquad (x,y)\in\Omega \quad (26)$$

$$u\left(x,y\right)=s\left(x,y\right),\qquad (x,y)\in\partial\Omega \quad (27)$$

where f and s are determined based on the exact solution [49].

$$u\left(x,y\right)=\exp^{(-100x^2-100y^2)} \quad (28)$$

The function of this example has a steep point in the center of the domain and is perfectly smooth in other regions. These changes of function cause the singularity of the interpolation matrix and it is necessary to modify the node distribution with adaptive algorithms. The exact solution and the adaptive distribution at each step are displayed in Fig. 5. The algorithm starts with a uniform distribution 19×19=361 nodes, and in the next steps of the algorithm, the number of nodes will gradually increase and finally with δ=1e-3 reaches 573 nodes after 3 iterations with RMS error less than 5×10^{-4}, while Esmaeilbeigi et al. [49] Achieved lower accuracy using a dynamic node adaptive with N=1241 nodes and after four steps of adaptation.

The number of nodes, optimal shape parameter, RMS error andL_∞ at each step of the algorithm is given in Table 5. The numerical results of a uniform and an adaptive distribution are compared in Fig. 6.

Conclusion

In this paper, a new adaptive node refinement algorithm to solve the MQ approximations of nearly singular PDEs has been proposed. For the first time, the concept of gradient of the estimation function has been used in refining the node arrangement. In the framework of the adaptive algorithm, the proposed formulation was used as the parameter that determine where the node distribution should be refined. Therefore, the number of center points is optimized and the approximation matrix of MQ method goes out of the ill-condition. The performance of the algorithm was investigated through various examples. Numerical results reveal that this algorithm is able to identify the interior and boundary singular regions correctly and achieves very good performance in terms of accuracy, convergence speed and CPU time-saving. One of the most advantages of the adaptive node refinement algorithm is that it is simple to be implemented and has less computational cost in comparison with other adaptive schemes. Although the present research considers the MQ approximation and the solution of PDEs with domain or boundary approximate singularities, the advantages of this adaptive algorithm may also be extended to other approximation methods and any other types of linear or non-linear boundary value problems in even higher dimensions.

References

[1] Powell, M. J. (1987). Radial basis functions for multivariable interpolation: a review. Algorithms for approximation.

[2] Yee, P. V., & Haykin, S. (2001). Regularized radial basis functional networks: theory and applications. John Wiley & Sons, Inc.

[3] Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., & Evans, T. R. (2001, August). Reconstruction and representation of 3D objects with radial basis functions. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (pp. 67-76). ACM.

[4] Ohtake, Y., Belyaev, A., & Seidel, H. P. (2004, June). 3D scattered data approximation with adaptive compactly supported radial basis functions. In Proceedings Shape Modeling Applications, 2004. (pp. 31-39). IEEE.

[5] Kansa, E. J. (1990). Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics—II solutions to parabolic, hyperbolic and elliptic partial differential equations. Computers & mathematics with applications, 19(8-9), 147-161.

[6] Larsson, E., & Fornberg, B. (2003). A numerical study of some radial basis function based solution methods for elliptic PDEs. Computers & Mathematics with Applications, 46(5-6), 891-902.

[7] Buhmann, M. D. (2003). Radial basis functions: theory and implementations (Vol. 12). Cambridge university press.

[8] Buhmann, M., & Dyn, N. (1993). Spectral convergence of multiquadric interpolation. Proceedings of the Edinburgh Mathematical Society, 36(2), 319-333.

[9] Driscoll, T. A., & Heryudono, A. R. (2007). Adaptive residual subsampling methods for radial basis function interpolation and collocation problems. Computers & Mathematics with Applications, 53(6), 927-939.

[10] Runge, C. (1901). Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. Zeitschrift für Mathematik und Physik, 46(224-243), 20.

[11] Boyd, J. P. (2005). Trouble with Gegenbauer reconstruction for defeating Gibbs' phenomenon: Runge phenomenon in the diagonal limit of Gegenbauer polynomial approximations. Journal of Computational Physics, 204(1), 253-264.

[12] Fornberg, B., & Zuev, J. (2007). The Runge phenomenon and spatially variable shape parameters in RBF interpolation. Computers & Mathematics with Applications, 54(3), 379-398.

[13] Schaback, R., & Wendland, H. (2000). Adaptive greedy techniques for approximate solution of large RBF systems. Numerical Algorithms, 24(3), 239-254.

[14] Hon, Y. C., Schaback, R., & Zhou, X. (2003). An adaptive greedy algorithm for solving large RBF collocation problems. Numerical Algorithms, 32(1), 13-25.

[15] Hon, Y. C. (1999). Multiquadric collocation method with adaptive technique for problems with boundary layer. Int. J. Appl. Sci. Comput, 6(3), 173-184.

[16] Ling, L., & Trummer, M. R. (2006). Adaptive multiquadric collocation for boundary layer problems. Journal of Computational and Applied Mathematics, 188(2), 265-282.

[17] Sarra, S. A. (2005). Adaptive radial basis function methods for time dependent partial differential equations. Applied Numerical Mathematics, 54(1), 79-94.

[18] Wu, Z. (2004). Dynamically knots setting in meshless method for solving time dependent propagations equation. Computer Methods in Applied Mechanics and Engineering, 193(12-14), 1221-1229.

[19] Wu, Z. (2005). Dynamical knot and shape parameter setting for simulating shock wave by using multi-quadric quasi-interpolation. Engineering analysis with boundary elements, 29(4), 354-358.

[20] Bozzini, M., Lenarduzzi, L., & Schaback, R. (2002). Adaptive interpolation by scaled multiquadrics. Advances in Computational Mathematics, 16(4), 375-387.

[21] Behrens, J., & Iske, A. (2002). Grid-free adaptive semi-Lagrangian advection using radial basis functions. Computers & mathematics with applications, 43(3-5), 319-327.

[22] Behrens, J., Iske, A., & Käser, M. (2003). Adaptive meshfree method of backward characteristics for nonlinear transport equations. In Meshfree methods for partial differential equations (pp. 21-36). Springer, Berlin, Heidelberg.

[23] Munoz-Gomez, J. A., Gonzalez-Casanova, P., & Rodriguez-Gomez, G. (2006, September). Adaptive node refinement collocation method for partial differential equations. In 2006 Seventh Mexican International Conference on Computer Science (pp. 70-80). IEEE.

[24] González-Casanova, P., Munoz-Gomez, J. A., & Rodriguez-Gomez, G. (2009). Node adaptive domain decomposition method by radial basis functions. Numerical Methods for Partial Differential Equations: An International Journal, 25(6), 1482-1501.

[25] Libre, N. A., Emdadi, A., Kansa, E. J., Shekarchi, M., & Rahimian, M. (2009). A multiresolution prewavelet-based adaptive refinement scheme for RBF approximations of nearly singular problems. Engineering analysis with boundary elements, 33(7), 901-914.

[26] Libre, N. A., Emdadi, A., Kansa, E. J., Shekarchi, M., & Rahimian, M. (2008). A fast adaptive wavelet scheme in RBF collocation for nearly singular potential PDEs.

[27] Shanazari, K., & Hosami, M. (2012). A two-dimensional adaptive nodes technique in irregular regions applied to meshless-type methods. Engineering Analysis with Boundary Elements, 36(2), 161-168.

[28] Esmaeilbeigi, M., & Hosseini, M. M. (2012). Dynamic node adaptive strategy for nearly singular problems on large domains. Engineering Analysis with Boundary Elements, 36(9), 1311-1321.

[29] Roque, C. M. C., Madeira, J. F. A., & Ferreira, A. J. M. (2015). Multiobjective optimization for node adaptation in the analysis of composite plates using a meshless collocation method. Engineering Analysis with Boundary Elements, 50, 109-116.

[30] Kamranian, M., Dehghan, M., & Tatari, M. (2017). An adaptive meshless local Petrov–Galerkin method based on a posteriori error estimation for the boundary layer problems. Applied Numerical Mathematics, 111, 181-196.

[31] Cavoretto, R., & De Rossi, A. (2019). Adaptive meshless refinement schemes for RBF-PUM collocation. Applied Mathematics Letters, 90, 131-138.

[32] Hardy, R.L. (1971). "Multiquadric equations of topography and other irregular surfaces ", J. Geophysics. Res, 176,1905-1915.

[33] Franke, R. (1979). "***A critical comparison of some methods for interpolation of scattered data***" , Naval Postgraduate school Monterey, No. NPS53-79-003.

[34] Micchelli, C. A. (1984). "Interpolation of scattered data: distance matrices and conditionally positive definite functions ", In *Approximation theory and spline functions* (Springer), Dordrecht, pp:143-145.

[35] Madych, W. R., & Nelson, S. A. (1983). "Multivariate interpolation: a variational theory ".*unpublished preprint* .

[36] Hardy, R. L. (1990). "Theory and applications of the multiquadric-biharmonic method 20 years of discovery 1968–1988 ",*Computers & Mathematics with Applications* , Vol.19, No:8-9, pp:163-208.

[37] Kansa, E. J. (1990). "Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics—I surface approximations and partial derivative estimates ", *Computers & Mathematics with applications* , Vol: 19, No:8-9, pp:127-145.

[38] Kansa, E. J. (1990). "Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics—II solutions to parabolic, hyperbolic and elliptic partial differential equations ", *Computers & mathematics with applications* , Vol: 19, No:8-9, pp:147-161.

[39] Rippa, S. (1999). An algorithm for selecting a good value for the parameter c in radial basis function interpolation. Advances in Computational Mathematics, 11(2-3), 193-210.

[40] Cheng, A. D. (2012). Multiquadric and its shape parameter—a numerical investigation of error estimate, condition number, and round-off error by arbitrary precision computation. Engineering analysis with boundary elements, 36(2), 220-239.

[41] Fallah, A., Jabbari, E., & Babaee, R. (2019). Development of the Kansa method for solving seepage problems using a new algorithm for the shape parameter optimization. Computers & Mathematics with Applications, 77(3), 815-829.

[42] Buhmann, M. D. (2003). Radial basis functions: theory and implementations (Vol. 12). Cambridge university press.

[43] Tsai, C. H., Kolibal, J., & Li, M. (2010). The golden section search algorithm for finding a good shape parameter for meshless collocation methods. Engineering Analysis with Boundary Elements, 34(8), 738-746.

[44] Huang, C. S., Lee, C. F., & Cheng, A. D. (2007). Error estimate, optimal shape factor, and high precision computation of multiquadric collocation method. Engineering Analysis with Boundary Elements, 31(7), 614-623.

[45] Wertz, J., Kansa, E. J., & Ling, L. (2006). The role of the multiquadric shape parameters in solving elliptic partial differential equations. Computers & Mathematics with Applications, 51(8), 1335-1348.

[46] Uddin, M. (2014). On the selection of a good value of shape parameter in solving time-dependent partial differential equations using RBF approximation method. Applied Mathematical Modelling, 38(1), 135-144.

[47] Deb, K. (1995). Optimization for engineering design: algorithms and examples.Prentice-Hall, New Delhi.

[48] Deb, K. (1999). An introduction to genetic algorithms. Sadhana, 24(4-5), 293-315.

[49] Esmaeilbeigi, M., & Hosseini, M. M. (2014). A new approach based on the genetic algorithm for finding a good shape parameter in solving partial differential equations by Kansa's method. *Applied Mathematics and Computation* , *249* , 419-428.

[50] Koushki, M., Jabbari, E., & Ahmadinia, M. (2020). Evaluating RBF methods for solving PDEs using Padua points distribution.*Alexandria Engineering Journal* .

Tables

Table1. Common RBFs

RBF	Definition
Multiquadric (MQ)	$\phi(r,c)=\sqrt{1+c^2r^2}$
Inverse Multiquadric (IMQ)	$\phi(r,c)=\frac{1}{\sqrt{1+c^2r^2}}$
Gaussian (GA)	$\phi(r,c)=e^{-\left(cr^2\right)}$
This plate spline (TPS)	$\phi(r)=r^2\text{logr}$
Cubic RBF	$\phi(r)=r^3$

Table 2. Characteristics of GA in the present research.

Population size	Elite count	Crossover function	Mutation	Crossover	Migration	Generation
20	1	0.8	Uniform	Two points	Forward	100

Table 3. Comparison of accuracies of MQ solutions in Example 1.

L_∞	RMS	C	N
4.46e-2	8.21e-1	2.540	361
2.763e-3	3.5e-2	8.297	742
3.610e-3	6.936e-5	11.337	1281

Table 4. Comparison of accuracies of MQ solutions in Example 2.

L_∞	RMS	C	N
5.722e-1	9.09e0	2.541	361
1.504e-2	8.301e-2	9.351	853
1.164e-3	4.612e-5	13.351	743

Table 5. Comparison of accuracies of MQ solutions in Example 3.

L_∞	RMS	C	N
4.477e-1	9.335e-1	2.115	361
7.561e-3	3.273e-2	2.569	544
1.015e-4	3.008e-4	3.219	573

Hosted file

`Figures.docx` available at https://authorea.com/users/339904/articles/467029-a-new-adaptive-node-refinement-algorithm-based-on-multiquadric-method-for-the-nearly-singular-problems