Neural, Parallel, and Scientific Computations 25 (2017) 135-148

THE METHOD OF PARTICULAR SOLUTIONS FOR FINDING CRITICAL DOMAINS FOR QUENCHING PROBLEMS

THIR DANGAL* AND C. S. CHEN*†‡

*Department of Mathematics, University of Southern Mississippi
Hattiesburg, MS 39406, USA
†College of Mathematics, Taiyuan University of Technology, Taiyuan, China

ABSTRACT. In this work, the method of particular solutions (MPS) has been used for solving nonlinear Poisson-type problems defined on different geometries. The polyharmonic splines is used as the basis function so that no shape parameter is needed in the solution process. The MPS is also applied to compute the sizes of critical domains of different shapes for a quenching problem and compared with the sizes of critical domains obtained from some other numerical methods. Numerical examples are presented to show the efficiency and accuracy of the method.

AMS (MOS) Subject Classification. 39A10.

1. INTRODUCTION

The nonlinear partial differential equation is one of the most active fields of recent research. Most of the real world problems, including gas dynamics, fluid mechanics, elasticity, relativity and many more, are modeled by the nonlinear partial differential equations. In general, the exact solutions of many of such problems are not available. Therefore, numerical approximation has to be done to find the solutions of the problems. There are many numerical schemes to solve nonlinear problems. Some of them are the finite element method (FEM) [15], the finite difference method (FDM) [14], the boundary element method (BEM) [12] and the meshless methods [2, 6, 11]. The mesh based methods like the FEM and the FDM require extensive work of mesh generation of the computational domain, resulting large computational resources to solve the problem. Furthermore, the generation of the mesh for the irregular domain is non trivial. The radial basis function (RBF) collocation methods are meshless methods which are able to overcome these drawbacks. In this paper, we have used the MPS [8, 6, 17], one of the RBF collocation methods, to solve some nonlinear Poisson-type problems defined on regular and irregular domains. In addition, the MPS is also applied to compute the size of the critical domains for the quenching problems [5]. The MPS is modified to make the method more efficient.

‡Corresponding author: cschen.math@gmail.com

Received June 30, 2016

Consider the initial boundary value problem

$$\Delta u - \frac{\partial u}{\partial t} = -f(t,u) \text{ in } \mathbf{\Omega}, \tag{1.1}$$

$$u(\boldsymbol{x},0) = u_0(\boldsymbol{x}) \text{ on } \bar{\mathbf{D}}, \tag{1.2}$$

$$u(\boldsymbol{x},t) = 0 \text{ on } \mathbf{S}, \tag{1.3}$$

where $\mathbf{D} \in \mathbb{R}^m$ is a bounded convex domain, $\mathbf{\Omega} = \mathbf{D} \times (0,\mathbf{T})$, $\mathbf{S} = \partial\mathbf{D} \times (0,\mathbf{T})$, and $\mathbf{T} \leq \infty$ such that

$$\lim_{u \to c^-} f(t,u) = \infty,$$

for some positive constant c. This kind of problem was first studied by Kawarada [13] in 1975. The solution u is said to quench if there exists a finite time $\mathbf{T}$ such that

$$\sup\left\{|\mathrm{u_t}(\boldsymbol{x},\mathrm{t})| : \boldsymbol{x} \in \bar{\mathbf{D}}\right\} \to \infty \text{ as } \mathrm{t} \to \mathbf{T}^-. \tag{1.4}$$

The time at which the quenching occurs is called the quenching time. When u is an increasing function of t, a necessary condition for (1.4) is

$$\max\left\{\mathrm{u}(\boldsymbol{x},\mathrm{t}) : \boldsymbol{x} \in \bar{\mathbf{D}}\right\} \to \mathrm{c}^- \text{ as } \mathrm{t} \to \mathbf{T}^-. \tag{1.5}$$

The point in the Euclidian space where the solution u reaches c is called quenching point. For one dimensional case, Acker and Kawohl [1] proved that the origin is the only quenching point for the problem (1.1)–(1.3) when the domain is a ball with center at origin. Also, Deng and Levine [9] extended the results from balls to convex domain $\mathbf{D}$ with smooth boundary $\partial\mathbf{D}$. They showed that the quenching points are in a compact subset of $\mathbf{D}$. In 1994, Chan and Ke [5] studied the critical domains and developed a method to find the size of such domain for the problem (1.1)–(1.3) defined on a domain with piecewise smooth boundary. A critical domain $\mathbf{D}^*$ is a domain such that the solution exists for all domain $\mathbf{D}$ at all time when $\mathbf{D} \subset \mathbf{D}^*$ and the quenching always occurs at finite time $\mathbf{T}$ in $\mathbf{D} \supseteq \mathbf{D}^*$. With $u_0 \equiv 0$, Chan and Ke [5] proved that a unique critical domain exists for each shape of domain of the problem (1.1)–(1.3). This means distinct shape of domain has distinct critical domain size. The size of the domain is specified by area in 2D and volume for 3D or higher. In 2007, Tian [16] applied a numerical method using Delta-shaped basis function to compute the critical domains for quenching problems and compared the results with the results already established with some other computational methods.

Let $\mathbf{D} \subseteq \mathbb{R}^m$ and $\mathbf{D}_1 \subseteq \mathbb{R}^m$. They will have same shape if there exists $\boldsymbol{x}_0 \in \mathbf{D} \cap \mathbf{D}_1$ and a positive constant λ such that

$$\mathbf{D}_1 = \{\boldsymbol{y} : \boldsymbol{y} = \boldsymbol{x}_0 + \lambda(\boldsymbol{x} - \boldsymbol{x}_0)\} \text{ for } \boldsymbol{x} \in \mathbf{D}. \tag{1.6}$$

If the problem (1.1)–(1.3) is defined on the domain $\mathbf{D}_1$ with known shape and x_0 is at the origin, we can transform this problem into a problem defined in different sized domain $\mathbf{D}$ having the same shape in the following way:

$$\Delta u_\lambda - \frac{\partial u_\lambda}{\partial t} = -\lambda^2 f(\lambda^2 t, u_\lambda) \text{ in } \mathbf{\Omega}, \tag{1.7}$$

$$u_\lambda = 0 \text{ on } \bar{\mathbf{D}} \cup \mathbf{S}, \tag{1.8}$$

where (cf. Chan and Ke [5]),

$$\lambda = \left(\frac{\text{size of } \mathbf{D}_1}{\text{size of } \mathbf{D}} \right)^{1/m}. \tag{1.9}$$

From (1.9), we see that the size of the critical domain is determined by λ. The modified MPS has been used to compute λ.

The paper is organized as follows. In Section 2, a brief review for the MPS for solving nonlinear Poisson-type problems with Dirichlet's boundary condition is given. In Section 3, an algorithm for the computation of quenching problems is given. In Section 4, some numerical examples for solving nonlinear Poisson-type problems are given, and critical domains for quenching problems are computed and compared to the results already established by some other numerical methods. Conclusion is given in Section 5.

2. THE METHOD OF PARTICULAR SOLUTIONS (MPS)

In this section, we give a brief review of the MPS [8, 6, 17] using polyharmonic splines. For simplicity, let us consider the following Poisson problem in 2D

$$\Delta u(x,y) = f(x,y), \quad (x,y) \in \mathbf{\Omega}, \tag{2.1}$$

$$u(x,y) = g(x,y), \quad (x,y) \in \partial\mathbf{\Omega}, \tag{2.2}$$

where $\mathbf{\Omega} \subseteq \mathbb{R}^2$ is a bounded and closed domain with boundary $\mathbf{\Omega}$, f and g are known functions.

Let ϕ be a radial basis function (RBF) and $\{p_l\}_{l=1}^{w}$ be a basis of $\mathbf{P}_m$, the set of two dimensional polynomials of degree $\leq m$ with $w = (m+1)(m+2)/2$. Let $\{(x_i,y_i)\}_{i=1}^{n}$ be a set of pairwise distinct interpolation points with $\{(x_i,y_i)\}_{i=1}^{n_i} \subseteq \mathbf{\Omega}$ and $\{(x_i,y_i)\}_{i=n_i+1}^{n} \subseteq \partial\mathbf{\Omega}$ such that $n = n_i + n_b$.

The MPS has been widely used in the context of RBFs with shape parameter such as MQ, inverse MQ, Gaussian. Recently, Yao *et al.* [18] has extended the MPS using polyharmonic splines with augmented polynomials as the basis function as follows

$$u(x,y) \simeq \hat{u}(x,y) = \sum_{j=1}^{n} a_i \Phi(r) + \sum_{l=1}^{w} a_{n+l} p_l(x,y), \quad (x,y) \in \mathbf{\Omega}, \tag{2.3}$$

where $r = \|(x,y) - (x_j,y_j)\|$ and

$$\Delta\Phi(r) = r^{2k} \ln r, \quad k \in \mathbb{Z}^+. \tag{2.4}$$

Note that Φ in (2.4) can be easily obtained through repeated integration in the polar coordinates. It follows that

$$\Phi(r) = \frac{r^{2k+2}}{4(k+1)^2}\ln r - \frac{r^{2k+2}}{4(k+1)^3}. \tag{2.5}$$

Furthermore,

$$f(x,y) \approx \Delta\hat{u}(x,y) = \sum_{j=1}^{n} a_j r^{2n}\ln r + \sum_{l=1}^{w} a_{n+l} q_l(x,y), \quad (x,y)\in\mathbf{\Omega}, \tag{2.6}$$

where

$$\Delta p_l(x,y) = q_l(x,y), \quad l = 1,2,\ldots,w.$$

In addition, the following augmented equation needs to be imposed [10]

$$\sum_{j=1}^{n} a_j p_l(x,y) = 0, \quad l = 1,2,\ldots,w. \tag{2.7}$$

Using (2.3) and (2.7), we can establish the following matrix system

$$\begin{bmatrix} \hat{\mathbf{u}} \\ \mathbf{0}_w \end{bmatrix} = \begin{bmatrix} A_\Phi & \mathbf{P}_{nw} \\ \mathbf{P}_{nw}^T & \mathbf{0}_{ww} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{n+w} \end{bmatrix}, \tag{2.8}$$

where $\mathbf{0}_w$ is the zero matrix of order $w \times 1$, $\hat{\mathbf{u}} = [\hat{u}(x_1,y_1),\ldots,\hat{u}(x_n,y_n)]^T$, $A_\Phi = [\Phi(r_{ij})]_{1\le i,j\le n}$ and $\mathbf{P}_{il} = p_l(x_i,y_i), \ \ i = 1,2,\ldots,n$. From (2.8), we have

$$\mathbf{a} = \begin{bmatrix} A_\Phi & \mathbf{P}_{nw} \\ \mathbf{P}_{nw}^T & \mathbf{0}_{ww} \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{u}} \\ \mathbf{0}_w \end{bmatrix}, \tag{2.9}$$

where $\mathbf{a} = [a_1, a_2, \ldots, a_{n+w}]^T$. By collocating the interior points, we have

$$f(x_i,y_i) = \sum_{j=1}^{n} a_j r^{2k}\ln r + \sum_{l=1}^{w} a_{n+l} q_l(x_i,y_i), \quad i = 1,2,\ldots,n_i. \tag{2.10}$$

with the additional augmented equations

$$\sum_{j=1}^{n_i} a_j q_l(x_j,y_j) = 0, \quad l = 1,2,\ldots,w. \tag{2.11}$$

Similarly, by collocating the boundary points, we have

$$g(x_i,y_i) = \sum_{j=1}^{n} a_j \Phi(r_{ij}) + \sum_{l=1}^{w} a_{n+l} p_l(x_i,y_i), \quad i = n_i+1,\ldots,n. \tag{2.12}$$

with additional augmented conditions

$$\sum_{j=n_i+1}^{n} a_j p_l(x_j,y_j) = 0, \quad l = 1,2,\ldots,w. \tag{2.13}$$

From (2.10)–(2.13), we have the following block matrix system

$$\begin{bmatrix} \phi_{n_i n} & \mathbf{Q}_{n_i w} \\ \Phi_{n_b n} & \mathbf{P}_{n_b w} \\ [\mathbf{Q}^T_{n_i w}, \mathbf{P}^T_{n_b w}] & \mathbf{0}_{ww} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{n+w} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{n_i} \\ \mathbf{g}_{n_b} \\ \mathbf{0}_w \end{bmatrix}. \tag{2.14}$$

Using (2.14), (2.9) becomes

$$\begin{bmatrix} \phi_{n_i n} & \mathbf{Q}_{n_i w} \\ \Phi_{n_b n} & \mathbf{P}_{n_b w} \\ [\mathbf{Q}^T_{n_i w}, \mathbf{P}^T_{n_b w}] & \mathbf{0}_{ww} \end{bmatrix} \begin{bmatrix} A_\Phi & \mathbf{P}_{nw} \\ \mathbf{P}^T_{nw} & \mathbf{0}_{ww} \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{u}} \\ \mathbf{0}_{ww} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{n_i} \\ \mathbf{g}_{n_b} \\ \mathbf{0}_w \end{bmatrix}. \tag{2.15}$$

We can rewrite the equation (2.15) as

$$\begin{bmatrix} \phi_{n_i n_i} & \phi_{n_i n_b} & \mathbf{Q}_{n_i w} \\ \Phi_{n_b n_i} & \Phi_{n_b n_b} & \mathbf{P}_{n_b w} \\ \mathbf{Q}^T_{n_i w} & \mathbf{P}^T_{n_b w} & \mathbf{0}_{ww} \end{bmatrix} \begin{bmatrix} \Phi_{n_i n_i} & \Phi_{n_i n_b} & P_{n_i w} \\ \Phi_{n_b n_i} & \Phi_{n_b n_b} & P_{n_b w} \\ P^T_{n_i w} & P^T_{n_b w} & \mathbf{0}_{ww} \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{u}}_{n_i} \\ \hat{\mathbf{u}}_{n_b} \\ \mathbf{0}_w \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{n_i} \\ \mathbf{g}_{n_b} \\ \mathbf{0}_w \end{bmatrix}. \tag{2.16}$$

Let

$$A = \begin{bmatrix} \phi_{n_i n_i} & \phi_{n_i n_b} & \mathbf{Q}_{n_i w} \\ \Phi_{n_b n_i} & \Phi_{n_b n_b} & \mathbf{P}_{n_b w} \\ \mathbf{Q}^T_{n_i w} & \mathbf{P}^T_{n_b w} & \mathbf{0}_{ww} \end{bmatrix}, \quad B = \begin{bmatrix} \Phi_{n_i n_i} & \Phi_{n_i n_b} & P_{n_i w} \\ \Phi_{n_b n_i} & \Phi_{n_b n_b} & P_{n_b w} \\ P^T_{n_i w} & P^T_{n_b w} & \mathbf{0}_{ww} \end{bmatrix}.$$

Since the matrices A and B have same second row, Lemma 1 in the Appendix implies that the second row of matrices AB^{-1} and BB^{-1} are equal. Thus, the product matrix AB^{-1} takes the form

$$AB^{-1} = \begin{bmatrix} \mathbf{C}_{n_i n_i} & \mathbf{C}_{n_i n_b} & \mathbf{C}_{n_i w} \\ \mathbf{0}_{n_b n_i} & \mathbf{I}_{n_b n_b} & \mathbf{0}_{n_b w} \\ \mathbf{C}_{w n_i} & \mathbf{C}_{w n_b} & \mathbf{C}_{ww} \end{bmatrix}. \tag{2.17}$$

Consequently, from (2.16) and (2.17), we have

$$\begin{bmatrix} \mathbf{C}_{n_i n_i} & \mathbf{C}_{n_i n_b} & \mathbf{C}_{n_i w} \\ \mathbf{0}_{n_b n_i} & \mathbf{I}_{n_b n_b} & \mathbf{0}_{n_b w} \\ \mathbf{C}_{w n_i} & \mathbf{C}_{w n_b} & \mathbf{C}_{ww} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}_{n_i} \\ \hat{\mathbf{u}}_{n_b} \\ \mathbf{0}_w \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{n_i} \\ \mathbf{g}_{n_b} \\ \mathbf{0}_w \end{bmatrix}. \tag{2.18}$$

Note that (2.9) can be further reduced to the following matrix system

$$\begin{bmatrix} \mathbf{C}_{n_i n_i} & \mathbf{C}_{n_i n_b} \\ \mathbf{0}_{n_b n_i} & \mathbf{I}_{n_b n_b} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}_{n_i} \\ \hat{\mathbf{u}}_{n_b} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{n_i} \\ \mathbf{g}_{n_b} \end{bmatrix}. \tag{2.19}$$

The equation (2.19) implies that

$$\mathbf{C}_{n_i n_i} \hat{\mathbf{u}}_{n_i} + \mathbf{C}_{n_i n_b} \hat{\mathbf{u}}_{n_b} = \mathbf{f}_{n_i}, \tag{2.20}$$

$$\hat{\mathbf{u}}_{n_b} = \mathbf{g}_{n_b}. \tag{2.21}$$

Substituting (2.21) into (2.20), we can further reduce to the following system

$$\hat{\mathbf{u}}_{n_i} = \mathbf{C}^{-1}_{n_i n_i} (\mathbf{f}_{n_i} - \mathbf{C}_{n_i n_b} \mathbf{g}_{nb}). \tag{2.22}$$

It is obvious that solving (2.22) is more efficient than solving (2.18).

Since the problems we considered are nonlinear, we used Picard method to carry out the iteration [7]. The iteration is summarized as follows:

Algorithm 1

Step 1:	`Construct a sequence` $\left\{\mathbf{u}^{(i)} : i \in \mathbb{N} \cup \{0\}\right\}$ `such that` $\mathbf{u}^{(0)} = 0$ `and` $\Delta \mathbf{u}^{(i+1)} = \mathbf{f}(\mathbf{u}^{(i)})$ *in* $\mathbf{\Omega}$, $\mathbf{u}^{(i+1)} = \mathbf{g}$ *on* $\partial\mathbf{\Omega}$.
Step 2:	`At each iteration, compute` $\mathbf{u}^{(i+1)}$ `using modified MPS.`
Step 3:	`If` $\left\|\mathbf{u}^{(i+1)} - \mathbf{u}^{(i)}\right\| < \epsilon,\ \ \epsilon > 0$, `stop,` `end.`
Step 4:	$\mathbf{u}^{(i+1)}$ `at the final iteration is the required approximate solution.`

The above mentioned MPS can be extended to more general equations containing variable coefficients or three dimensional cases. We refer readers to [6] for more details.

3. CRITICAL DOMAINS FOR QUENCHING PROBLEMS

In this section, we apply the MPS to find the critical domain for the quenching problem (1.7)–(1.8). We suppose $f(u) = 1/(1-u)$. Then, we have,

$$\Delta u - u_t = -\frac{\gamma}{1-u} \text{ in } \Omega, \tag{3.1}$$

$$u = 0 \text{ on } \bar{\mathbf{D}} \cup S, \tag{3.2}$$

where $\gamma = \lambda^2$. The critical value of γ is obtained from the steady state form of the above problem. We have,

$$\Delta U = -\frac{\gamma}{1-U} \text{ in } \mathbf{D}, \tag{3.3}$$

$$U = 0 \text{ on } \partial\mathbf{D}. \tag{3.4}$$

Once we compute the critical value γ^* of γ, we can use equation (1.9) to compute critical size of the considered domain. Here, we present an algorithm to compute the critical value γ^*.

Algorithm 2

Step 1:	`The upper bound` γ_{upper} `for` γ^* `is obtained from the solution of the problem` $\Delta\xi = -1$ in $\mathbf{D}$, $\xi = 0$ on $\partial\mathbf{D}$. `We have,` $\gamma_{upper} = 1/max_{\bar{\mathbf{D}}}\xi$ `and choose` $\gamma_{lower} = 0$. `Estimate` γ^* `with` $\gamma^{(1)} = (\gamma_{upper} - \gamma_{lower})/2$.

Step 2:	`For each` $\gamma^{(k)}$`, Compute` $\{U^{(i)}\}$ `defined by`
	$U^{(0)} = 0 \;\; on \;\; \bar{\mathbf{D}}$,
	`and for` $i \geq 1$,
	$\Delta U^{(i)} = -\frac{\gamma^k}{1-U^{(i-1)}} \;\; in \;\; \mathbf{D}$,
	$U^{(i)} = 0 \;\; on \;\; \partial\mathbf{D}$.
	`For each` $\gamma^{(k)}$`, Compute` $\{U^{(i)}\}$ `by the modified MPS.`
	`If` $\{U^{(i)}\}$ `converges, the corresponding` $\gamma^{(k)}$ `is the lower bound for` γ^*`; otherwise, it is an upper bound.`

Step 3:	`If` $\left|\gamma^{(k-1)} - \gamma^{(k-2)}\right| < \epsilon, \;\; \epsilon > 0$, `(a given tolerance), then` $\gamma^{(k-1)}$ `is the approximate critical value.`

Step 4:	`Update` $\gamma^{(k-1)}$ `in the following way:`
	`If` $\{U^{(i)}\}$ `in Step 2 converges,`
	$\gamma^{(k)} = \gamma^{(k-1)} + \frac{1}{2}\left|\gamma^{(k-1)} - \gamma^{(k-2)}\right|$;
	`otherwise`
	$\gamma^{(k)} = \gamma^{(k-1)} - \frac{1}{2}\left|\gamma^{(k-1)} - \gamma^{(k-2)}\right|$.

Step 5:	`Repeat Steps` $2-4$ `until the tolerance is reached.`

As a numerical experiment, the critical values are computed by using the modified MPS with the above algorithm for the problem (3.1)–(3.2) defined on different domains of given shape. Furthermore, using the relation (1.9), the critical size of the domains are calculated and compared with the sizes obtained from some other numerical methods.

4. NUMERICAL RESULTS

To demonstrate the effectiveness of the modified MPS, we consider three numerical examples of Poisson-type nonlinear problems with regular and irregular domains. The domains we considered are squares, amoeba-like, and peanut-like domains. Furthermore, critical size of the domains are computed for the quenching problems defined on regular and irregular domains and compared the results with the results obtained from some other computational methods. The parametric equation of the boundary $\partial\Omega$ is defined as follows

$$\partial\Omega = \{(x,y)|x = r(\vartheta)\cos(\vartheta), y = r(\vartheta)\sin(\vartheta), 0 \le \vartheta < 2\pi\},$$

where

$$r(\vartheta) = \mathrm{e}^{\sin\vartheta}\sin^2(2\vartheta) + \mathrm{e}^{\cos\vartheta}\cos^2(2\vartheta) \tag{4.1}$$

is the amoeba-like boundary and

$$r(\vartheta) = \sqrt{\cos(2\vartheta) + \sqrt{1.1 - \sin^2(2\vartheta)}} \tag{4.2}$$

is the peanut-like boundary. The profiles of amoeba-like and peanut-like domains are shown in Figure 1.

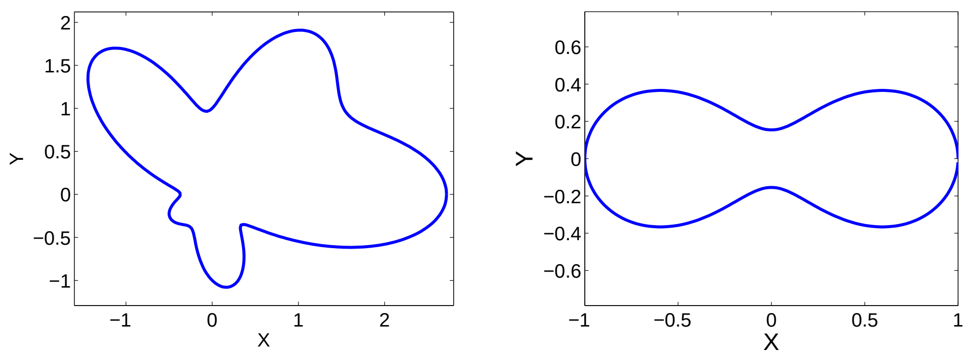

FIGURE 1. The profiles of amoeba-like and peanut-like domains.

The root-mean-squared error ($RMSE$) and the maximum error (MAX) are used to measure the accuracy of the solutions. They are defined as follows

$$RMSE = \sqrt{\frac{1}{n_t}\sum_{j=1}^{n_t}(\hat{u}_j - u_j)^2},$$

and

$$MAX = \max_{1\le j\le n_t} |\hat{u}_j - u_j|,$$

where n_t is the number of test points in the domain and $\hat{u}_j$ and u_j are the approximate solution and exact solution at the j^{th} test point respectively.

Through all the numerical examples except example 4.5 in this section, we have chosen the tolerance $\epsilon = 10^{-7}$ in Algorithm 1 to ensure the accuracy of the solution. We choose polyharmonic splines of order 2 as the radial basis function plus a polynomial basis up to degree 2; i.e., $\{1, x, y, x^2, xy, y^2\}$. If necessary, higher accuracy can be further achieved using high order of polyharmonic splines. We find polyharmonic splines of order 2 is sufficient for solving nonlinear problems in this section. The boundary and interior points are selected uniformly through the domain.

Example 4.1. Let us consider the nonlinear Poisson problem with Dirichlet boundary condition:

$$\Delta u(x,y) = 3u^2, \quad (x,y) \in \Omega, \tag{4.3}$$

$$u(x,y) = g(x,y), \quad (x,y) \in \partial\Omega, \tag{4.4}$$

where $g(x,y)$ is given based on the following analytical solution

$$u_{\text{exact}}(x,y) = \frac{4}{(3+x+y)^2}.$$

In Table 1, we show the numerical results for three different domains. From the table, we observe that the accuracy is quite high. The algorithm is also very efficient since only a small number of iterations is required for all these three domains. The reasons that the unit square domain has high accuracy could be due to the domain is more regular and the area of the domain is smaller than the other two domains.

TABLE 1. The RMSE and Maximum error with different domains.

Domain	(n_i, n_b)	# Iterations	RMSE	MAX
Unit Square	$(841, 236)$	6	$4.378e-08$	$1.233e-07$
Amoeba	$(861, 300)$	15	$5.220e-07$	$8.406e-06$
Peanut	$(832, 290)$	15	$1.204e-06$	$2.265e-05$

Example 4.2. Let us consider the following nonlinear Poisson type problem:

$$\Delta u(x,y) = u^3 - \frac{5}{2} - (1 - \frac{x^2}{4} - y^2)^3, \quad (x,y) \in \Omega, \tag{4.5}$$

$$u(x,y) = g(x,y), \quad (x,y) \in \partial\Omega, \tag{4.6}$$

where $g(x,y)$ is given based on the following exact solution

$$u_{\text{exact}}(x,y) = 1 - \frac{x^2}{4} - y^2.$$

In this example, we have tested our method to the Poisson-type nonlinear PDE which includes the space variables x and y in the source term. The numerical results are presented with the same domains as in previous examples. In Table 2, we observe that less number of iterations are required to obtain excellent accuracy. Therefore, the method can be a good alternative to solve such problem accurately and efficiently.

Example 4.3. In this example, we consider Poisson-Boltzmann Equation [3], a typical example of Poisson-type nonlinear equations. This equation has been widely applied in many physical problems including bio-molecular processes and electrostatic interactions between colloidal particles. Here, we solve this equation defined

TABLE 2. The RMSE and Maximum error with different domains.

Domain	(n_i, n_b)	# Iterations	RMSE	MAX
Unit Square	$(841, 236)$	7	$2.987e-10$	$6.224e-10$
Amoeba	$(861, 300)$	15	$3.132e-09$	$9.966e-09$
Peanut	$(832, 290)$	8	$1.937e-09$	$3.827e-09$

on a square domain to observe the efficiency and accuracy of the present method. Consider the problem

$$\nabla \cdot (\varepsilon \nabla u) = \kappa \sinh(u) + f, \quad (x, y) \in \Omega \tag{4.7}$$

with boundary condition

$$u(x, y) = g(x, y), \quad (x, y) \in \partial\Omega. \tag{4.8}$$

Here, ε and κ are some known field functions. In this experiment, we let $\varepsilon = 1$ and $\kappa = 1$. The function $g(x, y)$ is given based on the following exact solution, $u_{\text{exact}}(x, y) = x^2 + y^2 + e^x \cos(y)$ and the source term $f = 4 - \sinh(x^2 + y^2 + e^x \cos(y))$. We have generated different sets of interior points and boundary points on a square domain $D = [-1, 1] \times [-1, 1]$. In Table 3, various number of interior and boundary points are chosen and excellent results have been observed. The improvement of the accuracy is consistent with the increasing number of interior and boundary points. The number of iterations for various number of interior and boundary points in Table 3 is 26.

TABLE 3. The RMSE and Maximum error with different set of collocation points.

(n_i, n_b)	RMSE	MAX
$(361, 88)$	$3.516e-05$	$1.049e-04$
$(576, 116)$	$1.526e-05$	$4.830e-05$
$(1089, 316)$	$4.778e-06$	$1.646e-05$
$(1444, 476)$	$2.831e-06$	$9.953e-06$
$(3364, 596)$	$5.766e-07$	$2.084e-06$

Example 4.4. In this example, we extend the MPS for solving nonlinear problems to compute the critical domains of quenching problems using Algorithm 2. Let us consider the rectangular domain $D = [0, a] \times [0, b]$. The ratio a/b determines the shape of the rectangle. Using the relation (1.9), we have,

$$\text{Critical size of the domain} = \gamma^* \cdot (\text{Area of } D). \tag{4.9}$$

In Table 4, we show the results of the critical size of the rectangular domains with different ratios a/b and also compared our results with Chan and Ke [5], who had adopted the finite difference method (FDM). We have taken 841 interior points and 236 boundary points. The results of both approaches are very close to each other. Since the computational domain is rectangular, the FDM can be easily applied. However, for irregular domain, the FDM will have difficulty in implementing the solution algorithm. On the other hand, the numerical procedure for the proposed MPS for irregular domain is the same as the rectangular domain. This is one of the attractive features of the MPS for solving nonlinear or quenching problems.

TABLE 4. Critical size of domains for rectangles with different ratio a/b.

Ratio a/b	FDM	MPS
0.125	18.80540	18.81603
0.250	9.67221	9.67679
0.375	6.85011	6.85462
0.500	5.59863	5.60066
0.625	4.96792	4.97582
0.750	4.64531	4.64751
0.875	4.49641	4.50236
1.000	4.45375	4.46474

Example 4.5. In this example, the critical domain sizes of quenching problems are produced for three different domain shapes including regular and irregular geometries and compared with Chan [4] and Chan and Ke [5], Tian [16]. Chan [4] devised a computational method using Green's function whereas Chan and Ke [5] used the finite difference method and Tian [16] applied a numerical method using Delta-shaped basis function to compute the sizes of the critical domains.

In this numerical experiment, we have considered three domains, namely rectangle, ellipse and peanut. The rectangular domain is defined as $[0, 0.5] \times [0, 1]$ whereas the elliptic domain is defined as

$$D = \left\{ (x, y) : \frac{x^2}{a^2} + \frac{y^2}{b^2} < 1 \right\} \tag{4.10}$$

where $a = 0.4575$ and $b = 0.3$. The peanut-like domain we use in this example is different from the one we considered in the examples above. The domain is defined in polar form as

$$r(\vartheta) = \frac{1}{4}(1 + \cos^2(\vartheta)), \quad 0 \le \vartheta \le 2\pi.$$

To make the comparison, we choose the domain exactly the same as taken by previous authors in [4, 5, 16]. The profile of peanut-like domain in this example is shown in

Figure 2. In this example, we have chosen the tolerance $\epsilon = 10^{-9}$ in Algorithm 1. We have chosen 729 interior points and 208 boundary points of the rectangular domain and 858 interior points and 300 boundary points for the elliptical domain. Similarly, the interior points and boundary points taken for the peanut-like domain are 780 and 300 respectively. Furthermore, the area of the rectangle, ellipse and peanut-like domains are 0.5, 0.4312, and 0.4663 respectively. As shown in Table 5, we notice that the critical size of three domains using the modified MPS is close to the results obtained by other methods. The computer running time for rectangle, ellipse, and peanut-like domains are 4.44, 6.59, and 6.61 seconds respectively. There is no report of the numerical efficiency from previous work.

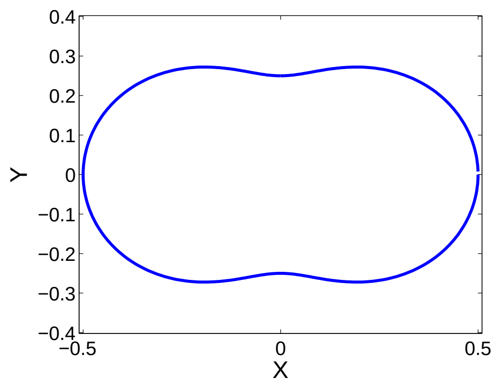

Figure 2. The profile of peanut-like domain.

Table 5. Critical size of domains for different geometries.

Domains	Delta-shaped function	MPS	FDM	Green's function
Rectangle	5.599	5.601	5.599	–
Ellipse	4.460	4.463	–	4.460
Peanut	5.052	5.053	–	–

5. CONCLUSIONS

The MPS has been modified and applied for solving nonlinear Poisson-type problems. The numerical results indicates that the method can be an attractive alternative to other traditional methods. Furthermore, the method is further used to compute the critical domains of the quenching problems and compared the results to some other established numerical methods such as finite difference method. The comparisons show that the MPS can effectively solve nonlinear singular problems. Unlike the FDM, it can solve such problems more efficiently defined on irregular domains with higher dimensions. The simplicity of the implementation is another attraction of this method, apart from the numerical accuracy and the efficiency. The proposed method can easily be extended to higher dimensional nonlinear problems with different boundary conditions defined on different geometries which is a grate challenge for the traditional mesh based methods.

APPENDIX

Lemma 1. Let A, B and C be three matrices of same size and they are in block matrix form with 3 row partitions and 3 column partitions such that the sub matrices on the second row of the matrix A are equal to the corresponding sub matrices of B. Let

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, B = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ A_{21} & A_{22} & A_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix}$$

$$\text{and} \qquad C = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}.$$

Then the sub matrices on the second row of the product matrix AC are equal to the corresponding sub matrices of the product matrix BC.

REFERENCES

[1] A. Acker and B. Kawohl, Remarks on quenching, Nonlinear Anal, 13 (1989), 53–61.

[2] T. Belytschko, Y. Krongauz, D. J. Organ, M. Fleming, and P. Krysl, Meshless methods: an overview and recent developments. Computer Methods in Applied Mechanics and Engineering, special issue, 139:3–47, 1996.

[3] G. C. Bourantas and V. N. Burganos, An implicit meshless scheme for the solution of transient non-linear poisson-type equations. Eng Anal Bound Elem, 37:1117–1126, 2013.

[4] C. Y. Chan, Computation of the critical domain for quenching in an elliptic plate, Neural Parallel Sci. Comput. 1 (1993), 153-162.

[5] C. Y. Chan and L. Ke, Parabolic quenching for nonsmooth convex domains, J. Math. Anal. Appl. 116 (1994), 52–65.

[6] C. S. Chen, C. M. Fan, and P. H. Wen, The method of particular solutions for solving certain partial differential equations. Numerical Methods for Partial Differential Equations, 28:506–522, 2012.

[7] C. S. Chen, The method of fundamental solutions for non-linear thermal explosions. Comm. Numer. Methods Engrg., 11(8):675–681, 1995.

[8] C. S. Chen, C. M. Fan, and P. H. Wen, The method of particular solutions for solving elliptic problems with variable coefficients. The International Journal of Computational Methods, 8:545–559, 2011.

[9] K. Deng and H. A. Lavine, On the blow up of u_t at quenching, Proc. Amer. Math. Soc. 106 (1989), 1049–1056.

[10] G. E. Fasshauer, Meshfree Approximation Methods with MATLAB. Interdisciplinary Mathematical Science, World Scientifc Publishing Company, 2007.

[11] E.J . Kansa, Multiquadrics - a scattered data approximation scheme with applications to computational fluid dynamics - II. Comput. Math. Applic., 19(8/9):147–161, 1990.

[12] J. J. Kasab, S. R. Karur , P. A. Ramachandran, Quasi linear boundary element method for nonlinear Poisson type problems. Eng Anal Bound Elem 1995; 15:277–82.

[13] H. Kawarada, On solutions of initial-boundary problem for $Ut = Uxt + 1/(1 - u)$, Publ. Res. Inst. Math. Sci. 10 (1975), 729–736.

[14] Z. Li, C. V. Pao, Z. A. Qiao, Finite difference method and analysis for 2D nonlinear Poisson-Boltzmann equations. J Sci Comput 2007; 30:61–81.

[15] Y. P. Lin, T. Zhang, Finite element methods for nonlinear Sobolev equations with nonlinear boundary conditions. J Math Anal Appl 1992; 165:180–191.

[16] H. Y. Tian, Computations of Critical Domains for Quenching Problems by Delta-Shaped Basis Functions. Neural, Parallel and Scientific Computations 15 (2007) 501–514.

[17] P. H. Wen and C. S. Chen, The method of particular solutions for solving scalar wave equations. The International Journal for Numerical Methods in Biomedical Engineering, 26:1878–1889, 2010.

[18] Guangming Yao, A Modified Method of Approximate Particular Solutions for Solving Partial Differential Equations. Submitted