

RESEARCH

Open Access



# An efficient meshless radial point collocation method for nonlinear $p$ -Laplacian equation

Samaneh Soradi-Zeid<sup>1</sup>, Mehdi Mesrizadeh<sup>2</sup> and Thabet Abdeljawad<sup>3,4,5\*</sup> 

\*Correspondence:

[tabdeljawad@psu.edu.sa](mailto:tabdeljawad@psu.edu.sa)

<sup>3</sup>Department of Mathematics and General Sciences, Prince Sultan University, Riyadh, Saudi Arabia

<sup>4</sup>Department of Medical Research, China Medical University, Taichung 40402, Taiwan

Full list of author information is available at the end of the article

## Abstract

This paper considered the spectral meshless radial point interpolation (SMRPI) method to unravel for the nonlinear  $p$ -Laplacian equation with mixed Dirichlet and Neumann boundary conditions. Through this assessment, which includes meshless methods and collocation techniques based on radial point interpolation, we construct the shape functions, with the Kronecker delta function property, as basis functions in the framework of spectral collocation methods. Studies in this regard require one to evaluate the high-order derivatives without any kind of integration locally over the small quadrature domains. Finally, some examples are given to illustrate the low computing costs and high enough accuracy and efficiency of this method to solve a  $p$ -Laplacian equation and it would be of great help to fulfill the implementation related to the element-free Galerkin (EFG) method. Both the SMRPI and the EFG methods have been compared by similar numerical examples to show their application in strongly nonlinear problems.

**Keywords:** Collocation method; Meshless method;  $p$ -Laplacian equation; Radial point interpolation technique; Dirichlet and Neumann boundary conditions

## 1 Introduction

We are concerned for the numerical solution of the following nonlinear  $p$ -Laplacian equation:

$$-\operatorname{div}(|\nabla u(\mathbf{x})|^{p-2}\nabla u(\mathbf{x})) = f(\mathbf{x}), \quad \mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \in \Omega, \quad (1)$$

with the Dirichlet and Neumann boundary conditions

$$u(\mathbf{x}) = g_1(\mathbf{x}), \quad \mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \in \Gamma_1, \quad (2)$$

$$n \cdot \nabla u(\mathbf{x}) = g_2(\mathbf{x}), \quad \mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \in \Gamma_2, \quad (3)$$

respectively. Furthermore,  $f$ ,  $g_1$  and  $g_2$  are given functions,  $p \in (1, \infty) \setminus \{2\}$  is a given parameter,  $\operatorname{div}$  and  $\nabla$  are the gradient and divergence operator. Also,  $\Omega$  is a two-dimensional bounded domain with boundary  $\Gamma = \Gamma_1 \cup \Gamma_2$ ,  $n = (n_1, n_2)^T$  is the unit outward normal to

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

$\Gamma$  and  $u$  is the unknown function with the following scalar product in  $\mathbb{R}^2$ :

$$|\nabla u(\mathbf{x})| = \sqrt{\nabla u(\mathbf{x}) \cdot \nabla u(\mathbf{x})}.$$

When  $p = 2$ , Eq. (1) degenerates to the linear Poisson equation. This issue has been used through modeling many nonlinear physical applications evidently. More specifically,  $p$ -Laplacian problem used to illustrate the mechanics and dynamic systems [1, 3, 5, 9, 11, 12]. Nonetheless it should be noted that, due to the high complexity of strongly nonlinear terms in partial differential equations (PDEs), the analytically handling of equations described by this nonlinearity is utterly difficult or even impossible. To prevail through this challenge, practical numerical/approximate methods have been presented to solve them [10, 15, 19]. The authors in [20], utilized the variational method and limiting approach to solving the minimization problems of the Dirichlet/Neumann eigenvalues of the one-dimensional  $p$ -Laplacian equation in which the existence of positive solutions of this problem had already been interrogated by Jankowski [13]. Liu and Yan [18] have used the quasi-norm techniques in order to perform an improved a posteriori error analysis for the  $p$ -Laplacian with only Dirichlet condition. Also, a nonlocal  $p$ -Laplacian-type diffusion equation with Dirichlet boundary condition has been investigated in [2]. An investigation on the Fucik spectrum of the negative  $p$ -Laplacian with different boundary conditions has been proposed in [22]. Other methods used to solve different types of this problems include the finite volume method [14], the finite element method [6], and the finite difference method [24]. Many meshless methods, as a special class of spectral methods, have been used for solving the  $p$ -Laplacian equation such as the Petrov–Galerkin method [21], the radial basis functions (RBFs)-based [4] and the WEB-spline-based mesh-free method [7]. On the contrary of easy implementation of these methods, the challenge of these methods is their low accuracy and high cost of computing in discretization. With an overview you will notice that in these references, the authors have tried to use the interpolate approximate basis functions to reformulated the  $p$ -Laplacian problem as an equivalent minimization problems. Since it is difficult to achieve satisfactory performance in these methods, the EFG method is expanded by Li and Dong in [16] for solving the problem (1)–(3) directly. Because of the high efficiency of nonlinear term in the  $p$ -Laplacian equation, the EFG discretization results in highly nonlinear algebraic systems. To overcome this challenge, the researcher used a linearized iterative process that vividly cause the much computational error in numerical results. Through the next year, they developed the EFG method and used the IIEFG method [17] for this problem and show that the IIEFG method can have a higher computational efficiency than the previous method.

The aforementioned properties of meshless methods have encouraged some researchers to develop new computing architectures and techniques where the primary focus was on hardware simplicity. In spite of great benefits in using the meshless weak form methods, the challenge of these methods is their limited accuracy, locality, complexity and high cost of computing in discretization. Unfortunately, because linear polynomials or linear elements have been used in all these methods, the use of a higher interpolation shows more complex dynamic characteristics in them. Furthermore, the complicated nature of the non-polynomial shape functions, which play an important role in the accuracy of these methods, is computationally expensive in the numerical integration scheme. To overcome these challenges, we invoke a new spectral collocation method (also called a

pseudo-spectral method) to expand a numerical solution of the  $p$ -Laplacian equations (1)–(3). The SMRPI method, comprised of meshless radial point interpolation and collocation techniques, has been assigned and applied to the 2-D and 3-D diffusion equations by Shivanian in [25–28]. Through this technique, the point interpolation method with the help of RBFs is proposed to construct shape functions which have the Kronecker delta function property and are used as basis functions in the framework of the SMRPI. The given method either utilizes any global basis functions for interpolating technique or uses arbitrary points for discretization, giving us a very flexible chart for solving the  $p$ -Laplacian equation. Using SMRPI as a meshless collocation method bears some advantages such as a simple evaluation of high-order derivatives of a given differential equation and being less expensive of computational costs. In addition, to implement this method we are not required to use any local integration in small quadrature domains, shape parameterization or refinement techniques as is essential in Galerkin weak form meshless methods such as the EFG method.

The overall organization of this paper is as follows. The next section presents SMRPI method briefly. Understanding functions in terms of the SMRPI approach can be a major challenge in this study. Then the third section shows our methodology. In this section, Eqs. (1)–(3) could be performed by the SMRPI method. The numerical results and comparison have been set in Sect. 4 and the conclusions as regards the results in the last section.

## 2 SMRPI scheme

To perform the continuous function  $u(\mathbf{x})$  of  $p$ -Laplacian equations (1)–(3) via the SMRPI method, we have to outline it in this section. The continuous function  $u(\mathbf{x})$  can be represented via RBF  $\mathbf{R}_i(\mathbf{x})$  and monomials in the point of interest  $\mathbf{x} \in \Omega_l$ . So, the coefficients  $a_i$  and  $b_j$  can be considered as

$$u(\mathbf{x}) = \sum_{i=1}^n R_i(\mathbf{x})a_i + \sum_{j=1}^m P_j(\mathbf{x})b_j = \mathbf{R}^T(\mathbf{x})\mathbf{a} + \mathbf{P}^T(\mathbf{x})\mathbf{b}, \quad (4)$$

in which  $\Omega_l$  is a disk centered at  $x_l$  with radius  $r_s$ ,  $n$  is the number of point in  $\Omega_l$  and  $m$  is the number of polynomial basis functions. When  $m = 0$ , only RBFs are used, otherwise, the RBF is augmented with  $m$  polynomial basis functions. In the point of interest  $x_l$ , we enforce Eq. (4) to be satisfied at those  $n$  nodes surrounding it. Then the linear algebraic system of equations (4) is represented as follows:

$$\mathbf{U}_s = \mathbf{R}_n^T \mathbf{a} + \mathbf{P}_m^T \mathbf{b}, \quad (5)$$

in which  $\mathbf{U}_s$  is the vector of function values defined as

$$\mathbf{U}_s = \{u_1, u_2, u_3, \dots, u_n\}^T, \quad (6)$$

$\mathbf{R}_n$  denotes the RBFs moment matrix as follows:

$$\mathbf{R}_n = \begin{pmatrix} \mathbf{R}_{1,1} & \mathbf{R}_{1,2} & \cdots & \mathbf{R}_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{n,1} & \mathbf{R}_{n,2} & \cdots & \mathbf{R}_{n,n} \end{pmatrix}, \quad (7)$$

and  $\mathbf{P}_m$  represents the polynomial moment matrix defined by

$$\mathbf{P}_m = \begin{pmatrix} p_1(\mathbf{x}_1) & p_1(\mathbf{x}_2) & \cdots & p_1(\mathbf{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ p_m(\mathbf{x}_1) & p_m(\mathbf{x}_2) & \cdots & p_m(\mathbf{x}_N) \end{pmatrix}. \quad (8)$$

Also, the vector of unknown coefficients for RBFs is

$$\mathbf{a} = \{a_1, a_2, a_3, \dots, a_n\}^T, \quad (9)$$

and the vector of unknown coefficients for the basis polynomial is

$$\mathbf{b} = \{b_1, b_2, b_3, \dots, b_m\}^T. \quad (10)$$

Assume that  $r_k$ ,  $k = 1, 2, \dots, n$ , being the distance between nodes in the support domain,  $\mathbf{R}_{k,i} = \mathbf{R}_i(r_k)$  are the RBFs in (7). We added the following  $m$  equations in (5) to make a square matrix:

$$\mathbf{P}_m^T \mathbf{a} = 0. \quad (11)$$

So, the following system of equations is obtained from (5) and (11):

$$\hat{\mathbf{U}}_s = \begin{pmatrix} \mathbf{U}_s \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{R}_n & \mathbf{P}_m \\ \mathbf{P}_m^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} = \mathbf{G} \hat{\mathbf{a}}_s, \quad (12)$$

in which the matrix  $\mathbf{G}$  is theoretically non-singular [29] and

$$\hat{\mathbf{a}}_s = \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix}. \quad (13)$$

Now from (12) we have

$$\hat{\mathbf{a}}_s = \mathbf{G}^{-1} \hat{\mathbf{U}}_s. \quad (14)$$

By rewriting Eq. (4) we obtain

$$u(\mathbf{x}) = \hat{\Phi}(\mathbf{x}) \hat{\mathbf{U}}_s, \quad (15)$$

such that

$$\hat{\Phi}(\mathbf{x}) = [\mathbf{R}_n(\mathbf{x}), \mathbf{P}_m(\mathbf{x})] \mathbf{G}^{-1}. \quad (16)$$

The shape functions corresponding to the nodal displacements of radial point interpolation method (RPIM), are the first  $n$  functions of the above vector and we show them by the vector  $\Phi^T(\mathbf{x})$ :

$$\Phi(\mathbf{x}) = \{\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_n(\mathbf{x})\}. \quad (17)$$

Now Eq. (15) converts to the following form:

$$u(\mathbf{x}) = \Phi(\mathbf{x})\mathbf{U}_s = \sum_{i=1}^n \phi_i(\mathbf{x})u_i. \quad (18)$$

Also it is well known that the Kronecker delta function property is attached to the RPIM shape functions by (16), which is explicitly written thus:

$$\phi_i(\mathbf{x}_j) = \begin{cases} 1, & i = j, j = 1, 2, \dots, n, \\ 0, & i \neq j, j = 1, 2, \dots, n, \end{cases} \quad (19)$$

and lead to a sparse global collocation system. We assume that the total number of nodes that cover the  $\bar{\Omega} = \Omega \cap \partial\Omega$  is  $N$ . By rewriting Eq. (18), we have

$$u(\mathbf{x}) = \Phi(\mathbf{x})\mathbf{U}_s = \sum_{i=1}^N \phi_i(\mathbf{x})u_i. \quad (20)$$

Since corresponding to node  $x_j$  there is a shape function  $\phi_i(\mathbf{x})$ ,  $i = 1, 2, \dots, N$ , obviously we have from Eq. (19)

$$\forall x_j \in \Omega_x^c, \quad \phi_i(x_j) = 0, \quad (21)$$

where  $\Omega_x^c = \{x_j : x_j \notin \Omega_x\}$ . Now the derivatives of  $u(\mathbf{x})$  respect to  $x_i$ ,  $i$ th component of  $\mathbf{x} = \{x_1, \dots, x_i, \dots, x_N\}$ , determined as

$$\frac{\partial u}{\partial x_i}(\mathbf{x}) = \sum_{j=1}^N \frac{\partial \phi_j}{\partial x_i}(\mathbf{x})u_j, \quad (22)$$

and for high derivatives of  $u(\mathbf{x})$  we have

$$\frac{\partial^s u}{\partial (x_i)^s}(\mathbf{x}) = \sum_{j=1}^N \frac{\partial^s \phi_j}{\partial (x_i)^s}(\mathbf{x})u_j, \quad (23)$$

where  $\frac{\partial^s}{\partial (x_i)^s}$  is for the  $s$ th derivatives with respect to  $x_i$  implying that due to Eq. (21),  $\forall x_j \in \Omega_x^c$ ,  $\frac{\partial^s \phi_j}{\partial (x_i)^s}(\mathbf{x}) = 0$ ,  $s = 1, 2, \dots$ . Denoting  $u_{x_i}^{(s)}(\cdot) = \frac{\partial^s u(\cdot)}{\partial (x_i)^s}$  and setting  $x = x_i$  in Eq. (20). Then the following matrix form is given:

$$\begin{pmatrix} u_{x_i}^{(s)}(\mathbf{x}_1) \\ \vdots \\ u_{x_i}^{(s)}(\mathbf{x}_N) \end{pmatrix} = \overbrace{\begin{pmatrix} \frac{\partial^s \phi_1}{\partial (x_i)^s}(\mathbf{x}_1) & \cdots & \frac{\partial^s \phi_N}{\partial (x_i)^s}(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \frac{\partial^s \phi_1}{\partial (x_i)^s}(\mathbf{x}_N) & \cdots & \frac{\partial^s \phi_N}{\partial (x_i)^s}(\mathbf{x}_N) \end{pmatrix}}^{D_{x_i}^s} \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix}. \quad (24)$$

This matrix-vector form for high-order derivatives is as follows:

$$U_{x_i}^s = D_{x_i}^s U, \quad (25)$$

where

$$U_{x_i}^s = \{u_{x_i}^{(s)}(\mathbf{x}_1), \dots, u_{x_i}^{(s)}(\mathbf{x}_N)\}. \quad (26)$$

### 3 Implementation of the SMRPI for $p$ -Laplacian equation

By using (20) and (23), the following form is obtained for Eq. (1):

$$-\operatorname{div}\left(\left|\sum_{j=1}^N \nabla \phi_j(\mathbf{x}) u_j\right|^{p-2} \sum_{j=1}^N \nabla \phi_j(\mathbf{x}) u_j\right) = f(\mathbf{x}). \quad (27)$$

For simplicity of computation, we replace the term  $|\sum_{j=1}^N \nabla \phi_j(\mathbf{x}) u_j|^{p-2}$  by a linear combination of the basis functions  $\phi_j(\mathbf{x})$ ,

$$\left|\sum_{j=1}^N \nabla \phi_j(\mathbf{x}) u_j\right|^{p-2} = \sum_{j=1}^N \phi_j(\mathbf{x}) \kappa_j(\mathbf{u}), \quad (28)$$

where  $\mathbf{u} = \{u_1, \dots, u_N\}^T$ . Let  $x = x_i$ ,  $i = 1, 2, \dots, N_\Omega$  ( $N_\Omega$  is the number of nodes in  $\Omega$ ), so with respect to (19) we have

$$\left|\sum_{j=1}^N \nabla \phi_j(\mathbf{x}_i) u_j\right|^{p-2} = \kappa_j(\mathbf{u}), \quad i = 1, 2, \dots, N_\Omega. \quad (29)$$

Applying Eq. (28) to Eq. (27) implies

$$-\operatorname{div}\left(\sum_{j=1}^N \phi_j(\mathbf{x}) \kappa_j(\mathbf{u}) \sum_{j=1}^N \nabla \phi_j(\mathbf{x}_i) u_j\right) = f(\mathbf{x}_i), \quad i = 1, 2, \dots, N_\Omega, \quad (30)$$

and equivalently

$$\sum_{k=1}^N -\operatorname{div}\left(\sum_{j=1}^N \phi_j(\mathbf{x}_i) \nabla \phi_k(\mathbf{x}_i) \kappa_j(\mathbf{u})\right) u_k = f(\mathbf{x}_i), \quad i = 1, 2, \dots, N_\Omega. \quad (31)$$

By using Eqs. (2) and (3) for the nodes which are located, respectively, on the boundary  $\Gamma_1$  and  $\Gamma_2$ , we have the following equivalent relations:

$$\sum_{j=1}^N \phi_j(\mathbf{x}_i) u_j = g_1(\mathbf{x}_i), \quad i = 1, 2, \dots, N_{\Gamma_1}, \quad (32)$$

$$\sum_{j=1}^N \nabla \phi_j(\mathbf{x}_i) \cdot \mathbf{n} u_j = g_2(\mathbf{x}_i), \quad i = 1, 2, \dots, N_{\Gamma_2}. \quad (33)$$

According to the above discussions, problem (1)–(3) is equivalent to the following system of algebraic equations:

$$\mathbf{K}(\mathbf{u})\mathbf{u} = \mathbf{F}, \quad (34)$$

where

$$[\mathbf{K}(\mathbf{u})]_{i,j} = -\operatorname{div} \left( \sum_{j=1}^N \phi_j(\mathbf{x}_i) \nabla \phi_k(\mathbf{x}_i) \kappa_j(\mathbf{u}) \right), \quad (35)$$

$$\mathbf{F}_i = f(\mathbf{x}_i), \quad i = 1, \dots, N_\Omega, j = 1, \dots, N,$$

$$[\mathbf{K}(\mathbf{u})]_{i,j} = \phi_j(\mathbf{x}_i), \quad (36)$$

$$\mathbf{F}_i = g_1(\mathbf{x}_i), \quad i = N_\Omega + 1, \dots, N_\Omega + N_{\Gamma_1}, j = 1, \dots, N,$$

and also

$$[\mathbf{K}(\mathbf{u})]_{i,j} = \nabla \phi_j(\mathbf{x}_i) \cdot \mathbf{n}, \quad (37)$$

$$\mathbf{F}_i = g_2(\mathbf{x}_i), \quad i = N_\Omega + N_{\Gamma_1} + 1, \dots, N, j = 1, \dots, N.$$

An iterative method with an initial guess  $\mathbf{u}_0$  is applied to obtain an approximate solution of (34). For comparable results between our results and other literature, we can solve the nonlinear system (34) by invoking the Matlab function *fsolve*.

#### 4 Numerical results

In this section, four numerical examples are solved to demonstrate the efficiency and accuracy of the present SMRPI method. These examples were solved by other methods such as EFG method [16], IEFM method [17], FVM [14], the DG method [8] and the MFEM [7, 23]. Since the best results of these methods are obtained using the EFG method, the comparison in this paper has been done between our results with those obtained in [16] and we omit any other attempts. Also, we need to recall the following Sobolev norms to analyze the error of our method:

$$|u - u_h|_{0,p} = \left[ \int_{\Omega} |u - u_h|^p d\Omega \right]^{\frac{1}{p}}, \quad (38)$$

$$|u - u_h|_{1,p} = \left[ \int_{\Omega} \left| \frac{\partial}{\partial x_1} (u - u_h) \right|^p + \left| \frac{\partial}{\partial x_2} (u - u_h) \right|^p d\Omega \right]^{\frac{1}{p}}, \quad (39)$$

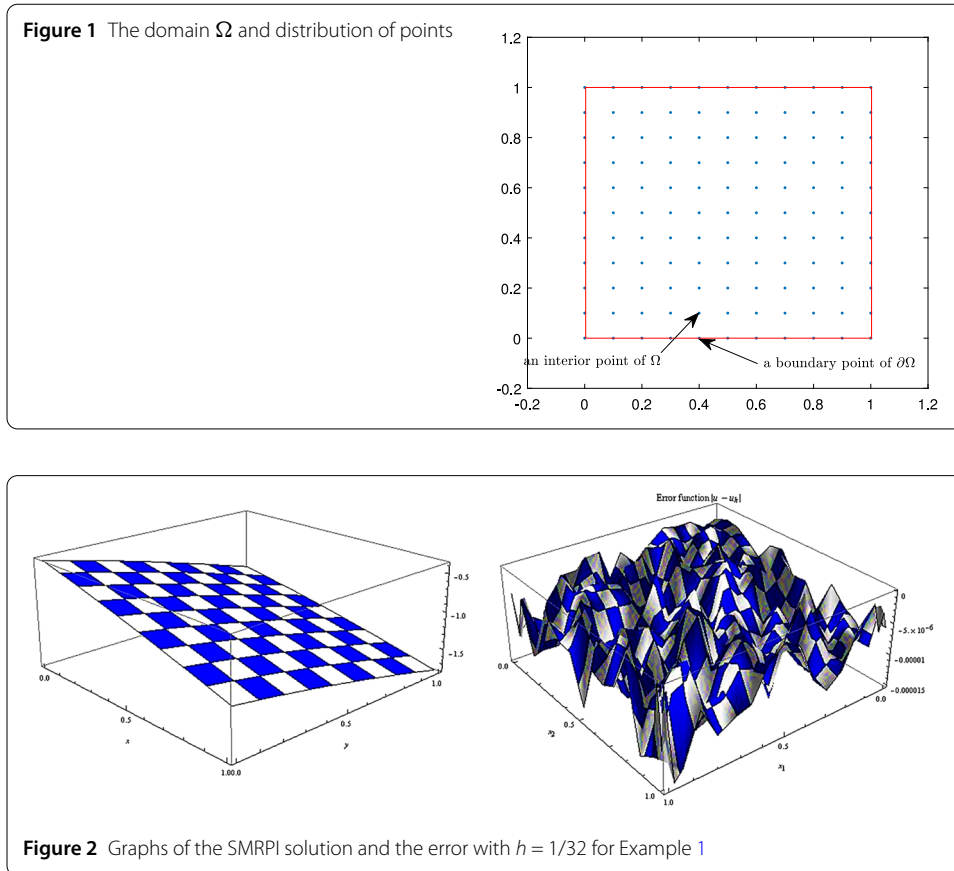
$$\|\sigma - \sigma_h\|_{0,q} = \left[ \int_{\Omega} |\sigma - \sigma_h|^q d\Omega \right]^{\frac{1}{q}}, \quad (40)$$

and quasi-norms as

$$|u - u_h|_{(u,p)} = \left[ \int_{\Omega} (|\nabla(u - u_h)| + |\nabla(u - u_h)|)^{p-2} |\nabla(u - u_h)|^2 d\Omega \right]^{\frac{1}{2}}, \quad (41)$$

$$|\sigma - \sigma_h|_{(\sigma,q)} = \left[ \int_{\Omega} (|\nabla(\sigma - \sigma_h)| + |\nabla(\sigma - \sigma_h)|)^{q-2} |\nabla(\sigma - \sigma_h)|^2 d\Omega \right]^{\frac{1}{2}}, \quad (42)$$

where  $\sigma = |\nabla u|^{p-2} \nabla u$  is the flux associated to  $u$  and  $q = \frac{p}{p-1}$  denotes the dual of  $p$ . Obviously, the norms in Eqs. (39)–(42) are equivalent when  $p = 2$ . To apply the SMRPI method on all examples, let  $r_s = 4.5h$ , which is significant enough to have a sufficient number of



nodes for any support domain in which the radius of disks of the support domain is used to construct the basis functions. The location of central nodes in the region of the test problem is shown in Fig. 1. In Eq. (4) we use the thin plate spline (TPS) for RBF with three order  $\phi(\mathbf{x}) = \|\mathbf{x}\|^6 \log \|\mathbf{x}\|$  where  $\|\cdot\|$  is the Euclidean norm in  $\mathbb{R}^2$  and  $\mathbf{x} = (x_1, x_2)$ . Also, let  $m = 21$ , leading to the polynomial basis functions

$$P_m^T(\mathbf{x}) = \{1, x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3, x_1^4, x_1^3x_2, x_1^2x_2^2, x_1x_2^3, x_2^4, x_1^5, x_1^4x_2, x_1^3x_2^2, x_1^2x_2^3, x_1x_2^4, x_2^5\}.$$

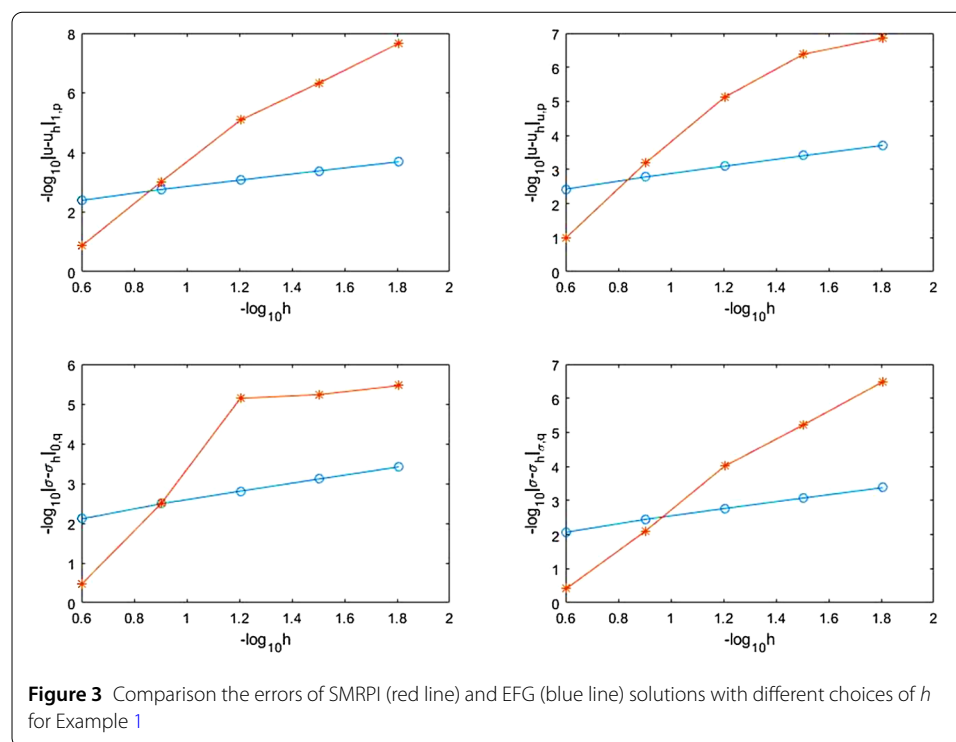
**Example 1** For the first example, we solved the radially symmetric problem in  $\Omega = [0, 1]^2$  with Dirichlet boundary condition. This problem has an analytical solution as follows:

$$u(x_1, x_2) = \frac{p-1}{p-s} \left( \frac{1}{2-s} \right)^{\frac{1}{p-1}} (1 - r^{\frac{p-s}{p-1}}), \quad r = \sqrt{(x_1+1)^2 + (x_2+1)^2},$$

in which  $p$  and  $s$  are two parameters,  $u$  and  $\sigma$  are infinitely smooth and the right-hand function in Eq. (1) is  $f(x_1, x_2) = r^{-s}$ . We take  $p = 4.0$  and  $s = 0.6$  for the solution of this problem. Figure 2 gives the SMRPI solution  $u_h$  and the associated error  $u - u_h$  with  $h = 1/32$ . The error results have been computed with different types of norms and quasi-norms as presented in Table 1. From the results which we can see in this table that the SMRPI method accurately provides approximate solutions. Moreover, the accuracy of the numerical results has been improved by decreasing the number of  $h$ . In addition, for evaluation of the

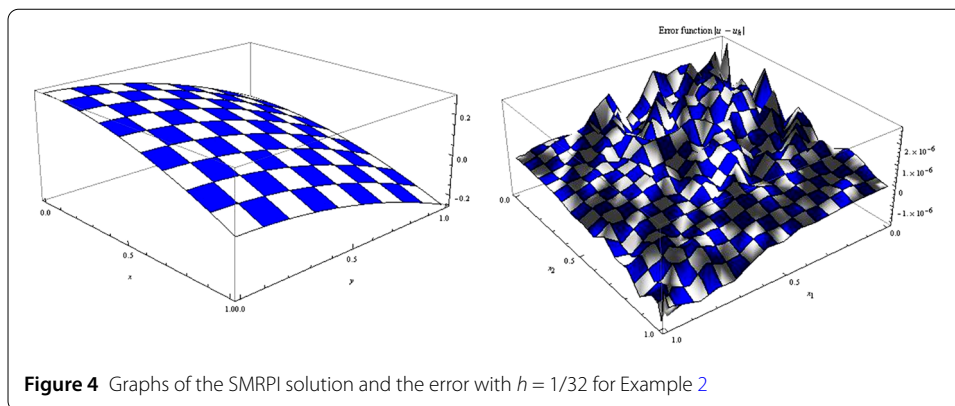
**Table 1** Comparing errors with different types of norms and quasi-norms for Example 1 with  $p = 4.0$  and  $s = 0.6$  for SMRPI and EFG methods

$h$		$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$
$ u - u_h _{1,p}$	EFG	$4.024 \times 10^{-3}$	$1.740 \times 10^{-3}$	$8.403 \times 10^{-4}$	$4.116 \times 10^{-4}$	$2.049 \times 10^{-4}$
	SMRPI	$1.334 \times 10^{-1}$	$9.632 \times 10^{-4}$	$8.286 \times 10^{-6}$	$4.583 \times 10^{-7}$	$2.189 \times 10^{-8}$
$ u - u_h _{u,p}$	EFG	$3.758 \times 10^{-3}$	$1.657 \times 10^{-3}$	$7.986 \times 10^{-4}$	$3.924 \times 10^{-4}$	$1.954 \times 10^{-4}$
	SMRPI	$1.004 \times 10^{-1}$	$6.520 \times 10^{-4}$	$7.601 \times 10^{-6}$	$4.113 \times 10^{-7}$	$1.398 \times 10^{-7}$
$ \sigma - \sigma_h _{0,q}$	EFG	$7.751 \times 10^{-3}$	$3.239 \times 10^{-3}$	$1.559 \times 10^{-3}$	$7.678 \times 10^{-4}$	$3.831 \times 10^{-4}$
	SMRPI	$3.335 \times 10^{-1}$	$3.212 \times 10^{-3}$	$7.256 \times 10^{-6}$	$5.813 \times 10^{-6}$	$3.459 \times 10^{-6}$
$ \sigma - \sigma_h _{\sigma,q}$	EFG	$8.527 \times 10^{-3}$	$3.573 \times 10^{-3}$	$1.715 \times 10^{-3}$	$8.422 \times 10^{-4}$	$4.197 \times 10^{-4}$
	SMRPI	$3.884 \times 10^{-1}$	$7.932 \times 10^{-3}$	$9.886 \times 10^{-5}$	$5.983 \times 10^{-6}$	$3.289 \times 10^{-7}$



accuracy of the obtained results, we compared our results with the corresponding values of EFG in this table. Error comparison results for the EFG method and the SMRPI are given in Fig. 3. Obviously, both methods converge monotonously but our suggested approach is more effective.

**Example 2** For the second case, we take  $p = 1.5$  and  $s = 0.6$  for the solution of Example 1. Here,  $r = \sqrt{x_1^2 + x_2^2}$  and  $u$  and  $\sigma$  have a limited regularity for the performance of the SMRPI method. Figure 4 gives the SMRPI solution  $u_h$  and the associated error  $u - u_h$  with  $h = 1/32$ . Table 2 shows the values of computed errors with different types of norms and quasi-norms obtained by the EFG method and our present method. In addition, this table states the effect of various values of the nodal spacing  $h$  on the results. As seen in this table, the accuracy of the SMRPI method can be easily investigated by comparing the error values improved by our introduced method with those obtained by the other methods.



**Figure 4** Graphs of the SMRPI solution and the error with  $h = 1/32$  for Example 2

**Table 2** Comparing the errors with different types of norms and quasi-norms for Example 2 with  $\rho = 1.5$  and  $s = 0.6$  for SMRPI and EFG methods

$h$		$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$
$ u - u_h _{1,p}$	EFG	$7.135 \times 10^{-3}$	$3.352 \times 10^{-3}$	$1.612 \times 10^{-3}$	$7.856 \times 10^{-4}$	$3.908 \times 10^{-4}$
	SMRPI	$3.321 \times 10^{-1}$	$6.632 \times 10^{-4}$	$8.286 \times 10^{-6}$	$4.151 \times 10^{-6}$	$7.194 \times 10^{-7}$
$ u - u_h _{u,p}$	EFG	$9.129 \times 10^{-3}$	$4.456 \times 10^{-3}$	$2.125 \times 10^{-3}$	$1.010 \times 10^{-4}$	$4.915 \times 10^{-4}$
	SMRPI	$4.225 \times 10^{-1}$	$2.025 \times 10^{-4}$	$7.887 \times 10^{-6}$	$9.301 \times 10^{-7}$	$1.811 \times 10^{-7}$
$ \sigma - \sigma_h _{0,q}$	EFG	$1.094 \times 10^{-2}$	$6.371 \times 10^{-3}$	$3.518 \times 10^{-3}$	$1.825 \times 10^{-4}$	$9.108 \times 10^{-4}$
	SMRPI	$4.521 \times 10^{-1}$	$2.012 \times 10^{-3}$	$8.986 \times 10^{-5}$	$9.103 \times 10^{-6}$	$1.129 \times 10^{-6}$
$ \sigma - \sigma_h _{\sigma,q}$	EFG	$7.023 \times 10^{-3}$	$3.527 \times 10^{-3}$	$1.704 \times 10^{-3}$	$8.150 \times 10^{-4}$	$3.971 \times 10^{-4}$
	SMRPI	$3.884 \times 10^{-1}$	$1.938 \times 10^{-3}$	$6.416 \times 10^{-5}$	$5.553 \times 10^{-6}$	$1.112 \times 10^{-6}$

**Example 3** In the third example, we consider a Dirichlet problem with

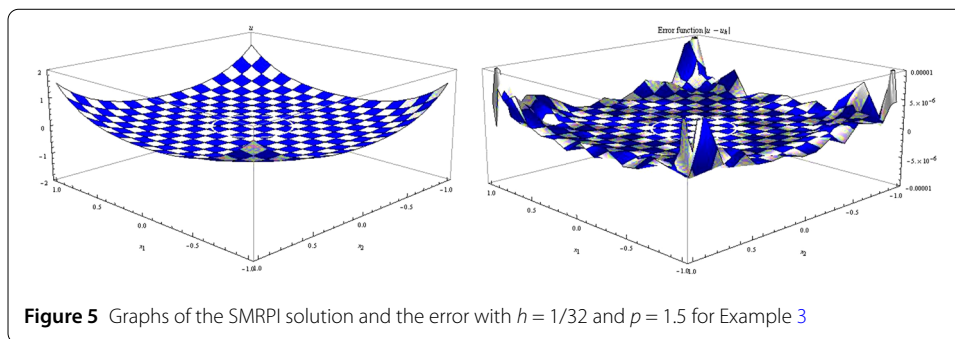
$$f(x, y) = \begin{cases} 0, & r = \sqrt{x_1^2 + x_2^2} < a, \\ 4^{p-1}(r-a)^{3p-4}(2-3p+\frac{a}{r}), & r \geq a, \end{cases}$$

in which  $(x_1, x_2)^T \in \Omega = [0, 1]^2$  and  $a = 0.3$ . The analytical solution of this problem has the following form:

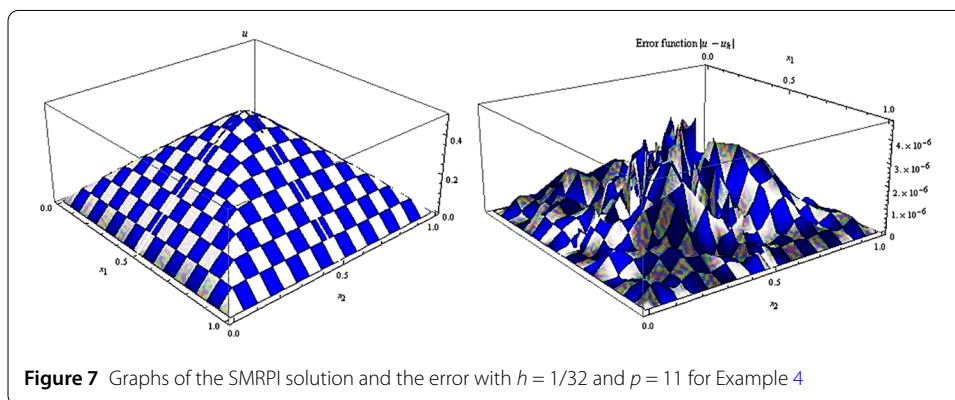
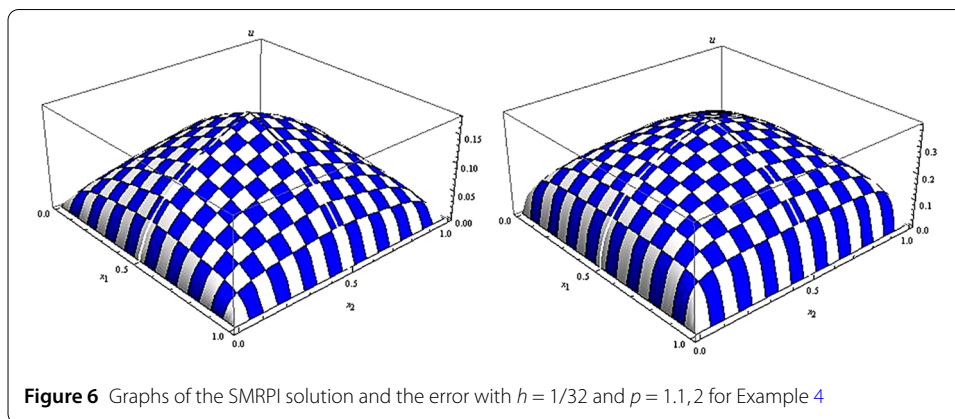
$$u(x_1, x_2) = \begin{cases} 0, & r < a, \\ (r-a)^4, & r \geq a. \end{cases}$$

Figure 5 gives the SMRPI solution  $u_h$  and the associated error  $u - u_h$  with  $h = 1/32$ . Table 3 gives the computed errors for EFG and SMRPI methods with  $p = 1.5, 4.0$ . Comparing these values reveals that the SMRPI method has a good convergence characteristic and the accuracy of the present method is higher than other methods.

**Example 4** In the last example we solved a torsional creep problem by the SMRPI method in which the behavior of this problem was introduced in [16], so we ignore the details. Figure 6 gives the SMRPI solution  $u_h$  with  $h = 1/32$  for  $p = 1.1, 2.0$ , respectively. Figure 7 gives the SMRPI solution  $u_h$  and the associated error  $u - u_h$  with  $h = 1/32$  and  $p = 11$ . The convergence rates of both the SMRPI and the EFG methods are compared for different parameters  $p$  in Table 4. Error comparison results for

**Table 3** Comparing typical errors for Example 3 for SMRPI and EFG methods

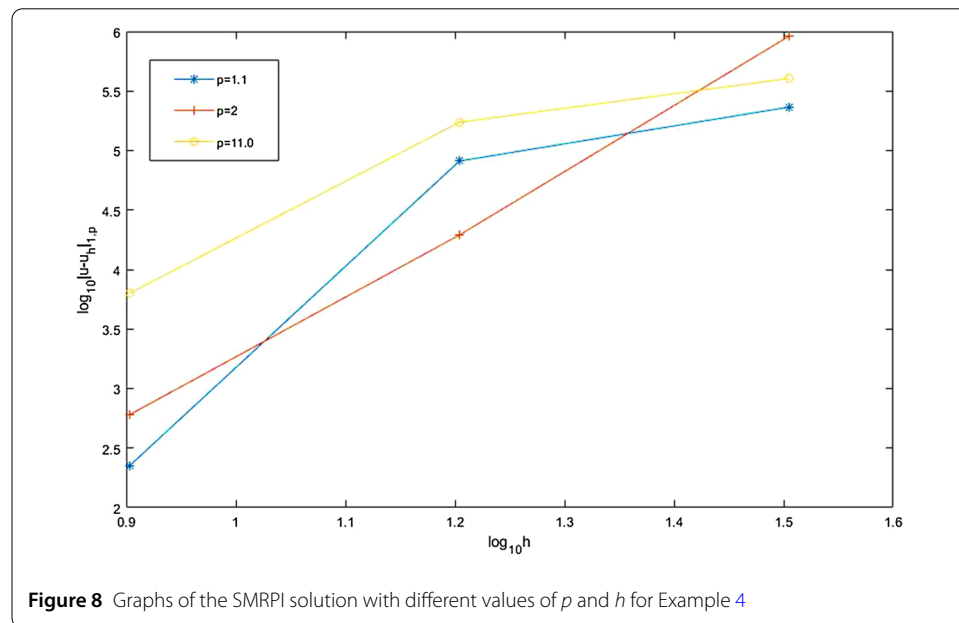
$h$		$\frac{1}{8}$		$\frac{1}{16}$	
		EFG	SMRPI	EFG	SMRPI
$p = 1.5$	$ u - u_h _{0,p}$	$1.582 \times 10^{-3}$	$5.595 \times 10^{-4}$	$2.509 \times 10^{-4}$	$1.411 \times 10^{-6}$
	$ u - u_h _{1,p}$	$5.491 \times 10^{-2}$	$4.427 \times 10^{-3}$	$2.193 \times 10^{-2}$	$5.125 \times 10^{-5}$
	$ u - u_h _{u,p}$	$4.232 \times 10^{-2}$	$1.011 \times 10^{-3}$	$1.674 \times 10^{-2}$	$2.514 \times 10^{-4}$
$p = 4.0$	$ u - u_h _{0,p}$	$5.514 \times 10^{-3}$	$1.144 \times 10^{-5}$	$1.407 \times 10^{-3}$	$5.985 \times 10^{-6}$
	$ u - u_h _{1,p}$	$8.098 \times 10^{-2}$	$1.402 \times 10^{-4}$	$2.656 \times 10^{-2}$	$9.948 \times 10^{-6}$
	$ u - u_h _{u,p}$	$1.151 \times 10^{-1}$	$2.221 \times 10^{-2}$	$4.856 \times 10^{-2}$	$5.552 \times 10^{-4}$



different choices of  $p$  are given in Fig. 8. The numerical results clearly show the improvement of accuracy using the SMRPI method. So, our suggested approach is more effective.

**Table 4** Comparing typical errors for Example 4 for SMRPI and EFG methods

$h$		$\frac{1}{8}$		$\frac{1}{16}$		$\frac{1}{32}$	
		EFG	SMRPI	EFG	SMRPI	EFG	SMRPI
$p = 1.1$	$ u - u_h _{0,p}$	$1.570 \times 10^{-1}$	$1.775 \times 10^{-3}$	$4.557 \times 10^{-3}$	$1.572 \times 10^{-5}$	$2.155 \times 10^{-3}$	$5.141 \times 10^{-6}$
	$ u - u_h _{1,p}$	$2.949 \times 10^{-1}$	$4.448 \times 10^{-3}$	$8.557 \times 10^{-2}$	$1.225 \times 10^{-5}$	$4.047 \times 10^{-3}$	$4.311 \times 10^{-6}$
	$ u - u_h _{u,p}$	$2.237 \times 10^{-2}$	$3.971 \times 10^{-3}$	$1.162 \times 10^{-2}$	$1.588 \times 10^{-4}$	$1.936 \times 10^{-2}$	$1.685 \times 10^{-5}$
$p = 2$	$ u - u_h _{0,p}$	$1.582 \times 10^{-3}$	$5.595 \times 10^{-4}$	$2.509 \times 10^{-4}$	$1.411 \times 10^{-6}$	$2.509 \times 10^{-4}$	$1.411 \times 10^{-6}$
	$ u - u_h _{1,p}$	$3.163 \times 10^{-2}$	$1.658 \times 10^{-3}$	$1.643 \times 10^{-2}$	$5.125 \times 10^{-5}$	$2.738 \times 10^{-3}$	$1.091 \times 10^{-6}$
$p = 11.0$	$ u - u_h _{0,p}$	$3.681 \times 10^{-2}$	$5.849 \times 10^{-3}$	$2.038 \times 10^{-2}$	$5.985 \times 10^{-5}$	$6.523 \times 10^{-3}$	$1.258 \times 10^{-6}$
	$ u - u_h _{1,p}$	$3.921 \times 10^{-2}$	$1.582 \times 10^{-4}$	$2.171 \times 10^{-2}$	$5.781 \times 10^{-6}$	$7.164 \times 10^{-3}$	$2.478 \times 10^{-6}$
	$ u - u_h _{u,p}$	$2.775 \times 10^{-2}$	$8.520 \times 10^{-3}$	$7.711 \times 10^{-3}$	$4.112 \times 10^{-4}$	$3.487 \times 10^{-3}$	$6.632 \times 10^{-5}$

**Figure 8** Graphs of the SMRPI solution with different values of  $p$  and  $h$  for Example 4

## 5 Conclusion

This paper is devoted to solving the  $p$ -Laplacian equation with mixed Dirichlet and Neumann boundary conditions by the SMRPI method. In spite of easy implementation of other methods for solving this problem, the challenge of these methods is their limited accuracy, locality, complexity and high cost of computing in linearization of the nonlinear terms. So, we propose an SMRPI scheme that, in addition, employs radial functions as a means of local interpolation, offers a very flexible framework with a high accuracy for solving  $p$ -laplacian equation. Numerical examples show that our suggested approach is more effective and accurate than the EFG method for various choices of norms and semi norms. We show that the SMRPI method has a good convergence characteristic in comparison to other methods. The interested reader is advised to apply this method for others strongly nonlinear PDEs.

## Acknowledgements

The third author would like to thank Prince Sultan University for funding this work through research group Nonlinear Analysis Methods in Applied Mathematics (NAMAM) group number RG-DES-2017-01-17.

## Funding

Not applicable.

**Availability of data and materials**

Not applicable in this article.

**Competing interests**

The authors declare that they have no competing interests.

**Authors' contributions**

All authors have made equal contributions to the writing of this article. The final manuscript read and approved by all authors.

**Author details**

<sup>1</sup>Faculty of Industry and Mining (khash), University of Sistan and Baluchestan, Zahedan, Iran. <sup>2</sup>Department of Mathematics, Kharazmi University, Karaj, Iran. <sup>3</sup>Department of Mathematics and General Sciences, Prince Sultan University, Riyadh, Saudi Arabia. <sup>4</sup>Department of Medical Research, China Medical University, Taichung 40402, Taiwan. <sup>5</sup>Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan.

**Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 25 May 2020 Accepted: 30 September 2020 Published online: 12 October 2020

**References**

1. Andreu, F., Mazon, J.M., Rossi, J.D., Toledo, J.: A nonlocal  $p$ -Laplacian evolution equation with Neumann boundary conditions. *J. Math. Pures Appl.* **90**(2), 201–227 (2008)
2. Andreu, F., Mazon, J.M., Rossi, J.D., Toledo, J.: A nonlocal  $p$ -Laplacian evolution equation with nonhomogeneous Dirichlet boundary conditions. *SIAM J. Math. Anal.* **40**(5), 1815–1851 (2009)
3. Andreu-Vaillio, F., Caselles, V., Mazon, J.M.: *Parabolic Quasilinear Equations Minimizing Linear Growth Functionals*. Progress in Mathematics, vol. 223, Springer, Basel (2004)
4. Bernal, F.: Trust-region methods for nonlinear elliptic equations with radial basis functions. *Comput. Math. Appl.* **72**(7), 1743–1763 (2016)
5. Bouchitte, G., Buttazzo, G., De Pascale, L.: A  $p$ -Laplacian approximation for some mass optimization problems. *J. Optim. Theory Appl.* **118**(1), 1–25 (2003)
6. Breit, D., Diening, L., Schwarzacher, S.: Finite element approximation of the  $p$ -Laplacian. *SIAM J. Numer. Anal.* **53**(1), 551–572 (2015)
7. Chaudhary, S., Srivastava, V., Kumar, V.S., Srinivasan, B.: WEB-spline-based mesh-free finite element approximation for  $p$ -Laplacian. *Int. J. Comput. Math.* **93**(6), 1022–1043 (2016)
8. Cockburn, B., Shen, J.: A hybridizable discontinuous Galerkin method for the  $p$ -Laplacian. *SIAM J. Sci. Comput.* **38**(1), A545–A566 (2016)
9. Cuccu, F., Emamizadeh, B., Porru, G.: Optimization of the first eigenvalue in problems involving the  $p$ -Laplacian. *Proc. Am. Math. Soc.* **137**(5), 1677–1687 (2009)
10. Deng, S.G.: A local mountain pass theorem and applications to a double perturbed  $p(x)$ -Laplacian equations. *Appl. Math. Comput.* **211**(1), 234–241 (2009)
11. Glowinski, R., Rappaz, J.: Approximation of a nonlinear elliptic problem arising in a non-Newtonian fluid flow model in glaciology. *ESAIM: Math. Model. Numer. Anal.* **37**(1), 175–186 (2003)
12. Huang, Y.X.: On eigenvalue problems of the  $p$ -Laplacian with Neumann boundary conditions. *Proc. Am. Math. Soc.* **109**, 177–184 (1990)
13. Jankowski, T.: Positive solutions of one-dimensional  $p$ -Laplacian boundary value problems for fourth-order differential equations with deviating arguments. *J. Optim. Theory Appl.* **149**(1), 47–60 (2011)
14. Kim, K.Y.: Error estimates for a mixed finite volume method for the  $p$ -Laplacian problem. *Numer. Math.* **101**(1), 121–142 (2005)
15. Kong, L.: Homoclinic solutions for a second order difference equation with  $p$ -Laplacian. *Appl. Math. Comput.* **247**, 1113–1121 (2014)
16. Li, X., Dong, H.: The element-free Galerkin method for the nonlinear  $p$ -Laplacian equation. *Comput. Math. Appl.* **75**(7), 2549–2560 (2018)
17. Li, X., Li, S.: Analyzing the nonlinear  $p$ -Laplacian problem with the improved element-free Galerkin method. *Eng. Anal. Bound. Elem.* **100**, 48–58 (2019)
18. Liu, W., Yan, N.: Quasi-norm local error estimators for  $p$ -Laplacian. *SIAM J. Numer. Anal.* **39**(1), 100–127 (2001)
19. Liu, Y.: Positive solutions of mixed type multi-point non-homogeneous BVPs for  $p$ -Laplacian equations. *Appl. Math. Comput.* **206**(2), 796–805 (2008)
20. Meng, G., Yan, P., Zhang, M.: Minimization of eigenvalues of one-dimensional  $p$ -Laplacian with integrable potentials. *J. Optim. Theory Appl.* **156**(2), 294–319 (2013)
21. Mirzaei, D., Dehghan, M.: MLPG approximation to the  $p$ -Laplace problem. *Comput. Mech.* **46**(6), 805–812 (2010)
22. Motreanu, D., Winkert, P.: The Fucik spectrum for the negative  $p$ -Laplacian with different boundary conditions. In: *Nonlinear Analysis*, pp. 471–485. Springer, New York (2012)
23. Naraveni, R., Chaudhary, S., Srinivas Kumar, V.V.K.: Minimization techniques for  $p(x)$ -Laplacian problem using WEB-spline based mesh-free method. *Int. J. Comput. Math.* **97**(3), 667–686 (2020)
24. Oberman, A.M.: Finite difference methods for the infinity Laplace and  $p$ -Laplace equations. *J. Comput. Appl. Math.* **254**, 65–80 (2013)
25. Shivanian, E.: Analysis of meshless local and spectral meshless radial point interpolation (MLRPI and SMRPI) on 3-D nonlinear wave equations. *Ocean Eng.* **89**, 173–188 (2014)

26. Shivanian, E.: A new spectral meshless radial point interpolation (SMRPI) method: a well-behaved alternative to the meshless weak forms. *Eng. Anal. Bound. Elem.* **54**, 1–12 (2015)
27. Shivanian, E.: Spectral meshless radial point interpolation (SMRPI) method to two-dimensional fractional telegraph equation. *Math. Methods Appl. Sci.* **39**(7), 1820–1835 (2016)
28. Shivanian, E., Jafarabadi, A.: The spectral meshless radial point interpolation method for solving an inverse source problem of the time-fractional diffusion equation. *Appl. Numer. Math.* **129**, 1–25 (2018)
29. Wendland, H.: Error estimates for interpolation by compactly supported radial basis functions of minimal degree. *J. Approx. Theory* **93**(2), 258–272 (1998)

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)