Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **12**, No. 5, pp. 1113-1136

DOI: 10.4208/aamm.OA-2019-0033
October 2020

The Modified Localized Method of Approximated Particular Solutions for Linear and Nonlinear Convection-Diffusion-Reaction PDEs

Wen Li[1,2], Kalani Rubasinghe[2], Guangming Yao[2,3,*] and L. H. Kuo[4]

[1] *College of Big Data Science, Taiyuan University of Technology, China*
[2] *Department of Mathematics, Clarkson University, USA*
[3] *Centre de Recherches Mathématiques, Université de Montréal, Canada*
[4] *Department of Mathematics, University of West Florida, USA*

Received 2 February 2019; Accepted (in revised version) 29 August 2019

Abstract. In this paper, a kernel based method, the modified localized method of approximated particular solutions (MLMAPS) [16, 23] is utilized to solve unsteady-state linear and nonlinear diffusion-reaction PDEs with or without convections. The time-space and spatial space are discretized by the higher-order Houbolt method with various time step sizes and the MLMAPS, respectively. The local truncation error associated with the time discretization is $\mathcal{O}(h^4)$, where h is the largest time step size used. The spatial domain is then treated by a special kernel, the integrated polyharmonic splines kernels together with low-order polynomial basis. Typical computational algorithms require a trade off between accuracy and rate of convergency. However, the experimental analysis has shown high accuracy and fast convergence of the proposed method.

AMS subject classifications: 65N35, 65N99

Key words: Polyharmonic spline, Houbolt method, time-dependent PDEs, method of approximated particular solutions, MLMAPS, convection-diffusion-reaction, nonlinear, kernel methods.

1 Introduction

The domain type meshfree methods utilizing radial basis functions, such as Kansa's method [11, 12] and the method of approximated particular solutions (MAPS) [3, 4], are classified as global meshfree methods because the methods result in the creation of a dense linear system. Yao et al. further localized the MAPS into the localized MAPS (LMPAS) [21] which allows the creation of a sparse system. This is especially useful for

*Corresponding author.
Email: gyao@clarkson.edu (G. M. Yao)

solving large-scale problems. The LMAPS utilizes the collocation scheme on overlapping local domains with integrated radial basis functions (RBFs), such as multiquadrics, inverse multiquadrics, and Matern. This technique drastically reduces the storage size of the collocation matrix. This improved the computational efficiency of the method for solving large-scale partial differential equations (PDEs). This allows the LMAPS to compete with the traditional numerical methods such as the finite element method (FEM) for large-scale problems. Since then, the LMAPS has been applied to various kinds of differential equations, such as the biharmonic equation [15], near-singular PDEs [24], the unsteady Burgers' equations [17, 27], convection-diffusion equations [2], 3D nonlinear Schrödinger equations [18], and wave equations [10], as well as unsteady Navier-Stokes problem [7, 26].

A modified LMAPS (MLMAPS) presented in [22] shows a significant improvement in terms of accuracy by using integrated polyharmonic splines in radial space together with the polynomial basis for linear and nonlinear elliptic PDEs in 2D and 3D. Since then, the method has been applied to time-dependent PDEs [16]. On the other hand, the Houbolt method in [9, 19, 25] is a high-order accurate time discretization scheme. Note that MLMAPS can be classified as a kernel-based method, which has similar ideas as the generalized finite difference method or radial basis function- finite difference method. The main difference is that MLMAPS uses commonly used kernels in the integrated forms, in addition to low-order polynomial basis. This is a combination of basis functions that amazingly preserve the high accuracy of the polynomial basis and flexibility of the kernel based methods.

In this paper, MLMAPS is coupled with the Houbolt method to solve linear or nonlinear diffusion-reaction PDEs with or without convection terms:

$$\frac{\partial u}{\partial t} = \mathcal{D}u + f(\boldsymbol{x},t), \qquad \boldsymbol{x}\in\Omega\subset\mathbb{R}^d, \tag{1.1}$$

with the boundary condition $\mathcal{B}u(\boldsymbol{x},t)=g(\boldsymbol{x},t)$, $\boldsymbol{x}\in\partial\Omega$, and the initial condition $u(\boldsymbol{x},0)=u^0$, $\boldsymbol{x}\in\Omega\cup\partial\Omega$, where $\mathcal{D}$ is linear or nonlinear diffusion-convection differential operator, f is a reaction function, and $\mathcal{B}$ is a linear or nonlinear boundary differential operator, functions f, g and u_0 are known, with physical domain Ω in $\mathbb{R}^d$. We will combine the implicit Euler method, the Houbolt method, and MLMAPS to solve this type of PDEs.

The rest of the paper is organized as follows: In Section 2, the first few time-steps are discretized by the traditional implicit time-stepping method with small evenly-spaced time-step h_0. This allows the Houbolt method to be used in the following time-steps after first three steps, the time step will jump from h_0 to a relatively larger time-step h. After transitioning to h, the third order Houbolt method will be used for the evenly-spaced time discretization with time-step h. The error analysis associated with time discretization is presented at the end of this section. The time discretization transforms the given time-dependent PDE to a series of elliptical differential equations. Therefore, in Section 3, the spatial discretization using MLMAPS with integrated polyharmonic splines together with polynomial basis are introduced. Section 4 illustrates the performance of the nu-

merical scheme by four examples.

2 Time discretization

To discretize the time variable, the time domain of the PDE is discretized by implicit method and the Houbolt finite difference method [9, 19, 25]. In order to fully implement the Houbolt method, we need to know the initial values of the first three time-steps, in which the implicit Euler method is used with tiny time step size h_0. The Houbolt finite difference is then used with a relatively large time step size h. Note that we need to perform nonuniform time steps at step four and five to transform the time-step size from h_0 to h. The details are introduced as follows. Assume that the time-space is discretized into $t=t_k$, $k=1,2,3,\cdots$, where $t_1=0$ is the initial time and

$$t_k=\begin{cases}(k-1)h_0, & k=1,2,3,4,\\ 3h_0+(k-4)h, & k=5,6,\cdots,\end{cases} \tag{2.1}$$

where $u^k\approx u(\boldsymbol{x},t_k)$, $k=1,\cdots$, as the approximation of u at t_k. Next subsection introduces the detailed time discretization.

Let the initial three time-step size be h_0, which is very small. That implies $t_k=(k-1)h_0$, $k=1,2,3,4$. The initial condition leads to $u^1=u(\boldsymbol{x},t_1)$ to be known, where $t_1=0$. The implicit Euler method with a very tiny step h_0 will be implemented to obtain the approximated values of the first three time steps, u^2, u^3 and u^4. To jump from the implicit Euler to Houbolt method, that is to jump from the tiny initial time steps h_0 to a relatively larger time step h, we need various time stepping schemes, at least at the first few steps:

- For the first three steps, the implicit Euler method implies that

$$\frac{\partial u^{k+1}}{\partial t}=\frac{1}{h_0}\left(u^{k+1}-u^k\right).$$

  Note that this time-discretization leads to a local truncation error $\mathcal{O}(h_0^2)$. Thus, Eq. (1.1) can be written as

$$\mathcal{D}u^{k+1}-\frac{1}{h_0}u^{k+1}=-\frac{1}{h_0}u^k-f^{k+1},\quad \boldsymbol{x}\in\Omega, \tag{2.2}$$

  where $f^{k+1}=f(\boldsymbol{x},t_{k+1})$. The boundary conditions are given through the original system

$$\mathcal{B}u^{k+1}(\boldsymbol{x})=g^{k+1}(\boldsymbol{x}),\quad \boldsymbol{x}\in\partial\Omega, \tag{2.3}$$

  where $g^{k+1}(\boldsymbol{x})=g(\boldsymbol{x},t_{k+1})$. MLMAPS will be introduced in the next section to solve Eqs. (2.2)-(2.3).

- For time-step five to approximate u^5 with time discretization $[h_0,h_0,h]$ for $t_5=3h_0+h$, assume a numerical scheme of form

$$a_1u^5+a_2u^4+a_3u^3+a_4u^2=\frac{\partial u^5}{\partial t}. \tag{2.4}$$

This requires the Taylor series expansions of u^4, u^3, and u^2 at t_5 as follows:

$$u^4=u^5-h\frac{\partial u^5}{\partial t}+\frac{(h)^2}{2}\frac{\partial^2u^5}{\partial t^2}-\frac{(h)^3}{6}\frac{\partial^3u^5}{\partial t^3}+\mathcal{O}(h^4), \tag{2.5a}$$

$$u^3=u^5-(h+h_0)\frac{\partial u^5}{\partial t}+\frac{(h+h_0)^2}{2}\frac{\partial^2u^5}{\partial t^2}-\frac{(h+h_0)^3}{6}\frac{\partial^3u^5}{\partial t^3}+\mathcal{O}((h+h_0)^4), \tag{2.5b}$$

$$u^2=u^5-(h+2h_0)\frac{\partial u^5}{\partial t}+\frac{(h+2h_0)^2}{2}\frac{\partial^2u^5}{\partial t^2}-\frac{(h+2h_0)^3}{6}\frac{\partial^3u^5}{\partial t^3}+\mathcal{O}((h+2h_0)^4), \tag{2.5c}$$

where

$$u^k\approx u(\boldsymbol{x},t_k), \qquad \partial u^{k+1}/\partial t\approx\partial u(\boldsymbol{x},t_{k+1})/\partial t,$$
$$\partial^2u^{k+1}/\partial t^2\approx\partial^2u(\boldsymbol{x},t_{k+1})/\partial t^2, \qquad \partial^3u^{k+1}/\partial t^3\approx\partial^3u(\boldsymbol{x},t_{k+1})/\partial t^3.$$

Thus, Taylor series expansion requires solving the following linear system

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & h & h_0+h & 2h_0+h \\ 0 & h^2 & (h_0+h)^2 & (2h_0+h)^2 \\ 0 & h^3 & (h_0+h)^3 & (2h_0+h)^3 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}=\begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \end{bmatrix}. \tag{2.6}$$

Note that the first equation comes from grouping the coefficients of u^{k+1} in Eqs. (2.4)-(2.5c), and second equation comes from grouping the coefficients of $\partial u^{k+1}/\partial t$, and $\partial^2u^{k+1}/\partial t^2$ for the third equation, and last for $\partial^3u^{k+1}/\partial t^3$. The above linear system can be solved analytically provided that $h_0(h+2h_0)\neq 0$ and $h(h+h_0)\neq 0$. This leads to

$$a_1=\frac{3h^2+6hh_0+2h_0^2}{h(h+h_0)(h+2h_0)}, \tag{2.7a}$$

$$a_2=-\frac{(h+h_0)(h+2h_0)}{2hh_0^2}, \tag{2.7b}$$

$$a_3=\frac{h(h+2h_0)}{h_0^2(h+h_0)}, \tag{2.7c}$$

$$a_4=-\frac{h(h+h_0)}{2h_0^2(h+2h_0)}. \tag{2.7d}$$

Note that this time-discretization leads to a local truncation error $\mathcal{O}(h^4)+\mathcal{O}((h+h_0)^4)+\mathcal{O}((h+2h_0)^4)$. The linear system can be solved numerically in MATLAB

with user-defined h and h_0, or various time-step sizes. The numerical scheme for the original PDE then is

$$\mathcal{D}u^5 - a_1u^5 = a_2u^4 + a_3u^3 + a_4u^2 - f^5, \quad \boldsymbol{x} \in \Omega, \tag{2.8}$$

with the same form of the boundary condition

$$\mathcal{B}u^5 = g^5, \quad \boldsymbol{x} \in \partial\Omega. \tag{2.9}$$

This will be solved using MLMAPS in Section 3.

- For time-step six, $t_6 = 3h_0 + 2h$, with time discretization $[h_0, h, h]$, similar to previous time-step, to approximate u^6, the Taylor series expansion of u^5, u^4 and u^3 at t_6 is needed. Assume a numerical scheme of form

$$a_1u^6 + a_2u^5 + a_3u^4 + a_4u^3 = \frac{\partial u^6}{\partial t}. \tag{2.10}$$

This requires to solve the following linear system

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & h & 2h & h_0+2h \\ 0 & h^2 & (2h)^2 & (h_0+2h)^2 \\ 0 & h^3 & (2h)^3 & (h_0+2h)^3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \end{bmatrix}. \tag{2.11}$$

The above linear system can be solved analytically provided that $h_0(2h^2 + 3hh_0 + h_0^2) \neq 0$ and $h \neq 0$. This leads to a solution

$$a_1 = \frac{8h + 3h_0}{4h^2 + 2hh_0}, \tag{2.12a}$$

$$a_2 = -\frac{2(2h + h_0)}{h(h + h_0)}, \tag{2.12b}$$

$$a_3 = \frac{1}{2h} + \frac{1}{h_0}, \tag{2.12c}$$

$$a_4 = -\frac{2h^2}{2h^2h_0 + 3hh_0^2 + h_0^3}. \tag{2.12d}$$

Note that this time-discretization leads to a local truncation error $\mathcal{O}(h^4) + \mathcal{O}((2h + h_0)^4)$. Again, the above linear system can be solved numerically in MATLAB with user-defined h and h_0. The numerical scheme then is

$$\mathcal{D}u^6 - a_1u^6 = a_2u^5 + a_3u^4 + a_4u^3 - f^6, \quad \boldsymbol{x} \in \Omega, \tag{2.13}$$

with the same type of boundary condition

$$\mathcal{B}u^6 = g^6, \quad \boldsymbol{x} \in \partial\Omega. \tag{2.14}$$

Again, this will be solved by MLMAPS in Section 3.

- For seventh time step and all the steps afterwards, the Houbolt method requires three third order Taylor series expansions of u^k, u^{k-1}, and u^{k-2} at t_{k+1}. Note that now we have an uniform time steps h between t_{k-2}, t_{k-1}, t_k and t_{k+1}. After solving the system

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & h & 2h & 3h \\ 0 & h^2 & (2h)^2 & (3h)^2 \\ 0 & h^3 & (2h)^3 & (3h)^3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} \tag{2.15}$$

with $h \neq 0$, we have that $a_1 = \frac{11}{6h}$, $a_2 = \frac{-18}{6h}$, $a_3 = \frac{9}{6h}$ and $a_4 = -\frac{2}{6h}$. Thus,

$$\frac{\partial u^{k+1}}{\partial t} = \frac{1}{6h}\left(11u^{k+1} - 18u^k + 9u^{k-1} - 2u^{k-2}\right).$$

Note that this time-discretization leads to a local truncation error $\mathcal{O}(h^4)$. Thus, Eq. (1.1) can be written as

$$\mathcal{D}u^{k+1} - \frac{11}{6h}u^{k+1} = \frac{1}{6h}\left(-18u^k + 9u^{k-1} - 2u^{k-2}\right) - f^{k+1}, \quad \boldsymbol{x} \in \Omega, \tag{2.16}$$

where $f^{k+1} = f(\boldsymbol{x}, t_{k+1})$. The boundary conditions are given through the original system

$$\mathcal{B}u^{k+1} = g^{k+1}, \quad \boldsymbol{x} \in \partial\Omega, \tag{2.17}$$

where $u^{k+1} \approx u(\boldsymbol{x}, t_{k+1})$ and $g^{k+1} = g(\boldsymbol{x}, t_{k+1})$. This is an equation of modified Helmholtz type.

At any time-step, the local truncation error from such scheme for approximating first derivative with respect to time can be bounded by

$$\begin{aligned} &\left|\frac{\partial u^{k+1}}{\partial t} - \frac{\partial u(\boldsymbol{x}, t_{k+1})}{\partial t}\right| \\ =&\max\Big\{\mathcal{O}(h_0^2), \mathcal{O}(h^4), \mathcal{O}(h^4) + \mathcal{O}((h_0+2h)^4), \\ &+ \mathcal{O}(h^4) + \mathcal{O}((h+h_0)^4) + \mathcal{O}((h+2h_0)^4)\Big\}. \end{aligned}$$

This can be seen as $\mathcal{O}(h^4)$ when h_0 is chosen to be tenth of h. If we desire to use an even higher-order time discretization scheme, we can expand a higher-order Taylor series expansions of $u^k, u^{k-1}, u^{k-2}, \cdots, u^{k-l}$ at t_{k+1}.

Thus, through various time discretization schemes, Eqs. (2.2)-(2.3) need to solved in the first three time-steps, then Eqs. (2.8)-(2.9) at step five and Eqs. (2.13)-(2.14) at step six, and finally Eqs. (2.16)-(2.17) at all steps afterwards.

3 Spatial discretization

In every time-step, a general elliptical equation needs to be solved. For our simplicity, we will rewrite those elliptical equations in the following form:

$$\tilde{\mathcal{D}}w(\boldsymbol{x})=\tilde{f}(\boldsymbol{x}), \tag{3.1a}$$

$$\mathcal{B}w(\boldsymbol{x})=g(\boldsymbol{x}), \tag{3.1b}$$

where $\tilde{\mathcal{D}}$ is the differential operator appeared in Eqs. (2.2), (2.8), (2.13), or (2.16), $w(\boldsymbol{x})=u(\boldsymbol{x},t_{k+1})\approx u^{k+1}(\boldsymbol{x})=\hat{w}(x)$, and $\tilde{f}(\boldsymbol{x})$ various depending on which time step is being considered. For example after step seven, $\tilde{\mathcal{D}}=\mathcal{D}-\frac{11}{6h}\boldsymbol{I}$ and

$$\tilde{f}=\frac{1}{6h}\Big(-18u^k(\boldsymbol{x})+9u^{k-1}(\boldsymbol{x})-2u^{k-2}(\boldsymbol{x})\Big)-f^{k+1}(\boldsymbol{x}).$$

MLMAPS is presented in this section to solve (3.1a)-(3.1b). In the modified method, the spatial domain is discretized into a set of points in vector format $\{\boldsymbol{x}_i\}_{i=1}^N$, where N represents the total number of discrete points in the domain and on the boundary. Assume that the first n_i points are the interior points, and next n_b points are the boundary points. Note that $N=n_i+n_b$. To find a numerical solution that makes the residual

$$R(\boldsymbol{x})=\tilde{\mathcal{D}}\hat{w}(\boldsymbol{x})-\tilde{f}(\boldsymbol{x}) \tag{3.2}$$

small, the idea is to expand the numerical solution $\hat{w}$ in a finite-dimensional subspace $\mathcal{P}_N$ of some Hilbert space $\mathcal{W}$: $\mathcal{P}_N=\text{span}\{\phi_1(\boldsymbol{x}),\phi_2(\boldsymbol{x}),\cdots,\phi_N(\boldsymbol{x})\}$. Let

$$\hat{w}(\boldsymbol{x})=\sum_{j=1}^{N}\alpha_j\phi_j(\boldsymbol{x}).$$

The focus of this paper is on choosing RBFs as ϕ, particularly the polyharmonic splines. However, the polyharmonic spline is not used directly, instead, the integrated form will be used. In this paper, ϕ is the particular solution of Laplacian operator Δ with respect to polyharmonic splines $\psi(r)$, in two dimensional space. Therefor, the basis ϕ is found by integrating polyharmonic spline $\psi(r)=r^{2m}\ln(r)$ in the following manner in two dimensional space:

$$\phi(r)=\int\frac{1}{r}\int r\cdot\psi(r)drdr. \tag{3.3}$$

By direct integration, we obtain that

$$\phi(r)=\begin{cases}\dfrac{r^{2m+2}\ln(r)}{4(m+1)^2}-\dfrac{r^{2m+2}}{4(m+1)^3}, & r>0,\\ 0, & r=0.\end{cases} \tag{3.4}$$

Additionally, to avoid singularity a low order polynomial space of dimension q is needed:

$$\mathcal{P}=\text{span}\{p_1,\cdots,p_q\}.$$

The solution to Eqs. (3.1a)-(3.1b) can be approximated at the given discrete nodes $\boldsymbol{x}_j$, $j=1,\cdots,N$. The word "localized" is coming from influence domain Ω_x of every discrete point, $\boldsymbol{x}$, in the domain which contains the n nearest discrete points to $\boldsymbol{x}$ where n is much smaller than the total number of discrete points, $n\ll N$. For convenience, denote the local domain of influence at interpolation point $\boldsymbol{x}_j$ by $\Omega_j=\{\boldsymbol{x}_i,\, i=1,\cdots,n\}$, then

$$w(\boldsymbol{x})\approx\hat{w}(\boldsymbol{x})=\sum_{i=1}^{n}\alpha_i\phi(\|\boldsymbol{x}-\boldsymbol{x}_i\|)+\sum_{l=1}^{q}\alpha_{n+l}p_l(\boldsymbol{x}),\quad \boldsymbol{x}\in\Omega_j. \tag{3.5}$$

Thus,

$$\sum_{i=1}^{n}\alpha_i\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_i\|)+\sum_{l=1}^{q}\alpha_{n+l}p_l(\boldsymbol{x}_j)=\hat{w}(\boldsymbol{x}_j),\qquad j=1,\cdots,n, \tag{3.6a}$$

$$\sum_{i=1}^{n}\alpha_i p_l(\boldsymbol{x}_i)=0,\qquad l=1,\cdots,q. \tag{3.6b}$$

Note that Eqs. (3.6a)-(3.6b) is a system of equations with $n+q$ unknowns,

$$\boldsymbol{\alpha}=[\alpha_1,\cdots,\alpha_n,\alpha_{n+1},\cdots,\alpha_{n+q}]^T,$$

which are to be determined. Denote the above small system of linear equations in matrix-vector form, we have $\boldsymbol{A\alpha}=\hat{\boldsymbol{w}}$, where $\boldsymbol{A}$ is the coefficient matrix of the linear system, and $\hat{\boldsymbol{w}}=[\hat{w}(\boldsymbol{x}_1),\cdots,\hat{w}(\boldsymbol{x}_n),\underbrace{0,\cdots,0}_{\text{q 0's}}]$. The unknown coefficients $\boldsymbol{\alpha}=\boldsymbol{A}^{-1}\hat{\boldsymbol{w}}$. Thus,

$$\tilde{\mathcal{D}}\hat{w}(\boldsymbol{x}_j)=\sum_{i=1}^{n}\alpha_i\tilde{\mathcal{D}}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_i\|)+\sum_{l=1}^{q}\alpha_{n+l}\tilde{\mathcal{D}}p_l(\boldsymbol{x}_j)=\mathcal{B}^T\boldsymbol{A}^{-1}\hat{\boldsymbol{w}}, \tag{3.7}$$

where

$$\mathcal{B}^T=[\tilde{\mathcal{D}}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_1\|),\tilde{\mathcal{D}}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_2\|),\cdots,\tilde{\mathcal{D}}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_n\|),\tilde{\mathcal{D}}p_1(\boldsymbol{x}_j),\cdots,\tilde{\mathcal{D}}p_q(\boldsymbol{x}_j)].$$

Similarly, we have that

$$\boldsymbol{B}\hat{w}(\boldsymbol{x}_j)=\sum_{i=1}^{n}\alpha_i\boldsymbol{B}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_i\|)+\sum_{l=1}^{q}\alpha_{n+l}\boldsymbol{B}p_l(\boldsymbol{x}_j)=\boldsymbol{c}^T\boldsymbol{A}^{-1}\hat{\boldsymbol{w}}, \tag{3.8}$$

where

$$\boldsymbol{c}^T=[\boldsymbol{B}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_1\|),\boldsymbol{B}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_2\|),\cdots,\boldsymbol{B}\phi(\|\boldsymbol{x}_j-\boldsymbol{x}_n\|),\boldsymbol{B}p_1(\boldsymbol{x}_j),\cdots,\boldsymbol{B}p_q(\boldsymbol{x}_j)].$$

By direct collocation on the discrete nodes in the domain, we have

$$\mathcal{B}^T \boldsymbol{A}^{-1} \hat{\boldsymbol{w}} = f(\boldsymbol{x}_j), \qquad j=1,\cdots,n_i, \tag{3.9a}$$

$$\boldsymbol{c}^T \boldsymbol{A}^{-1} \hat{\boldsymbol{w}} = g(\boldsymbol{x}_j), \qquad j=n_i+1,\cdots,N. \tag{3.9b}$$

Note that $\mathcal{B}^T \boldsymbol{A}^{-1}$ is a "notation" only, in the sense that the inverse matrix were never computed in practice. In fact, a small linear system $\boldsymbol{A}\boldsymbol{v} = \mathcal{B}$ is solved, and the resulted solution is $\boldsymbol{v}^T = \mathcal{B}^T \boldsymbol{A}^{-1}$. This is an $N \times N$ system of algebraic equations, and each equation in the system contains n non-zero entries. The system is a sparse system with N unknowns $\{\hat{w}(\boldsymbol{x}_j)\}_{j=1}^N$. Even though an additional polynomial basis needs to be added when we create local small linear systems in LMAPS using polyharmonic splines, the resulting global sparse system remains the same size as in the original LMAPS using other positive definite RBFs.

- Linear PDEs

  If the differential operator $\tilde{\mathcal{D}}$ is linear, the sparse system (3.9) is linear as well. Thus, many existed linear sparse matrix solver can be used directly in MATLAB or other computer languages.

- Nonlinear PDEs

  In the case that the differential operator $\tilde{\mathcal{D}}$ is nonlinear, a direct Picard method is used to solve the nonlinear system of algebraic equation (3.9). At each time step, a iteration is needed. The initial guess value at each time step is set to be the approximated solution at the end of last time step, i.e., $u^{k+1}(x,y) = u^k(x,y)$ for all $(x,y) \in \Omega \cup \partial\Omega$. Then the $u^{k+1}(x,y)$ is updated iteratively with stopping criteria of differences between two consecutive iterations is less than 10^{-12}. A numerical example on nonlinear Burger's equation is given in Section 4.

4 Numerical examples

In this section, we will apply the above numerical scheme to examples such as heat equation, convection-diffusion equation and nonlinear Burger's equation in 2D. The maximum absolute errors, the average absolute errors and the root mean squared errors are defined as

$$\text{max abs} = \max_{\boldsymbol{x}\in\Omega} |u^k(\boldsymbol{x}) - u(\boldsymbol{x},t_k)|, \tag{4.1a}$$

$$\text{ave abs} = \frac{1}{N} \sum_{\boldsymbol{x}\in\Omega} |u^k(\boldsymbol{x}) - u(\boldsymbol{x},t_k)|, \tag{4.1b}$$

$$\text{rms} = \sqrt{\frac{1}{N} \sum_{\boldsymbol{x}\in\Omega} |u^k(\boldsymbol{x}) - u(\boldsymbol{x},t_k)|^2}, \tag{4.1c}$$

where N is the total number of collocation points in $\Omega\cup\partial\Omega$. For numerical simplicity, we consider evenly-spaced spatial collocation points, in which

- number of points in each direction: n_p;
- number of boundary points: $n_b=4n_p$;
- number of interior points: $n_i=(n_p-1)^2$;
- total number of collocation points: $N=n_i+n_b=(n_p+1)^2$;
- number of points in local domains: n;
- initial small time-step size: h_0; and
- time-step size for the Houbolt method: h or Δt.

Example 4.1. In this example, we consider a simple heat equation in two-dimensional space [5,13]:

$$\begin{aligned}
&\frac{\partial u(x,y,t)}{\partial t}=\Delta u(x,y,t)+\sin(x)\sin(y)(2\sin(t)+\cos(t)), && (x,y)\in\Omega,\\
&u(x,y,t)=\sin(x)\sin(y)\sin(t), && (x,y)\in\partial\Omega,\\
&u(x,y,0)=0, && (x,y)\in\Omega\cup\partial\Omega.
\end{aligned}$$

This problem is chosen due to its simplicity and no other effect to the performance of the proposed method other than time and spatial discretizations.

The performance of the proposed method on a square domain of $[-2,2]^2$ (as shown in [5] and [13]) is much more accurate.

To show effectiveness of our proposed method on irregular domains, we consider a star-shape domain Ω which is defined by the following parametric equation:

$$\Omega=\{(x,y)|x=\rho\cos(\theta),\ y=\rho\sin(\theta),\ 0\le\theta<2\pi\},$$

where

$$\rho=1+\cos^2(4\theta).$$

The profiles of the computational domain is shown in Fig. 1.

Fig. 2 shows the profiles of the analytical solution and the absolute errors on the top. The maximum absolute errors, average absolute errors, and the root mean squared errors are plotted against time variable, as shown on the bottom of Fig. 2. The maximum absolute error at $t=9.9757$ is within 10^{-7} which is extremely accurate. Note that the number of points in the local domains is chosen as 35 while $N=61^2$, $\Delta x=\Delta y=4/60$, and $h_0=h=0.0015$. The order of polynomial basis is 4 and the order of polyharmonic splines is 1.

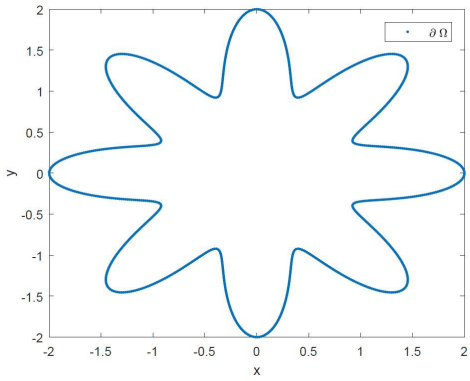

Figure 1: The computational domain in Example 4.1.

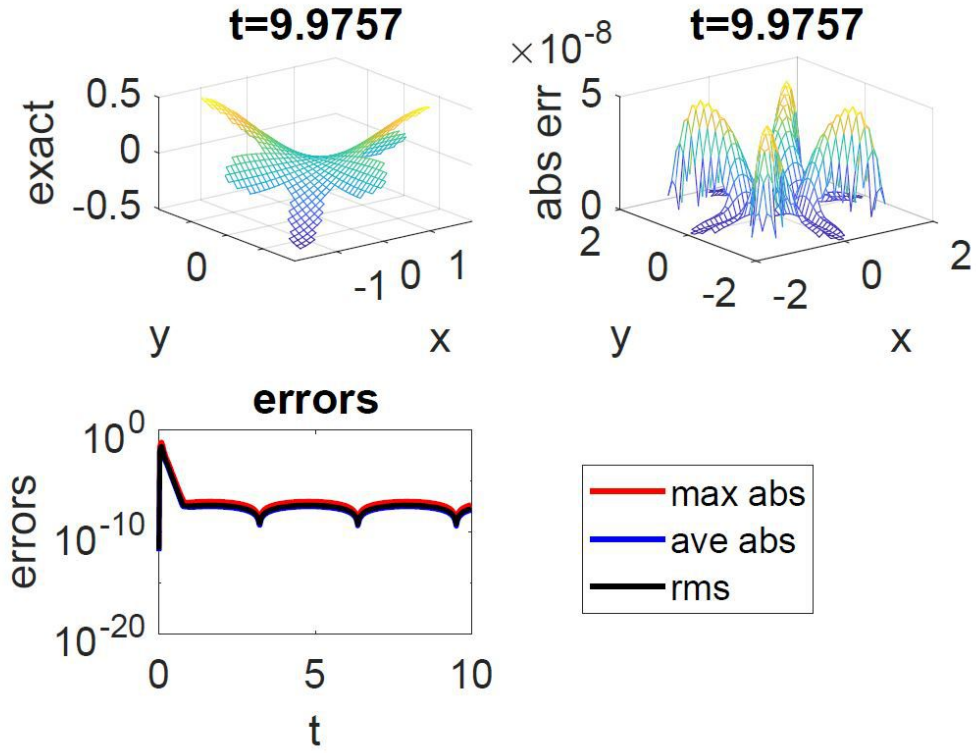

Figure 2: Top: profiles of the analytical solution and the absolute errors at $t=9.9757$ in Example 4.1. Bottom: the maximum absolute errors, the average absolute errors and the root mean squared errors versus time when $n=35$, $h=h_0=0.0015$ are chosen in Example 4.1.

Fig. 3 shows the rate of convergence of the proposed method in terms of time discretization. The rate is about 2.5 when considering approximation to the solution at $t=0.1$, and rate is about 3.3 when considering approximation to the solution at $t=0.4$.

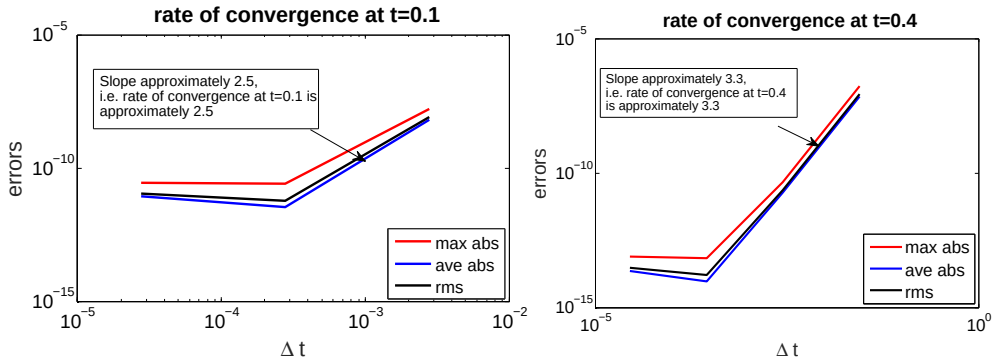

Figure 3: Rates of convergence of the proposed method in Example 4.1 when $t=0.1$ and $t=0.4$ are 2.5 and 3.3, respectively.

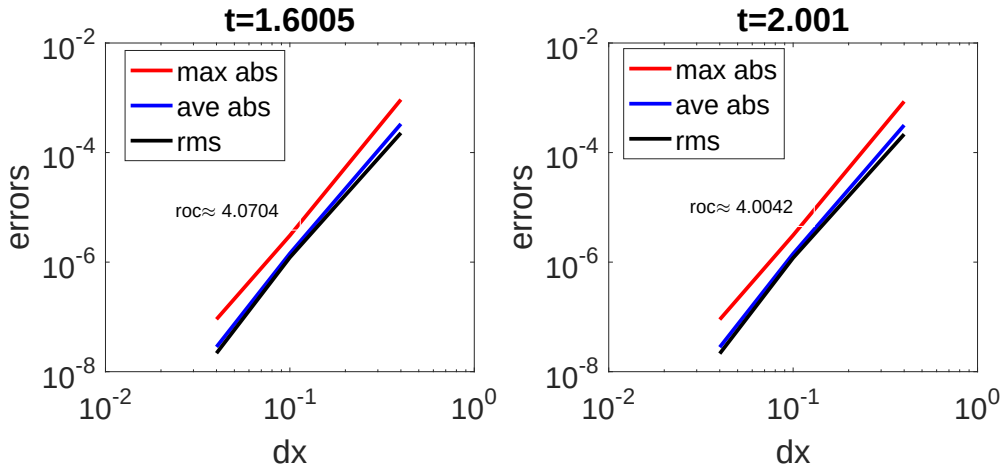

Figure 4: Rates of convergence of the proposed method in Example 4.1 when $t=1.6005$ and $t=2.001$ are 4.0704 and 4.0042, respectively.

This particular time is selected since it is the largest maximum/average absolute errors and the root mean square errors occur for the first time. When time is relatively large, the errors do not change very much with small h such as 10^{-4} in comparison with 10^{-2}. This is partially due to the errors from spatial discretization, in addition to the implicit time stepping scheme can tolerate bigger time-step sizes.

Fig. 4 shows the rates of convergence of the proposed method in Example 4.1 with respect to spatial discretization are 4.0704 and 4.0042, when $t=1.6005$ and $t=2.001$ respectively. Note the parameters in both cases are $n=35$ collocation points in local domains, $n_p=40$, $h=h_0=0.0015$, the order of polynomial basis is 4 and the order of polyharmonic splines is 1.

Example 4.2. Consider the parabolic problem

$$\frac{\partial u}{\partial t} = \Delta u + f \qquad \text{in} \quad \Omega, t > 0, \tag{4.2a}$$

$$u|_{\partial\Omega} = g \qquad \text{for} \quad t > 0, \tag{4.2b}$$

$$u|_{t=0} = u_0 \qquad \text{on} \quad \Omega \cup \partial\Omega, \tag{4.2c}$$

where f, g and u_0 are selected so that its exact solution is

$$u(x,y,t) = 0.8e^{-80[(x-r(t))^2+(y-s(t))^2]}, \tag{4.3}$$

where

$$r(t) = \frac{1}{4}(2+\sin(\pi t)), \quad s(t) = \frac{1}{4}(2+\cos(\pi t)). \tag{4.4}$$

(a) Consider $\Omega = [0,1]\times[0,1]$ [1,14]. The analytical solution Eq. (4.3) is a cone centered at $(1/2,3/4)$ at $t=0$, and then the center of the cone moves around the origin in a clockwise direction. After $t=2$, the cone will rotate back to the original center. Thus, we will consider to find the approximated solution for $0 \le t \le 2$.

To compare the proposed method with MFS-DRM in [1] and Kansa's method in [14], algorithm parameters are chosen to be the same as what were in the references, which are $h = 0.01$ and 19×19 uniform grid inside Ω with 32 boundary points uniformly distributed on $\partial\Omega$. The proposed method is clearly as accurate as the MFS-DRM and Kansa's method (around 2×10^{-3} in maximum norm).

Additionally, the method implemented in this paper is a localized method which can be used for large-scale problem with little loss of efficiency and accuracy, the method can easily employ more collocation points to achieve higher-accuracy. For example, when 41 by 41 collocation points in Ω with the same time-step sizes $h = 0.01$ and $h_0 = 0.1h$, order of the polyharmonic splines is 4, order of polynomials is 3, and 55 points in local domains are used, we can achieve maximum absolute error of $4.9114e-5$ accuracy.

Fig. 5 shows the profiles of the analytical solutions and the absolute errors, on the left and on the right respectively, at $t = 0.503$, 1.003, 1.503 and 2.003. The errors remain in the order of 5 magnitude.

Fig. 6, on the left, shows the maximum absolute errors, the average absolute errors and root mean squared errors as a function of time. The errors stay at similar order of magnitude, where 10^{-5} for the maximum absolute errors and 10^{-6} for the other two errors. The numerical accuracy is consistent with the truncation error $\mathcal{O}(h^3)$. On the right of Fig. 6, the rate of convergence of the proposed scheme is shown in terms of the spatial discretization. The rates are ranging from 2.3 to 2.5, although the rates dropped compared to the previous example.

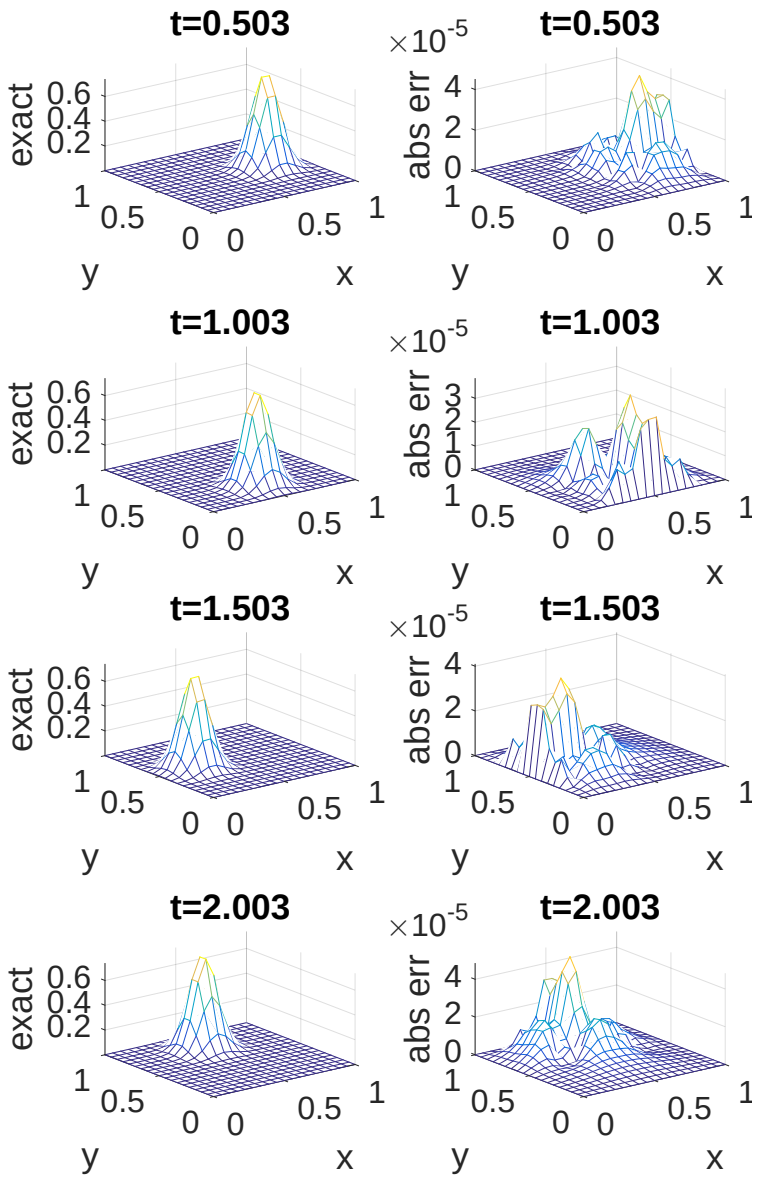

Figure 5: The profiles of the analytical solutions and the absolute errors in Example 4.2 at $t=0.503$, 1.003, 1.503 and 2.003, respectively.

(b) Next, we exam the same problem with a different domain, which is an irregular domain where Ω is shown in Fig. 7 and the boundary $\partial\Omega$ is defined by the following parametric equation:

$$\partial\Omega=\{(x,y)|x=\rho\cos\theta+1.5,\ y=\rho\sin\theta+1.5,\ 0\leq\theta<2\pi\},$$

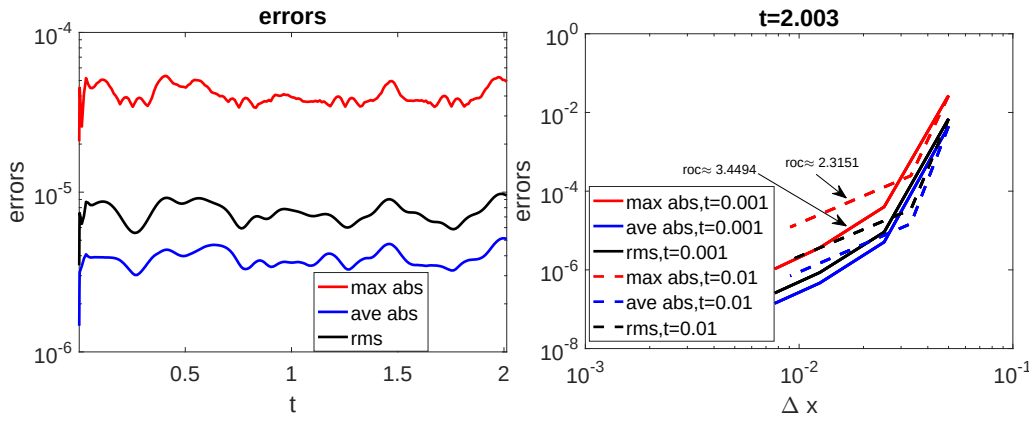

Figure 6: Left: The errors as functions of time in Example 4.2. Right: The rate of convergence of the proposed scheme in terms of the spatial discretization.

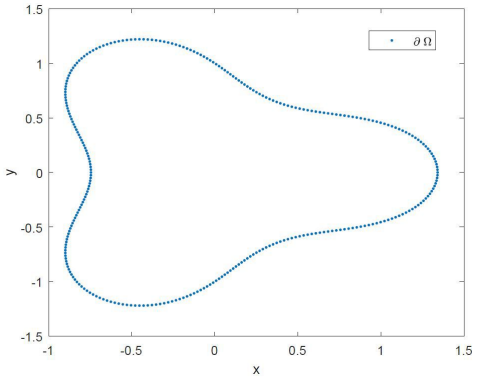

Figure 7: The computational domain in Example 4.2.

where

$$\rho=\left(\cos(3\theta)+\sqrt{2-\sin^2(3\theta)}\right)^{\frac{1}{3}}.$$

In the numerical implementation, we choose $n_i=9189$ and $n_b=300$ and the solution for $0\leq t\leq 2$ is approximated with a maximum absolute error of $1.2603e-5$ accuracy. Fig. 8 shows the profiles of the analytical solutions and the absolute errors in Example 4.2 with an irregular domain at $t=0.503$, 1.003, 1.503 and 2.003, respectively.

Fig. 9 on the left, shows the maximum absolute errors, the average absolute errors and root mean squared errors as a function of time. The orders of errors are 10^{-6}, 10^{-8}, 10^{-7}, respectively. The numerical accuracy of the method when applied to the irregular domain is consistent with the truncation error $\mathcal{O}(h^3)$. On the right of Fig. 9, the rate of convergence of the proposed scheme is shown in terms of the spatial discretization. The rates are approximately 4.2.

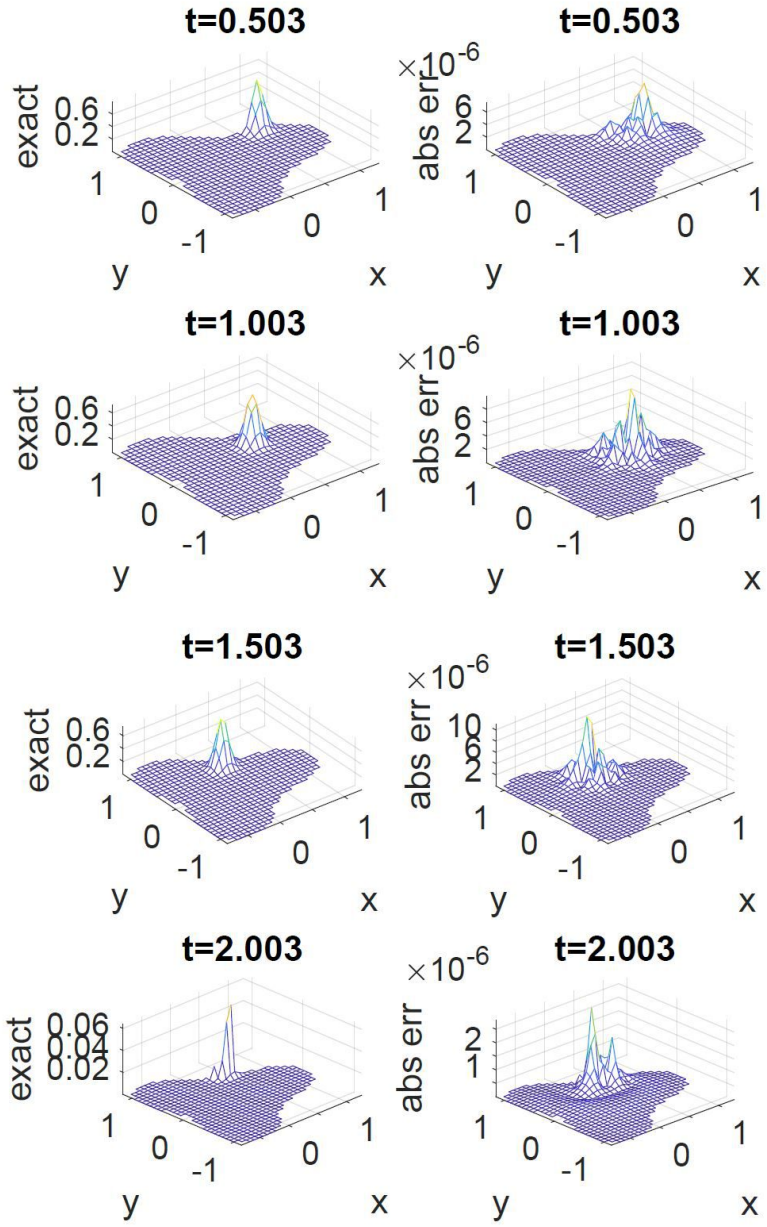

Figure 8: The profiles of the analytical solutions and the absolute errors in Example 4.3 at $t=0.503$, 1.003, 1.503 and 2.003, respectively.

Example 4.3. Next, we test the method by solving convection-diffusion equation [6,20]

$$\frac{\partial u}{\partial t} = d\Delta u - \boldsymbol{b}\cdot\nabla u, \qquad \Omega, t>0, \tag{4.5a}$$

$$u|_{\partial\Omega} = g, \qquad t>0, \tag{4.5b}$$

$$u|_{t=0} = u_0, \qquad \Omega\cup\partial\Omega, \tag{4.5c}$$

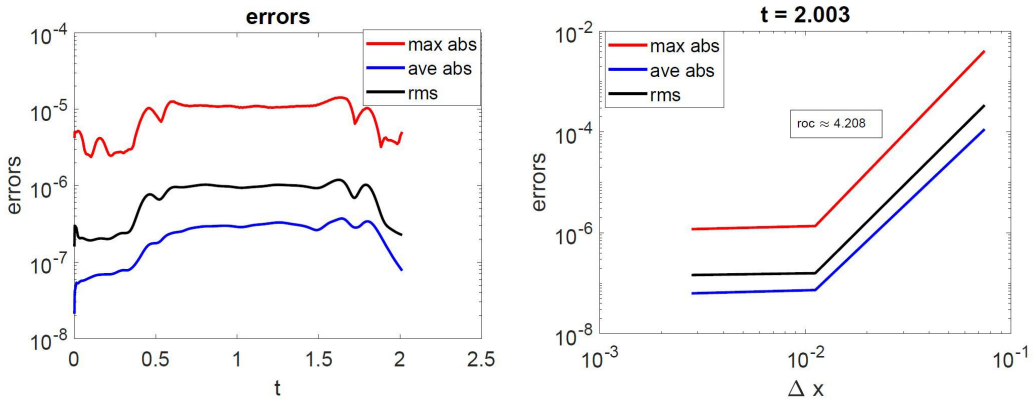

Figure 9: Left: The errors as functions of time in Example 4.2 with irregular domain. Right: The rate of convergence of the proposed scheme in terms of the spatial discretization.

where with $\Omega=[0,2]\times[0,2]$, $d=0.01$ and $\boldsymbol{b}=(b_1,b_2)=(0.8,0.8)$, g and u_0 are obtained from the analytical solution

$$u(x,y,t)=\frac{1}{1+4t}\exp\left[\frac{-(x-b_1t-0.5)^2}{d(1+4t)}-\frac{(y-b_2t-0.5)^2}{d(1+4t)}\right]. \tag{4.6}$$

Similar nodes arrangements and number of collocation points are used in this paper as seen in [6, 20], where $N=61^2$ uniformly distributed interpolation nodes and $n=13$ nearest nodes are selected in a local influence domain in this example. The order of polynomial basis and the order of polyharmonic splines are both 3 with $h_0=0.001$ and $h=0.01$. Fig. 10 shows the profiles of exact solutions at different time and the corresponding absolute errors.

The errors of our numerical solution over time are shown in Fig. 11. Compare to the results obtained by finite collocation (FC) method in [20], we get similar accuracy. Since the discretization and assembling process of FC method in [20] is almost same as the LMAPS, our method has the same order of computational complexity as FC method. The FC method in [20] is based on multiquadric (MQ) RBF. However, as we know, the choice of its shape parameter is still an ad hoc topic in the research community of radial basis function methods. The proposed MLMAPS is based on integrated polyharmonic splines with polynomial, which has no shape parameters to be determined. Moreover, in [20] the error of this problem increases over time. Using our method, it can be seen that the errors decrease as time goes by in Fig. 11. The reason is that the Gaussian pulse decreases and fades to the outside of the domain. This consistency of errors and solutions shows the stability of the proposed method. The rate of convergence with respect to time-step size for this problem at $t=2$ is approximately 2 which can be seen in Fig. 12.

Example 4.4. At last, we consider the nonlinear Burger's equation [8, 17, 20, 27]

$$\frac{\partial u}{\partial t}+uu_x+uu_y=c\Delta u \quad \text{in } \Omega, \quad t>0, \tag{4.7}$$

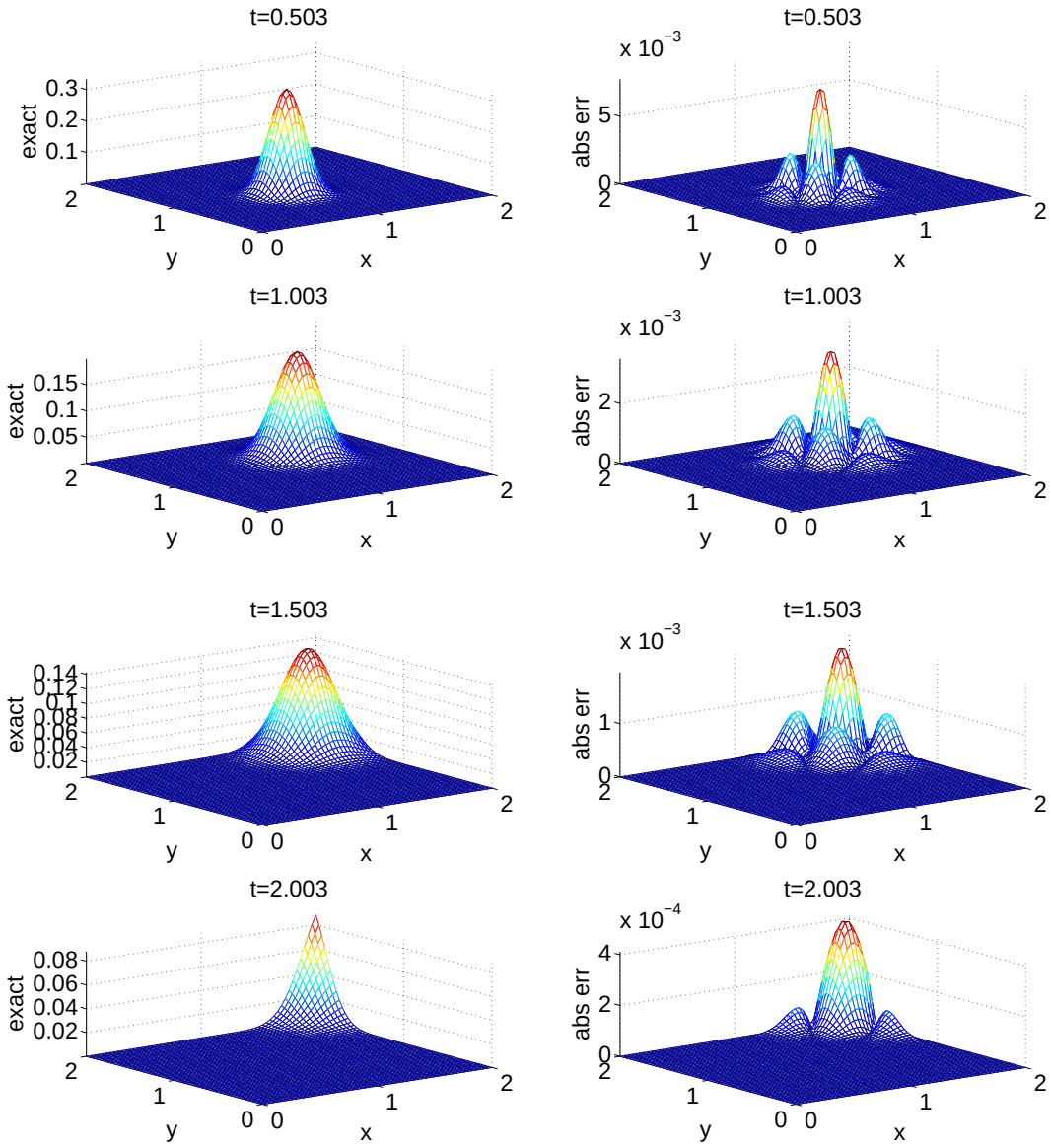

Figure 10: The profiles of the analytical solutions and the absolute errors in Example 4.3 at $t=0.503,1.003,1.503$ and 2.003 respectively.

where with $\Omega=[0,1]\times[0,1]$, $c=1/R$ where R is the Reynolds number. Initial and Dirichlet boundary conditions are obtained from the analytical solution

$$u(x,y,t)=\frac{1}{1+e^{(x+y-t)/2c}}. \tag{4.8}$$

In this example, nearest 6 nodes in the local influence domain are used. The order of

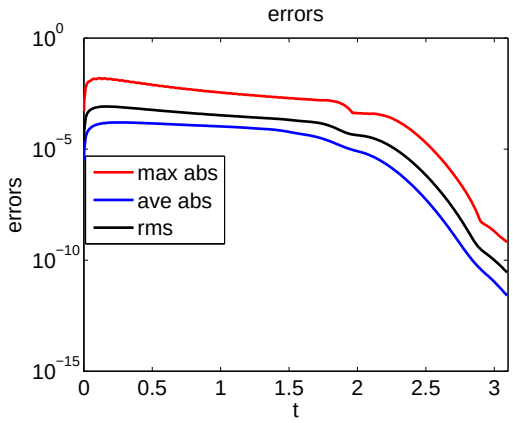

Figure 11: The errors versus time in Example 4.3 with $N=61^2$, $n=13$, $h=0.01$, and $h_0=0.001$.

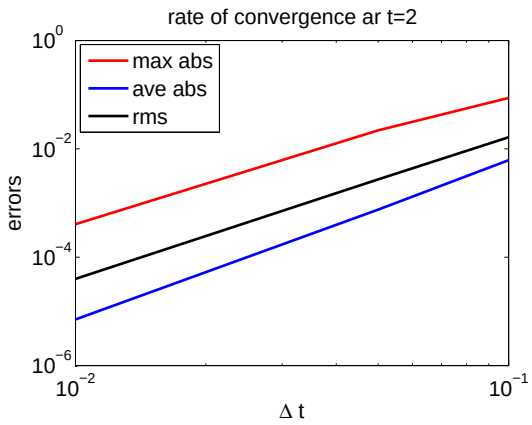

Figure 12: The rate of convergence of our method in Example 4.3 when time $t=2$ is approximately 2.

polynomial basis and the order of polyharmonic splines are both 2. The profiles of numerical solution and corresponding absolute errors when $c=0.2$ based on 31×31 uniform nodes are shown in Fig. 13.

Fig. 14 shows the root mean squared errors versus the number of collocation points at $t=1$ with different c's. In comparison with the results in Fig. 11 in [20], where FC method based on MQ is used, current results from the proposed methods in this paper are more accurate and consistent. The errors decrease with growth of values of c. That is because the greater the c is, the smoother the solution is. From both Figs. 14 and 15, it can be seen that the numerical results converge with respect to the number of collocation nodes, N used.

In [27], LMAPS based on MQ is used for solving Burger's equation. A "good" shape parameter in MQ has to be carefully selected in order to achieve reported accuracy. The optimal shape parameter is specifically effected by the number of collocation points, the parameter c in the Burger's equation, and domain size and shapes. In the proposed method using polyharmonic spline, there is no need to determine a shape parameter. Compare errors in Fig. 15 with the results in [27], where same c are used, our results are competitive and stable. This advantage is especially manifested in solving large-scale problems, because we do not need to change any parameters when the distance between nodes is getting smaller. It can be seen in Table 1 that the proposed method has robust performance for solving large-scale Burger's equation.

Combining with the Houbolt scheme, we can solve real problems with high order of convergence and stability. In this example, we pick $c=0.5$, $t=1$, $N=51\times51$ to test the rate of convergence with respect to h in Fig. 16. The rate of convergence is approximately 5.

In Fig. 17, we can see that our method is still very effective with greater time step $h=0.1$.

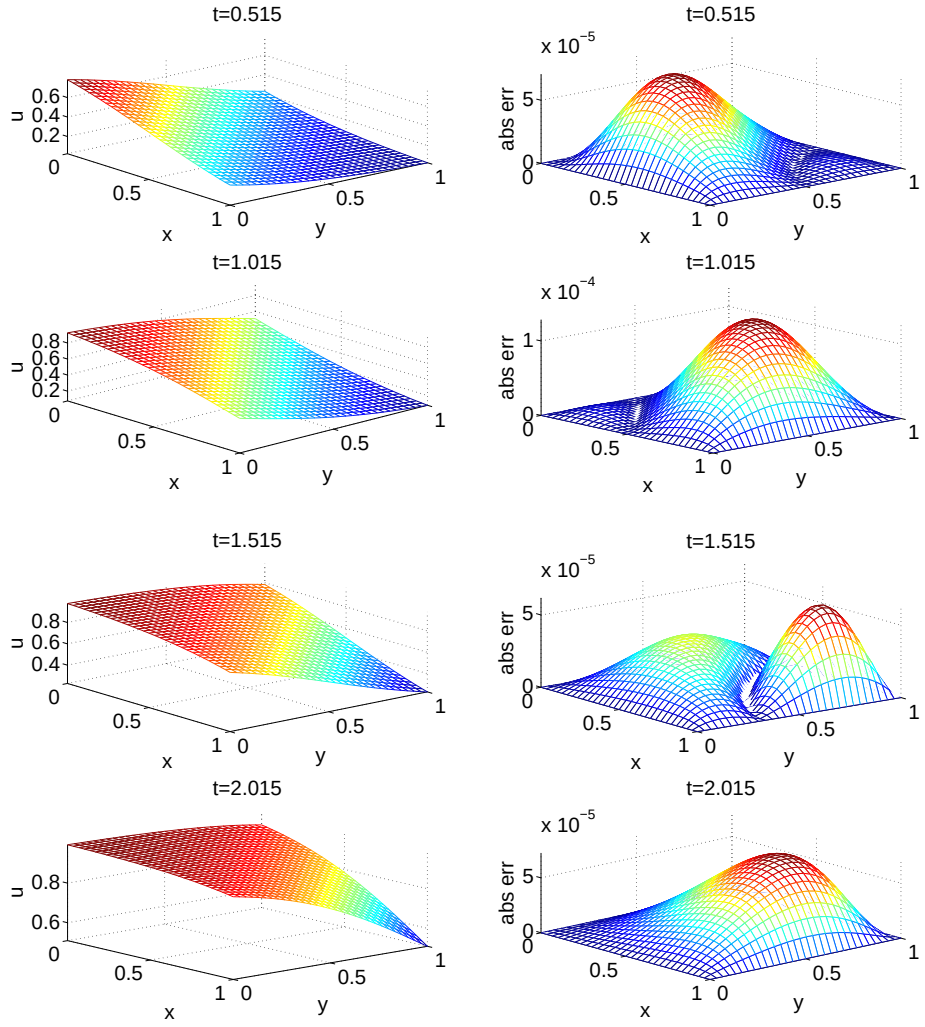

Figure 13: The profiles of the analytical solutions and the absolute errors in Example 4.4 at $t=0.515,1.015,1.515$ and 2.015 respectively with $c=0.2$, $N=31^2$, $h=0.05$, and $h_0=0.005$.

Table 1: Errors of MLMAPS for solving Burger's equation in Example 4.4 at $t=1$ with $h=0.05$, $h_0=0.005$, $N=101^2,201^2,301^2,401^2$.

N	$c=0.5$			$c=1$		
	max abs	ave abs	rms	max abs	ave abs	rms
10201	4.7643e-7	1.6536e-7	2.1851e-7	2.0602e-8	8.9441e-9	1.0851e-8
40401	2.7521e-7	8.6343e-8	1.1515e-7	7.1061e-9	2.5314e-9	3.3019e-9
90601	2.4032e-7	7.5904e-8	9.9748e-8	4.8652e-9	1.5690e-9	2.0787e-9
160801	2.2837e-7	7.2812e-8	9.4991e-8	4.1307e-9	1.3249e-9	1.7318e-9

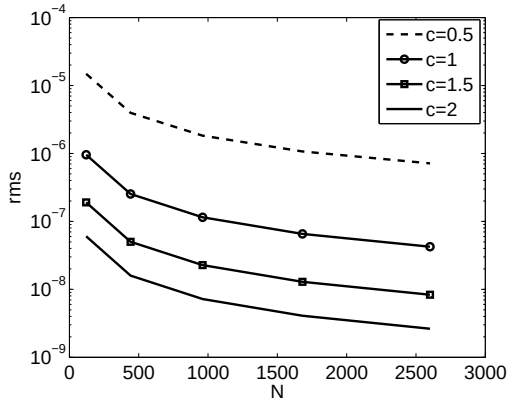

Figure 14: The root mean squared errors versus N in Example 4.4 at $t=1$ with $n=6$, $h=0.05$, $h_0=0.005$.

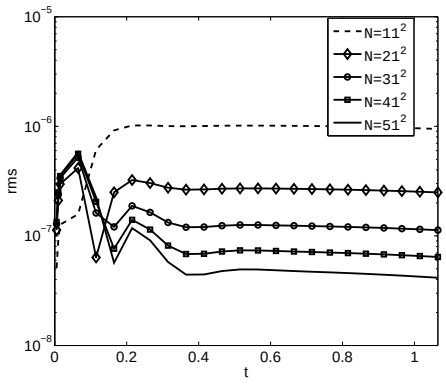

Figure 15: The root mean squared errors versus time in Example 4.4 with $c=1$, $n=6$, $h=0.05$, $h_0=0.005$.

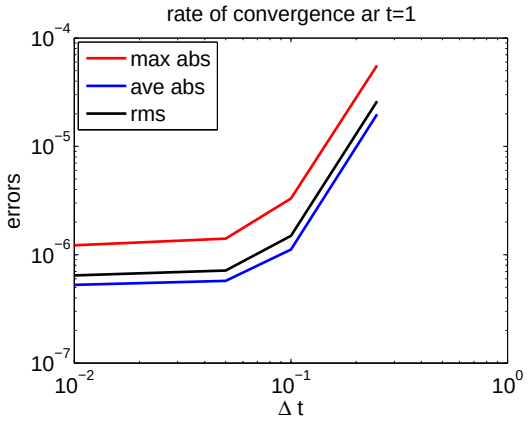

Figure 16: The rate of convergence of our method in Example 4.4 when time $t=1$ ($c=0.5$, $N=51^2$) is approximately 5.

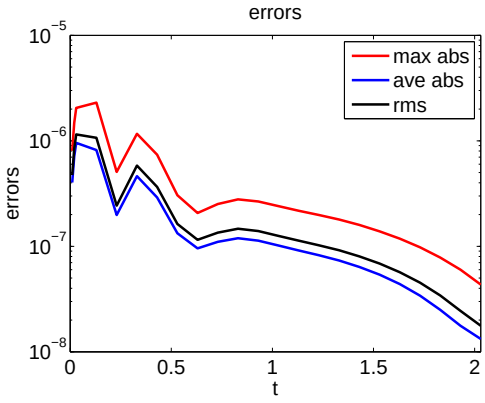

Figure 17: The maximum absolute errors, the average absolute errors and the root mean squared errors versus time in Example 4.4 with $c=1$, $N=31^2$, $n=6$, $h=0.1$, and $h_0=0.01$.

5 Conclusions

In this paper, the implicit time-stepping, the Houbolt method, and the modified localized method of approximated particular solutions (LMAPS) are applied to some linear or nonlinear diffusion-reaction-convection partial differential equations. The performance of the proposed scheme is examined on four examples, including two basis linear diffusion-reaction equations, one linear diffusion-reaction-convection equations and one nonlinear Burger's equation. The focuses of presented numerical discussions are high accuracy and fast rate of convergence.

This is uncommon to be seem due to computational trade offs. The simulation results show that the Houbolt method can significantly improve the numerical accuracy for

time-dependent problems when MLMAPS is used. Both time-discretization and spatial discretization are highly accurate so the numerical performance of the proposed scheme is extremely well. It is competitive to the MFS-DRM and Kansa's methods reported in [1] and the original LMAPS.

The proposed scheme has outstanding performance for solving convection-diffusion problem and nonlinear Burger's equation due to the following reasons:

- The numerical scheme is relatively stable with respect to time discretization for approximation of the solution at a large time. Thus, the numerical scheme is only sensitive when approximating the solution at beginning time. This depends on how small h_0 was chosen and its relation with h.

- The proposed numerical method is very flexible as it can solve large-scale problems due to the sparsity, it can deal with high-dimensional problems due to the kernels chosen, it can solve nonlinear or linear problems with high accuracy due to the use of low-order polynomial basis. Our on-going research focuses on theoretical analysis of the algorithms.

- Since our method is compared with other methods, for the sake of fairness, we use the same domains and same number of collocation points as in the paper being compared. In Example 4.4, as shown in Table 1, we tested the performance of this method for solving large-scale problem. The accuracy is further improved as number of points increases. This method can be easily used to solve large-scale problems without changing any parameter.

- Our method has same order of computational complexity as finite collocation method in [19] and LMAPS in [26]. The methods in [19] and [26] are based on MQ RBF whose shape parameter need to be carefully selected. The proposed MLMAPS is based on integrated polyharmonic splines with polynomial, which has no shape parameters to be determined. This advantage is especially manifested in solving large-scale problems, because we do not need to change any parameters when the distance between nodes is getting smaller. Moreover, our solutions are more consistent and stable as changes of time and other parameters in Example 4.3 and Example 4.4.

Acknowledgements

We thank editor and anonymous reviewers for their comments and suggestions that greatly improved the manuscript. This research was supported in part by funding from the Simons Foundation and the Centre de Recherches Mathématiques, through the Simons-CRM Scholar-in-Residence Program.

References

[1] SLIMANE ADJERID AND JOSEPH E FLAHERTY, *A local refinement finite-element method for two-dimensional parabolic systems,* SIAM J. Sci. Stat. Comput., 9(5) (1988), pp. 792–811.

[2] C. A. BUSTAMANTE, H. POWER, AND W. F. FLOREZ, *An efficient accurate local method of approximate particular solutions for solving convection–diffusion problems,* Engineering Analysis with Boundary Elements, 47 (2004), pp. 32–37.

[3] C. S. CHEN, C. M. FAN AND P. H. WEN, *The method of particular solutions for solving elliptic problems with variable coefficients*, Commun. Numer. Methods Eng., (2010).

[4] C. S. CHEN, C. M. FAN AND P. H. WEN, *The method of approximate particular solutions for solving certain partial differential equations,* Numer. Methods Partial Differential Equations, 28(2) (2012), pp. 506–522.

[5] C. S. CHEN, Y. C. HON AND R. A. SCHABACK, Scientific Computing with Radial Basis Functions, Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS, 39406, 2005.

[6] MEHDI DEHGHAN AND AKBAR MOHEBBI, *High-order compact boundary value method for the solution of unsteady convection–diffusion problems,* Math. Comput. Simulation, 79(3) (2008), pp. 683–699.

[7] CHIA-MING FAN, CHI-HUNG YANG AND WEI-SHIANG LAI, *Numerical solutions of two-dimensional flow fields by using the localized method of approximate particular solutions,* Engineering Analysis with Boundary Elements, 57 (2015), pp. 47–57.

[8] OLIVIER GOYON, *Multilevel schemes for solving unsteady equations,* Int. J. Numer. Methods Fluids, 22(10) (1996), pp. 937–959.

[9] JOHN C. HOUBOLT, *A recurrence matrix solution for the dynamic response of elastic aircraft,* J. Aeronautical Sci., 17 (1950), pp. 540–550.

[10] SU JINGBO, ZHU FENG, GENG YING AND NI XINGYE, *Numerical study of wave overtopping based on local method of approximate particular solution method,* Adv. Mech. Eng., 6 (2014), pp. 541–717.

[11] E. J. KANSA, *Multiquadrics–a scattered data approximation scheme with applications to computational fluid dynamics–I,* Comput. Math. Appl., 19(8/9) (1990), pp. 127–145.

[12] E. J. KANSA, *Multiquadrics–a scattered data approximation scheme with applications to computational fluid dynamics–II,* Comput. Math. Appl., 19(8/9) (1990), pp. 147–161.

[13] HANS PETTER LANGTANGEN AND SVEIN LINGE, Finite Difference Computing with PDEs: a Modern Software Approach, Volume 16, Springer, 2017.

[14] JICHUN LI, Y. C. HON AND C. S. CHEN, *Numerical comparisons of two meshless methods using radial basis functions,* Engineering Analysis with Boundary Elements, 26(3) (2002), pp. 205–225.

[15] MING LI, GHIZLANE AMAZZAR, AHMED NAJI AND C. S. CHEN,, *Solving biharmonic equation using the localized method of approximate particular solutions,* Int. J. Comput. Math., 91(8) (2014), pp. 1790–1801.

[16] WEN LI, GUOHUI SONG AND GUANGMING YAO, *Piece-wise moving least squares approximation,* Appl. Numer. Math., 115 (2017), pp. 68–81.

[17] C. Y. LIN, M. H. GU, D. L. YOUNG AND C. S. CHEN, *Localized method of approximate particular solutions with cole–Hopf transformation for multi-dimensional burgers equations,* Engineering Analysis with Boundary Elements, 40 (2014), pp. 78–92.

[18] JI LIN, YONGXING HONG, LEI-HSIN KUO AND CHEIN-SHAN LIU, *Numerical simulation of 3d nonlinear Schrödinger equations by using the localized method of approximate particular solutions,*

Engineering Analysis with Boundary Elements, 78 (2017), pp. 20–25.
[19] ARAM SOROUSHIAN AND JAMSHID FARJOODI, *A unified starting procedure for the houbolt method*, Commun. Numer. Methods Eng., 24(1) (2008), pp. 1–13.
[20] FARIBA TAKHTABNOOS AND AHMAD SHIRZADI, *A new implementation of the finite collocation method for time dependent PDEs,* Engineering Analysis with Boundary Elements, 63 (2016), pp. 114–124.
[21] GUANGMING YAO, Local Radial Basis Function Methods for Solving Partial Differential Equations, PhD thesis, The University of Southern Mississippi, 2010.
[22] GUANGMING YAO, *An improved localized method of approximate particular solutions for solving elliptic PDEs,* Comput. Math. Appl., 71(1) (2016), pp. 171–184.
[23] GUANGMING YAO, CHING-SHYANG CHEN AND HUI ZHENG, *A modified method of approximate particular solutions for solving linear and nonlinear PDEs,* Numerical Methods for Partial Differential Equations, 33 (2017), pp. 1839–1858.
[24] GUANGMING YAO, C. S. CHEN, WEN LI, AND D. L. YOUNG, *The localized method of approximated particular solutions for near-singular two-and three-dimensional problems,* Comput. Math. Appl., 70(12) (2015), pp. 2883–2894.
[25] D. L. YOUNG, M. H. GU AND C. M. FAN, *The time-marching method of fundamental solutions for wave equations,* Engineering Analysis with Boundary Elements, 33(12) (2009), pp. 1411–1425. Special Issue on the Method of Fundamental Solutions in Honour of Professor Michael Golberg.
[26] XUEYING ZHANG, MUYUAN CHEN, C. S. CHEN AND ZHIYONG LI, *Localized method of approximate particular solutions for solving unsteady navier-stokes problem,* Appl. Math. Model., 40(3) (2016), pp. 2265–2273.
[27] XUEYING ZHANG, HAIYAN TIAN AND WEN CHEN, *Local method of approximate particular solutions for two-dimensional unsteady burgers' equations,* Comput. Math. Appl., 66(12) (2014), pp. 2425–2432.