 CMES, vol.108, no.6, pp.375-396, 2015

Solving a Class of PDEs by a Local Reproducing Kernel Method with An Adaptive Residual Subsampling Technique

H. Rafieayan Zadeh[1], **M. Mohammadi**[1,2] **and E. Babolian**[1]

Abstract: A local reproducing kernel method based on spatial trial space spanned by the Newton basis functions in the native Hilbert space of the reproducing kernel is proposed. It is a truly meshless approach which uses the local sub clusters of domain nodes for approximation of the arbitrary field. It leads to a system of ordinary differential equations (ODEs) for the time-dependent partial differential equations (PDEs). An adaptive algorithm, so-called adaptive residual subsampling, is used to adjust nodes in order to remove oscillations which are caused by a sharp gradient. The method is applied for solving the Allen-Cahn and Burgers' equations. The numerical results show that the proposed method is efficient, accurate and be able to remove oscillations caused by sharp gradient.

Keywords: Local reproducing kernel method, method of lines, Newton basis functions, adaptive residual subsampling algorithm.

1 Introduction

During last decades, meshless methods are considered by many researchers. Unlike traditional numerical methods in solving PDEs, meshless methods [Belytschko, Krongauz, Organ, Fleming, and Krysl (1996)] need no mesh generation. Taking translates of kernels as trial functions in collocation methods, which are truly meshless, leads to a highly successful method [C. Franke (1998); Huang, Ren, and Russell (1990a); Huang, Ren, and Russell (1990b)]. Implementation of these methods is easy and allow good accuracy at low computational cost. In addition, it was proven recently [Schaback (2015)] that symmetric collocation using kernels is optimal along all linear PDE solvers using the same input data. There are plenty of papers which use kernel-based methods for solving various problems [Abbas-

[1] Faculty of Mathematical Sciences and Computer, Kharazmi University, 50 Taleghani Ave., Tehran 1561836314, Iran
[2] Corresponding author

bandy, Azarnavid, and Alhuthali (2014); Atluri and Shen (2003); Dong, Alotaibi, Mohiuddine, and Atluri (2014); Elgohary, Dong, Junkins, and Atluri (2014); Han and Atluri (2014); Hon and Schaback (2008); Mohammadi and Mokhtari (2014); Mohammadi and Mokhtari (2011); Mohammadi and Mokhtari (2013); Mohammadi, Mokhtari, and Panahipour (2013); Mohammadi, Mokhtari, and Panahipour (2014); Mohammadi, Mokhtari, and Schaback (2014); Mohammadi, Mokhtari, and Isfahani (2014); Mokhtari, Isfahani, and Mohammadi (2012)].

It is well known that representations of kernel-based approximants in terms of the standard basis of translated kernels are notoriously unstable. A more useful basis, so-called Newton basis, is offered in [Muller and Schaback (2009)]. The Newton basis turns out to be orthonomal in the reproducing native Hilbert space, and it is complete, if infinitely many data locations are reasonably chosen. A timesaving calculation of Newton basis arising from a pivoted Cholesky factorization has been introduced in [Pazouki and Schaback (2011)].

For the time-dependent PDEs, a spatial interpolation is applied by using expansion in terms of kernel functions. But since the coefficients are time-dependent, a system of ODEs is obtained. This is the well-known method of lines (MOL), and it turns out to be approximately useful in various cases. In this paper, we use expansion in terms of the Newton basis functions (NBFs) for constructing the spatial interpolation.

In global collocation methods, collocation matrix is made by considering the whole domain, since the obtained matrix will be full. This limits the applicability of those methods to solve large scale problems. But local methods [Chen, Ganesh, Golberg, and Cheng (2002); Lee, Liu, and Fan (2003); Mai-Duy and Tran-Cong (2002)] use overlapping sub-domains of the whole domain, thus a sparse matrix is obtained. A general idea behind the local reproducing kernel method is the use of the local sub clusters of domain nodes, named local influence domain, for approximation of the arbitrary field. With the selected influence domain, an approximation function is introduced as a sum of weighted NBFs. Then the collocation approach is used to determine weights. After the successful approximation function creation all the needed differential operators can be constructed by applying an arbitrary operator on the approximation function. The main advantage to using the local method is that less computer storage and flops are needed.

The kernel functions are defined by using a set of points called centers or trial points. The centers can set anywhere of given domain independently, therefore, center points can be moved or added or removed in the domain and this is the base of adaptive algorithms. Positions of the centers set influence on approximation quality and stability of the interpolation [Schaback (1995)], in addition, increasing number of centers causes large condition numeber of interpolation matrix. Adap-

tive algorithms are often applied for problems containing rapid variations in the given domain. Different kinds of adaptive methods have been developed in the literature, e.g. Greedy algorithm [Ling, Opfe, and Schaback (2006); Ling and Schaback (2009)], upwind technique [Lin and Atluri (2000); Lin and Atluri (2001)], r-adaptive mesh method (moving mesh strategy) [Huang, Ren, and Russell (1994)]. Four kinds of center choosing algorithms with some algorithmic analysis are introduced in [Gong, Wei, Wang, Feng, and Wang (2010)]. In this paper, we use adaptive residual subsampling method [Driscoll and Heryudono (2007)], where nodes can be added or removed based on interpolation residuals evaluated at a finer point set. In the adaptive residual subsampling algorithm, an interpolant has been computed for the center set, the residual of the resulting approximation is sampled on a finer node set. Nodes from the finer set are added to or removed from the set of centers based on the size of the residual of the interpolation at those points. The interpolant is then recomputed using the new center set for a new approximation.

In this study, we use a local reproducing kernel method based on spatial trial space spanned by the NBFs accompanied with an adaptive residual subsampling technique for the numerical solution of a class of PDEs including the Allen-Cahn and Burgers' equations with different kinds of initial and boundary conditions. The Allen-Cahn equation describes the motion of anti-phase boundaries in crystalline solids. It has been widely used in material science applications. The Burgers' equation is the simplest nonlinear model equation for diffusive waves in fluid dynamics. The Burgers' equation arises in many physical problems including one- dimensional turbulence, sound waves in a viscous medium, shock waves in a viscous medium, waves in fluid filled viscous elastic tubes, and magneto-hydrodynamic waves in a medium with finite electrical conductivity. The Burgers' equation is similar to the one dimensional Navier-Stokes equation without the stress term.

The rest of the paper is organized as follows. In section 2, the kernel-based trial functions and the NBFs are reviewed. The proposed local reproducing kernel method is given in section 3. The method is illustrated in section 4 by solving the nonlinear Allen-Cahn equation with mixed boundary conditions. In section 5, the adaptive residual subsampling algorithm is illustrated. Numerical experiments are given in section 6. The last section is devoted to a brief conclusion.

2 Kernel-based trial functions

Let Ω be a nonempty set. A function $K : \Omega \times \Omega \to \mathbb{R}$ is called a kernel on Ω. Let $K : \Omega \times \Omega \to \mathbb{R}$ be a symmetric positive definite kernel on Ω. This means that for all finite sets $\{x_i \in \Omega, i = 1, \ldots, n\}$ the kernel matrix $A = [K(x_i, x_j)]_{i,j=1,\ldots,n}$ is symmetric and positive definite. These kernels are *reproducing* in "native" Hilbert

space $\mathcal{N}_K = \overline{\text{span}\{K(x,\cdot)\ |x \in \Omega\}}$ of functions on Ω in the sense

$$\langle f, K(x,\cdot)\rangle_{\mathcal{N}_K} = f(x) \quad \text{for all } x \in \Omega, f \in \mathcal{N}_K.$$

The most important examples are the Whittle-Matern kernels $r^{m-d/2}\mathrm{K}_{m-d/2}(r)$, $r = \|x-y\|$, $x,y \in \mathbb{R}^d$, reproducing in the Sobolev space $W_2^m(\mathbb{R}^d)$ for $m > d/2$, where K_ν is the modiffed Bessel function of the second kind [Scahabck (2011)]. The following will be independent of the kernel chosen, but ones should be aware that the kernel should be smooth enough to allow suffciently many derivatives for the PDE and additional smoothness for fast convergence [Wendland (2005)]. For scattered nodes $\{x_i \in \overline{\Omega}, i = 1,\ldots,n\}$ the *translates* $K_j(x) = K(x,x_j)$ are the trial functions. The Newton basis functions (NBFs) $\{N_k(x)\}_{k=1}^n$ can be expressed by

$$N_k(x) = \sum_{j=1}^{n} K(x,x_j)c_{jk}, \quad k = 1,\ldots,n. \tag{1}$$

Considering $N(x) = [N_1(x)\cdots N_n(x)]$, $T(x) = [K(x,x_1)\cdots K(x,x_n)]$ and $C = [c_{jk}]_{j,k=1,\ldots,n}$, Eq. (1) can be written as

$$N(x) = T(x)\cdot C.$$

Subsequently, we have $N = A \cdot C$, where, $N = [N_j(x_i)]_{i,j=1,\ldots,n}$ and $A = [K(x_i, x_j)]_{i,j=1,\ldots,n}$. It has been proved [Pazouki and Schaback (2011)] that the Cholesky decomposition $A = L\cdot L^T$ with a nonsingular lower triangular matrix L leads to the Newton basis

$$N(x) = T(x)\cdot (L^T)^{-1}, \tag{2}$$

with

$$N = L, \quad C = (L^T)^{-1}. \tag{3}$$

Hence the condition number of the collocation matrix corresponding to the NBFs is smaller than the one corresponding to translated kernels. Consequently, using the NBFs for collocation will lead to more stable methods than using the basis of translates. Note that,

$$\mathscr{L}N(x) = \mathscr{L}T(x)\cdot (L^T)^{-1}, \tag{4}$$

where, $\mathscr{L}$ can be any linear operator like derivatives. Moreover, for computing the values of the Newton basis at the other points, for example $\{y_i \in \overline{\Omega}, i = 1,\ldots,n_y\}$,

(2) is used. The Newton basis commonly is constructed by positive definite kernels like radial basis functions (RBFs). The RBF is defined as

$$\phi_j(x) = K(x,x_j) = \phi(\|x-x_j\|_2),$$

where $\{x_j \in \Omega, j = 1,\ldots,n\}$ is a set of distinct points called centers. Some kinds of RBFs are given in Tab. 1. The RBFs may have a free parameter, called the shape parameter, denoted by ε. As the shape parameter changes, the shape of the RBFs changes, and subsequently the accuracy of interpolant and the condition number of the interpolation matrix will change. For interpolation of scattered data by RBFs, an uncertainty relation between the error and the condition of the interpolation matrix is proven. It states that the error and the condition number cannot both be kept small [Schaback (1995)] and there is a trade-off between the accuracy and the condition number. Note that the GA, IMQ and MS RBFs are examples of positive definite kernels and can be used for constructing the NBFs.

Table 1: Some kinds of RBFs

Gaussian (GA)	$\phi(r) = \mathrm{e}^{-(\varepsilon r)^2}$
Powers (P)	$\phi(r) = r^{\varepsilon}$
Multiquadric (MQ)	$\phi(r) = \sqrt{1+(\varepsilon r)^2}$
Inverse Multiquadric (IMQ)	$\phi(r) = 1/\sqrt{1+(\varepsilon r)^2}$
Thin plate splines (TPS)	$\phi(r) = r^{2\varepsilon}\log(r), \quad 2\varepsilon > 0$
Matern Sobolov (MS)	$\phi(r) = r^{\varepsilon}\mathrm{K}_{\varepsilon}(r)$, K_{ε} is the modified Bessel function of second kind

3 Local reproducing kernel method

Consider the following time-dependent PDE

$$\mathscr{L}u(x,t) = f(x,t), \quad x \in \Omega, \quad t \in [0,T], \tag{5}$$

with boundary condition

$$\mathscr{B}u(x,t) = g(x,t), \quad x \in \partial\Omega, \tag{6}$$

and initial condition

$$\mathscr{I}u(x,0) = u_0(x), \quad x \in \overline{\Omega}, \tag{7}$$

where $\mathscr{L} : H \to F$ is a differential operator, H and F are Hilbert spaces of functions on $\overline{\Omega}$, $\mathscr{B}$ is Dirichlet or Neumann or mixed boundary condition operator and $\mathscr{I}$ is

a linear operator. It is assumed that the problem (5)–(7) is well-posed. We choose discrete points $X = \{x_i \in \overline{\Omega}, i = 1,\ldots,n\}$ and a symmetric positive definite kernel $K : \Omega \times \Omega \to \mathbb{R}$. Let $X_I = \{x_i, i = 1,\ldots,ni\}$ be the interior points and $X_B = \{x_i, i = ni+1,\ldots,n\}$ be the boundary points, where ni is the number of interior points. For each $x_i \in X$, we consider a stencil $\Omega_i = \{x_k^i\}_{k=1}^m$ which contains the center x_i and its $m-1$ nearest neighboring points. In addition, $N_1^i,\ldots,N_m^i$ are considered as the NBFs corresponding to the stencil $\{x_k^i\}_{k=1}^m$. To approximate the solution $u(x,t)$, we consider kernel-based approximant in terms of the NBFs on the local domain Ω_i instead of the whole domain Ω. Then the approximate solution in the local domain Ω_i will be in the following form

$$u(x,t) = \sum_{k=1}^{m} \alpha_k^i(t)\, N_k^i(x). \tag{8}$$

Therefore, for $x_i \in \Omega_i$ we have

$$u(x_i,t) = \sum_{k=1}^{m} \alpha_k^i(t)\, N_k^i(x_i). \tag{9}$$

Consequently, (9) can be written as the following vector form

$$u(x_i,t) = N^i \cdot \alpha^i, \quad x_i \in \Omega_i, \tag{10}$$

where, $N^i = [N_1^i(x_i)\cdots N_m^i(x_i)]$ and $\alpha^i = [\alpha_1^i(t)\cdots\alpha_m^i(t)]^T$. By using (8) the linear system

$$U^i = \mathbf{N}^i \cdot \alpha^i$$

is obtained, where

$$U^i = [u(x_1^i,t)\cdots u(x_m^i,t)]^T,$$

and

$$\mathbf{N}^i = [N_k^i(x_p^i)]_{k,p=1,\ldots,m},$$

thus

$$\alpha^i = (\mathbf{N}^i)^{-1} \cdot U^i. \tag{11}$$

By substituting (11) in (10) we have

$$u(x_i,t) = N^i \cdot (\mathbf{N}^i)^{-1} \cdot U^i.$$

We now write the PDE (5) at a point $x_i \in X_I$ as follows

$$\mathscr{L}\left(N^i \cdot (\mathbf{N}^i)^{-1} \cdot U^i\right) = f(x_i,t), \quad i = 1,\ldots,ni. \tag{12}$$

The boundary condition (6) implies that

$$\mathscr{B}\left(N^i \cdot (\mathbf{N}^i)^{-1} \cdot U^i\right) = g(x_i,t), \quad i = ni+1,\ldots,n. \tag{13}$$

Moreover, (7) results

$$\mathscr{I}U(0) = U_0, \tag{14}$$

where $\mathscr{I}U(0) = [\mathscr{I}u(x_1,0)\cdots\mathscr{I}u(x_n,0)]^T$ and $U_0 = [u_0(x_1)\cdots u_0(x_n)]^T$. The equations (12)–(13) with the initial condition (14) lead to a system of ODEs at which the unknown vector $U = [u(x_1,t)\cdots u(x_n,t)]^T$ is to be determined. In order to obtain a matrix form for the ODE system, matrices of spatial derivatives must be calculated. For the spatial partial derivatives, we have

$$\frac{\partial^s}{\partial x^s}u(x_i,t) = N^i_{(s)} \cdot (\mathbf{N}^i)^{-1} \cdot U^i = C^i_{(s)} \cdot U^i,$$

where

$$N^i_{(s)} = [\frac{\partial^s}{\partial x^s}N^i_1(x_i)\cdots\frac{\partial^s}{\partial x^s}N^i_m(x_i)],$$

and

$$C^i_{(s)} = N^i_{(s)} \cdot (\mathbf{N}^i)^{-1}.$$

Now for constructing global derivative matrices from local contributions, we define global ni-by-n sparse matrices $D_{(s)}$, according to derivatives appeared in (5), of the form

$$D_{(s)}(i,I_i) = C^i_{(s)}, \quad i = 1,\ldots,ni,$$

where I_i is a vector that contains the indices of center x_i and it's $m-1$ nearest neighboring points. To clarify proposed method, it is applied for solving the Allen-Cahn equation in the next section.

4 Method validation

In this section, we apply the proposed method for solving the Allen-Cahn equation of the form

$$u_t - u(1-u^2) = \nu u_{xx}, \quad x \in [a,b],\ t \in [0,T], \tag{15}$$

with mixed boundary conditions,

$$\beta_1 u(a,t) + \gamma_1 \frac{\partial}{\partial x} u(a,t) = g_1(t), \quad t \in [0,T], \tag{16}$$

$$\beta_2 u(b,t) + \gamma_2 \frac{\partial}{\partial x} u(b,t) = g_2(t), \quad t \in [0,T], \tag{17}$$

and initial condition,

$$u(x,0) = u_0(x), \quad x \in [a,b], \tag{18}$$

where β_1, β_2, γ_1, and γ_2 are known parameters, $g_1(t)$ and $g_2(t)$ are known functions, and ν is the kinematics viscosity. Let $X = \{x_1, x_2, \ldots, x_{n-1}, x_n\}$ be discretization points in the interval $[a,b]$ where $x_1 = a$, and $x_n = b$. Based on the methodology and notation described in the previous section, we write the PDE (15) at a point $x_i, i = 2, \ldots, n-1$, as follows

$$u_t(x_i,t) - u(x_i,t)\left(1 - (u(x_i,t))^2\right) = \nu N^i_{(2)} \cdot \left(\mathbf{N}^i\right)^{-1} \cdot U^i, \quad i = 2, \ldots, n-1. \tag{19}$$

Then the equation (19) leads to the following matrix form

$$U_I' = U_I .* (\mathbf{1} - U_I .\wedge 2) + \nu D_{(2)} \cdot U, \tag{20}$$

where $.*$ and $.\wedge$ denote the pointwise product and power between two matrices or vectors,

$$D_{(2)}(i, I_{i+1}) = C^{i+1}_{(2)}, \quad i = 1, \ldots, n-2,$$
$$U_I = [u(x_2,t) \cdots u(x_{n-1},t)]^T,$$
$$U_I' = [u_t(x_2,t) \cdots u_t(x_{n-1},t)]^T,$$

and $\mathbf{1}$ is a $(n-2)$-by-1 vector with 1 in its entries. Now we implement the boundary conditions (16)–(17) at the points x_1 and x_n as follows

$$\beta_1 u(x_1,t) + \gamma_1 C^1_{(1)} \cdot U^1 = g_1(t), \tag{21}$$

$$\beta_2 u(x_n,t) + \gamma_2 C^n_{(1)} \cdot U^n = g_2(t). \tag{22}$$

Let the 2-by-n sparse matrix W be as follows:

$$W(1,I_1) = \beta_1\ (\mathbf{1}_1) + \gamma_1\ (C^1_{(1)}),$$
$$W(2,I_n) = \beta_2\ (\mathbf{1}_n) + \gamma_2\ (C^n_{(1)}),$$

where $\mathbf{1}_i$ is a $1 \times n$ vector with 1 in the ith entry and zero elsewhere. Then the Eqs. (21)–(22) lead to

$$W \cdot U = (g_i(t), i = 1,2)^T.$$

So the unknown vector $(u(x_i,t), i = 1,n)^T$ can be written in terms of the unknown vector U_I by solving the following equations:

$$W(:,[1,n]) \cdot (u(x_i,t), i = 1,n)^T = (g_i(t), i = 1,2)^T - W(:,2:n-1) \cdot U_I. \tag{23}$$

By subsituting (23) in (20), we get the system of ODEs with the initial conditions

$$U_I(0) = (u_0(x_i), 2 \le i \le n-1).$$

Note that the nonlinearity of the PDE is preserved, and a good ODE solver will automatically use a reasonable time-stepping and detect stiffness of the ODE system.

5 Adaptive residual subsampling algorithm

Since kernel-based methods are completely meshfree, some adaptive algorithms for finding optimal point sets may be devised. For example, in problems that exist rapid variations in given domain, such as steep gradients, corners, and topological changes resulting from nonlinearity, adaptive methods may be preferred over fixed grid methods. In order to achieve accuracy and stability, adaptive methods select optimal centers by moving, adding or removing points. In adaptive residual subsampling algorithm, some points may be added or removed by using computed residuals [Driscoll and Heryudono (2007)]. In this section, we describe adaptive residual subsampling algorithm for a local reproducing kernel method based on spatial trial space spanned by the NBFs.

Implementation of adaptive residual subsampling technique for time-dependent PDEs is to alternate time stepping with adaptation. First, initial centers $\{x_i, i = 1,\ldots,n\}$ are generated using n equally spaced points in the given domain. Then, using NBF interpolation at centers and initial condition of the PDE, unknown coefficients vector $\alpha = [\alpha_1 \cdots \alpha_n]^T$ is calculated from the linear system

$$N\alpha = U_0,$$

where, $N = [N_j(x_i)]_{i,j=1,\ldots,n}$ is the NBF matrix, $U_0 = [u_0(x_1)\cdots u_0(x_n)]^T$ and $u_0(x)$ is the initial condition function of the PDE. Now, the set $\{y_i = \frac{1}{2}(x_{i+1} - x_i), i = 1,\ldots,n-1\}$ is considered halfway between the centers. The residuals vector r is calculated by

$$r = |N^y \alpha - U_0^y|,$$

where, N^y is the NBF matrix for the points $\{y_i\}_{i=1}^{n-1}$, which is calculated by (2)–(3), $U_0^y = [u_0(y_1)\cdots u_0(y_{n-1})]^T$, and $r = [r_1 \cdots r_{n-1}]^T$. Points at which the residual exceeds a threshold θ_r are to become centers, and centers that lie between two points whose error is below a smaller threshold θ_c are removed. This means, if $r_i > \theta_r$, then y_i will be added to centers set, and if $r_i < \theta_c$ and $r_{i+1} < \theta_c$, then x_{i+1} will be removed. This process is called coarse–refine (coarse for removing centers and refine for adding new points). Therefore, a new centers set is given and the coarse–refine process is repeated while any new point can not be added. After ending coarse–refine processes, a new centers set is obtained. These new centers are used to advance the discrete solution up to a predetermined time $t = \tau$ by using the local method, which is described in the previous section. τ must be large enough to avoid excessive adaptation steps, while keeping it small enough that the adaptation can keep up with emerging or changing features in the solution. So solution is obtained at the time $t = \tau$. Now residual subsampling algorithm is applied by using the solution at this time level as a new initial state for further time. This process continues to achieve $t = T$.

6 Numerical results

In this section, we present the results of our scheme for the numerical solution of some equations. Functions with steep variation in the domain are often employed. The obtained results state the ability of the method for adapting centers to the regions with steep variations. Some results without using adaptive method are presented to show unstability in regions with rapid variations. In this work, we take the MS and IMQ RBFs for constructing the Newton basis, in addition, we take $\tau = 0.01$. In examples, $\nu = 1/Re$, where Re is the Reynold number and ν is the kinematics viscosity.

Example 1: Consider the Allen-Cahn equation [Driscoll and Heryudono (2007)]

$$u_t - u(1-u^2) = \nu u_{xx}, \quad x \in [-1,1],\ t \in [0,T], \tag{24}$$

with $Re = 10^6$, Dirichlet boundary condition,

$$u(\pm 1, t) = \pm 1, \tag{25}$$

and initial condition,

$$u(x,0) = 0.6x + 0.4\ \sin\left(\frac{\pi}{2}(x^2 - 3x - 1)\right). \tag{26}$$

As shown in Fig. 1, the solution of the equations (24)–(26), without using the adaptive residual subsampling method, have small oscillations in regions with steep gradients that are corrected by using the adaptive method. Adapting the nodes in steep gradients is shown in Fig. 2. In this example, we use the MS RBF with the shape parameter $\varepsilon = 1$, $\theta_r = 10^{-3}$ and $\theta_c = 10^{-9}$. In addition, the number of nodes starts with 30 and finally grows to 176 at $T = 10$.

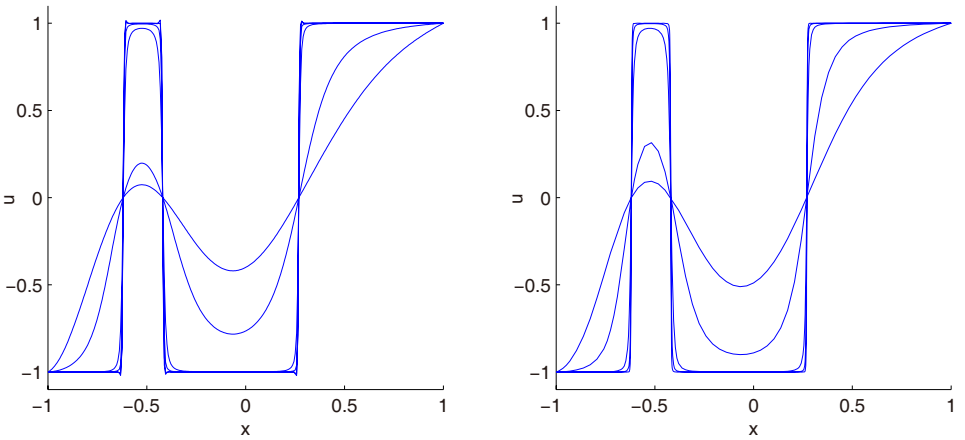


Figure 1: Solution of the Allen-Cahn equation, (left) without the adaptive method, (right) with the adaptive method.

Example 2: Consider the moving front problem given by the Burgers' equation [Driscoll and Heryudono (2007)]

$$u_t = -uu_x + \nu u_{xx}, \quad x \in [-1,1],\ t \in [0,T], \tag{27}$$

with $Re = 1000$, Dirichlet boundary condition,

$$u(0,t) = u(1,t) = 0, \tag{28}$$

and initial condition,

$$u(x,0) = \sin(2\pi x) + \frac{1}{2}\sin(\pi x). \tag{29}$$

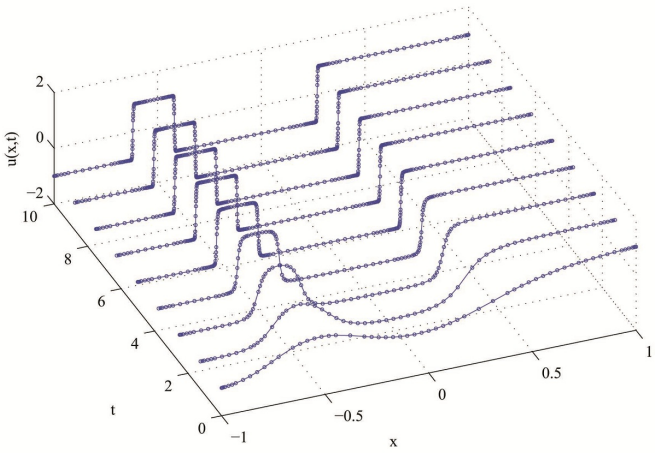


Figure 2: Solution of the Allen-Cahn equation with the adaptive method. Number of nodes increases in region with steep gradients.

As the previous example, the solution of the equations (27)–(29) generate steep front and consequently have unstabilities in steep front. Correction of these unstabilities and adaption of nodes by using the adaptive method are shown in Figures 3 and 4 respectively. Here, we employed the MS RBF with the shape parameter $\varepsilon = 1$, $\theta_r = 10^{-4}$ and $\theta_c = 10^{-8}$. Furthermore, number of nodes at $T = 1$ grows to 76.

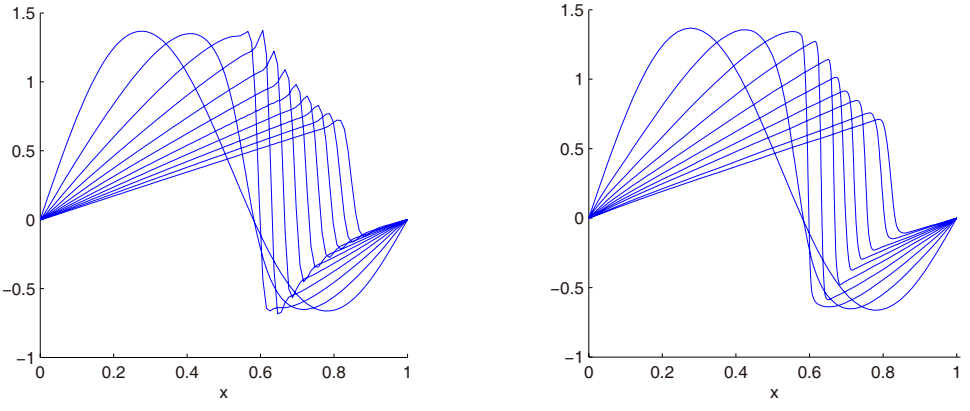


Figure 3: Solution of the equations (27)–(29), (left) whitout the adaptive method, (right) with the adaptive method.

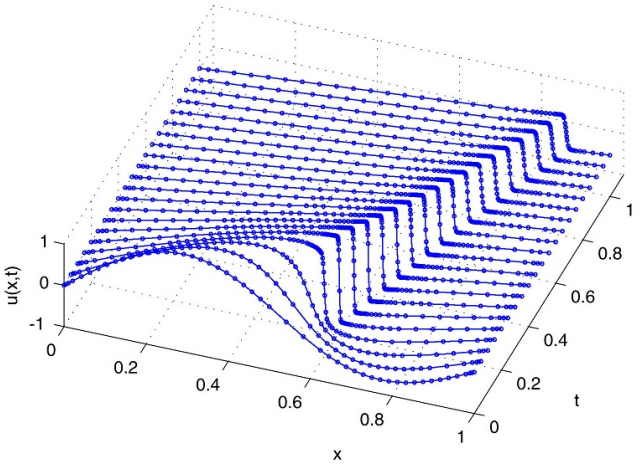


Figure 4: Solution of the equations (27)–(29); Adaption of nodes.

Table 2: Numerical results of the equations (30)–(32) for $Re = 0.1, 10, 100$.

	$Re = 0.1$	$(T = 0.02)$	$Re = 10$	$(T = 1)$	$Re = 100$	$(T = 1)$
x	Exact	Approx	Exact	Approx	Exact	Approx
0.1	0.0428	0.0428	0.0663	0.0663	0.0754	0.0754
0.2	0.0815	0.0815	0.1312	0.1312	0.1506	0.1508
0.3	0.1122	0.1122	0.1928	0.1927	0.2257	0.2259
0.4	0.1320	0.1320	0.2480	0.2480	0.3003	0.3006
0.5	0.1389	0.1389	0.2919	0.2919	0.3744	0.3748
0.6	0.1322	0.1322	0.3161	0.3160	0.4478	0.4482
0.7	0.1125	0.1125	0.3081	0.3081	0.5203	0.5207
0.8	0.0818	0.0818	0.2537	0.2537	0.5915	0.5919
0.9	0.0430	0.0430	0.1461	0.1460	0.6600	0.6600
θ_r		10^{-4}		10^{-4}		10^{-2}
θ_c		10^{-9}		10^{-9}		10^{-9}

Example 3: Consider the Burgers' equation [Hon and Mao (1998)]

$$u_t = -uu_x + \nu u_{xx}, \quad x \in [-1,1],\ t \in [0,T], \tag{30}$$

with Dirichlet boundary condition,

$$u(0,t) = u(1,t) = 0, \tag{31}$$

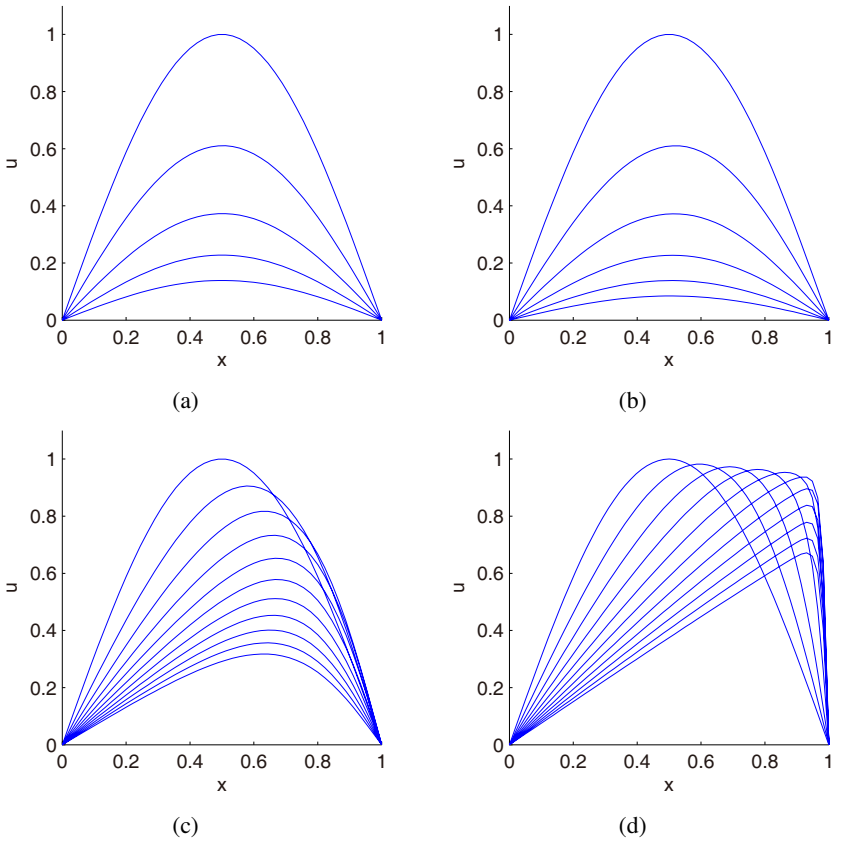


Figure 5: Solution of the equations (30)–(32), (a) $Re = 0.1$ for $T = 0.02$, (b) $Re = 1$ for $T = 0.25$, (c) $Re = 10$ for $T = 1$, (d) $Re = 100$ for $T = 1$.

and initial condition,

$$u(x,0) = \sin(\pi x). \tag{32}$$

The exact solution of equations (30)–(32) is given by [Caldwell and Smith (1982)]

$$u(x,t) = \frac{4\pi\nu \sum_{n=1}^{\infty} n \mathrm{I}_n(1/2\pi\nu)\, \sin(n\pi x)\, \mathrm{e}^{(-n^2\nu\pi^2 t)}}{\mathrm{I}_0(1/2\pi\nu) + 2\sum_{n=1}^{\infty} \mathrm{I}_n(1/2\pi\nu)\, \cos(n\pi x)\, \mathrm{e}^{(-n^2\nu\pi^2 t)}} \quad ,$$

where I_n denotes the modiffed Bessel function of order n. In this example, we use the IMQ RBF with the shape parameters $\varepsilon = 5, 5, 5, 5.8$ for $Re = 0.1, 1, 10, 100$ respectively. The numerical solutions are shwon in Fig. 5. Furthermore, for $Re = 0.1, 10, 100$, comparison with the exact solutions and the thersholds are represented in Tab. 2.

For $Re = 1000, 10000$, the solution of equations (30)–(32) generates steep gradient toward $x = 1$, subsequently the approximate solution has oscillations in regions with steep gradient. For $Re = 1000$, by implementing the proposed method with the MS RBF with the shape parameter $\varepsilon = 1.5$, without using the adaptive algorithm, the oscillations are gradually disappeared when the number of nodes increases (Fig. 6). The adaptive residual subsampling method starts with 30 nodes and gradually grows to maximum of 53 nodes, and consequently obtains solution at $T = 1$ with 50 nodes (Fig. 7).

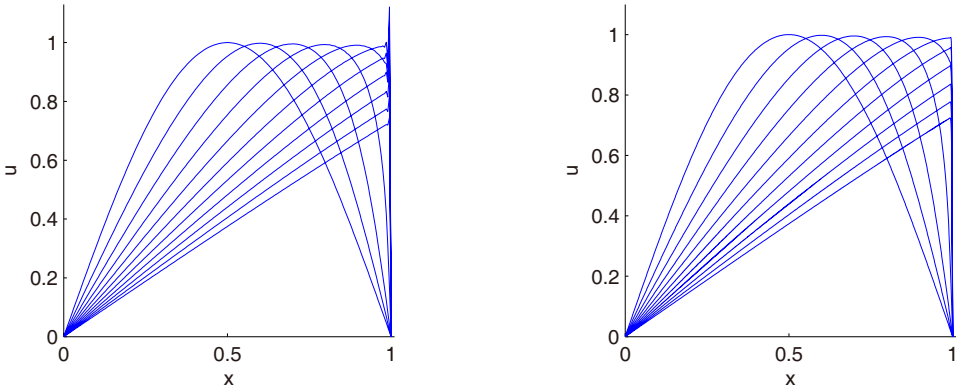


Figure 6: Solution of the equations (30)–(32), for $Re = 1000$ whit MS RBF, (left) 200 nodes, (right) 1000 nodes.

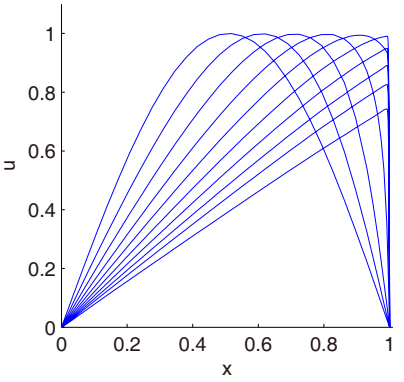


Figure 7: Solution of the equations (30)–(32) by using the adaptive method with 50 nodes at $T = 1$.

For $Re = 10000$, the oscillations do not vanish even if the number of nodes be very large. Fig. 8 shows the oscillations with 1000 nodes and 2000 nodes in this case. As

shown in Fig. 9, by using the adaptive residual subsampling algorithm and MS RBF with $\varepsilon = 1$, the oscillations are disappeared at $T = 1$ with 108 nodes (Fig. 9). For $Re = 10000$, the numerical solution has been compared with the Christie accurate solution given in [Hon and Mao (1998)]. Absolute error graph of implementation of the proposed method for $Re = 10000$ is shown in Fig. 9. In addition, adaption of nodes are shown in Fig. 10.

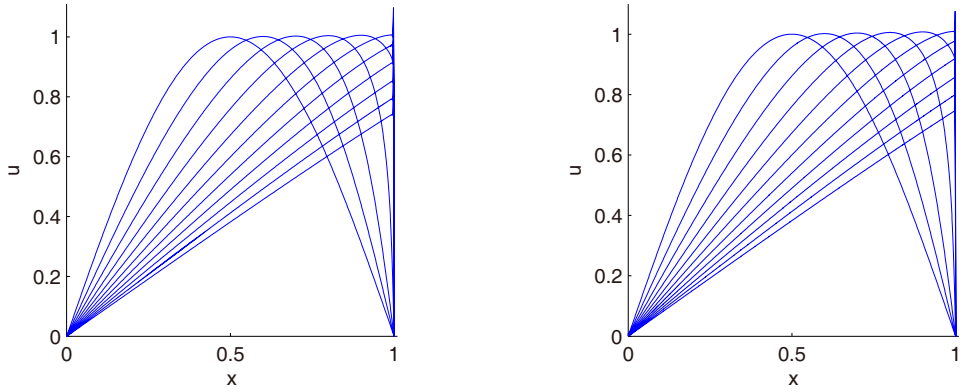


Figure 8: Solution of the equations (30)–(32), for $Re = 10000$ with MS RBF, (left) 1000 nodes, (right) 2000 nodes.

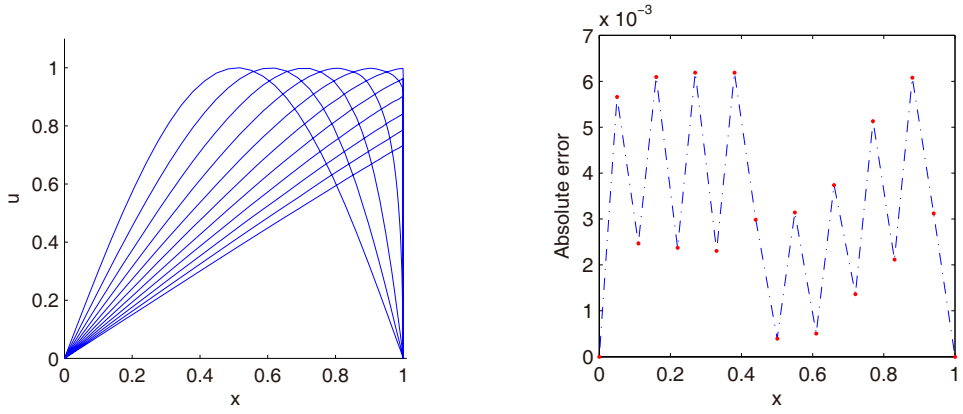


Figure 9: (left) Solution of the equations (30)–(32) for $Re = 10000$ with the adaptive method, (right) Absolute error graph at $x = 0, 0.05, 0.11, 0.16, 0.22, 0.27, 0.33, 0.38, 0.44, 0.50, 0.55, 0.61, 0.66, 0.72, 0.77, 0.83, 0.88, 0.94, 1$.

Example 4: Consider the Burgers' equation [Pugh (1995)]

$$u_t = -uu_x + \nu u_{xx} + f(x,t), \quad x \in [0,1],\ t \in [0,T], \tag{33}$$

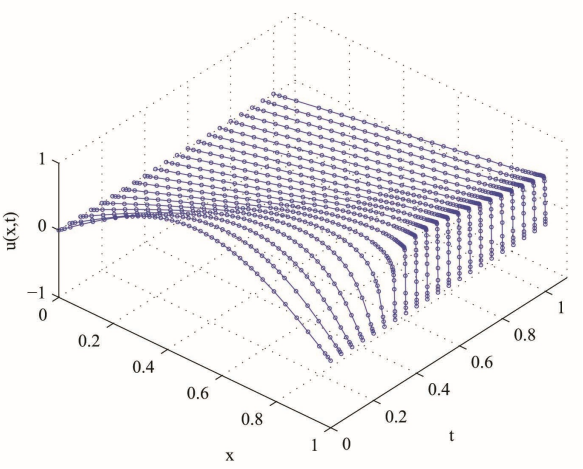


Figure 10: Solution of the equations (30)–(32); Adaption of nodes.

with Neumann boundary condition,

$$u_x(0,t) = u_x(1,t) = 0, \tag{34}$$

initial condition,

$$u(x,0) = \frac{1}{4}\cos(\pi x), \tag{35}$$

and

$$f(x,t) = -\frac{1}{4}\mathrm{e}^{-\nu t}\cos(\pi x)\left(\nu + \frac{\pi}{4}\mathrm{e}^{-\nu t}\sin(\pi x) - \nu\pi^2\right), \tag{36}$$

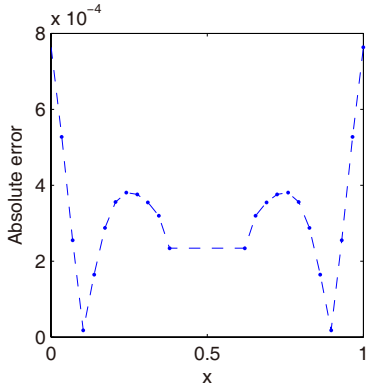
and the exact solution

$$u(x,t) = \frac{1}{4}\mathrm{e}^{-\nu t}\cos(\pi x).$$

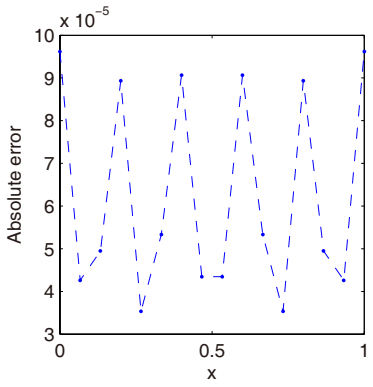
The equations (33)–(35) are solved for $Re = 60, 120, 240, 1000, 10000$ by the proposed method with the MS RBF with the shape parameter $\varepsilon = 1.5$. The absolute error graphs at $T = 0.5$ are also shown in Fig. 11.

7 Conclusion

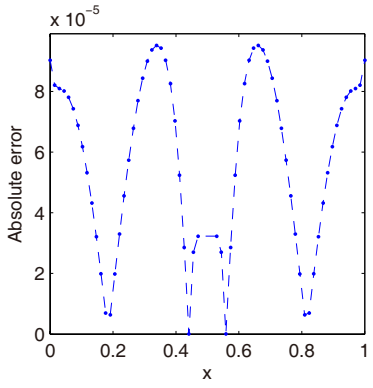
In this paper, a local reproducing kernel method based on spatial trial spaces spanned by the Newton basis functions was used for solving some time-dependent



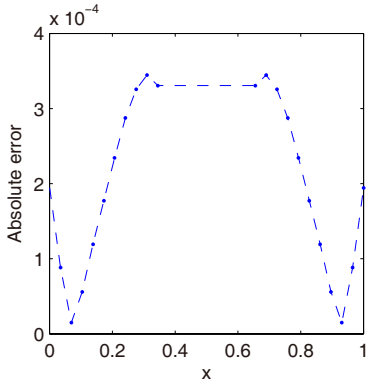
(a) Re = 60



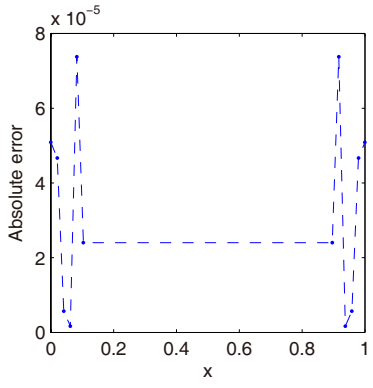
(b) Re = 120



(c) Re = 240



(d) Re = 1000



(e) Re = 10000

Figure 11: Absolute error graphs for solution of the equations (33)–(35) at $T = 0.5$.

PDEs. In addition, the adaptive residual subsampling algorithm was employed to correct oscillations. Numerical results show that the method works properly for problems with oscillatory behaviour due to steep gradients. It seems that this method can be applied for solving higher dimensional time dependent PDEs. We leave this to our further work.

References

Abbasbandy, S.; Azarnavid, B.; Alhuthali, M. (2014): A shooting reproducing kernel hilbert space method for multiple solutions of nonlinear boundary value problems. *J. Comput. Appl. Math.* http://dx.doi.org/10.1016/j.cam.2014.11.014.

Atluri, S.; Shen, S. (2003): *The Meshless Local Petrov-Galerkin (MLPG) Method*. Tech. Science Press.

Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M.; Krysl, P. (1996): Meshless methods: an overview and recent developments. *Comput. Methods Appl. Mech., Engrg.*, vol. 139, pp. 3–47.

C. Franke, R. S. (1998): Solving partial differential equations by collocation using radial basis functions. *Appl. Math. Comput.*, vol. 93, pp. 73–82.

Caldwell, J.; Smith, P. (1982): Solution of burgers' equation with a large reynolds number. *Appl. Math. Modelling.*, vol. 6, pp. 381–385.

Chen, C.; Ganesh, M.; Golberg, M.; Cheng, A. (2002): Multilevel compact radial basis functions based computational scheme for some elliptic problems. *Comput. Math. Appl*, vol. 43, pp. 359–378.

Dong, L.; Alotaibi, A.; Mohiuddine, S.; Atluri, S. N. (2014): Computational methods in engineering: A variety of primal & mixed methods, with global & local interpolations, for well-posed or ill-posed bcs. *CMES Comput. Model. Eng. Sci.*, vol. 99, pp. 1–85.

Driscoll, T.; Heryudono, A. (2007): Adaptive residual subsampling methods for radial basis function interpolation and collocation problems. *Comput. Math. Appl*, vol. 53, pp. 927–939.

Elgohary, T.; Dong, L.; Junkins, J.; Atluri, S. N. (2014): Time domain inverse problems in nonlinear systems using collocation & radial basis functions. *CMES Comput. Model. Eng. Sci.*, vol. 100, pp. 59–84.

Gong, D.; Wei, C.; Wang, L.; Feng, L.; Wang, L. (2010): adaptive methods for center choosing of radial basis function interpolation: a review. *Lecture Notes in Comput. Sci.*, vol. 6377, pp. 573–580.

Han, Z.; Atluri, S. (2014): On the (meshless local petrov-galerkin) mlpg-eshelby method in computational finite deformation solid mechanics - part ii. *CMES Comput. Model. Eng. Sci.*, vol. 97, pp. 199–237.

Hon, Y.; Mao, X. (1998): An efficient numerical scheme for burgers' equation. *Appl. Math. Comput.*, vol. 95, pp. 37–50.

Hon, Y.; Schaback, R. (2008): Solvability of partial differential equations by meshless kernel methods. *Adv. Comput. Math.*, vol. 28, pp. 283–299.

Huang, W.; Ren, Y.; Russell, R. D. (1990): Multiquadrics–a scattered data approximation scheme with applications to computational fluid dynamics i: surface approximations and partial derivative estimates. *Computers Math. Appl.*, vol. 19, pp. 127–145.

Huang, W.; Ren, Y.; Russell, R. D. (1990): Multiquadrics–a scattered data approximation scheme with applications to computational fluid-dynamicsii: solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers Math. Appl.*, vol. 19, pp. 147–161.

Huang, W.; Ren, Y.; Russell, R. D. (1994): Moving mesh partial differential equations (mmpdes) based on the equidistribution principle. *SIAM J. Numer. Anal.*, vol. 31, pp. 709–730.

Lee, C.; Liu, X.; Fan, S. (2003): Local multiquadric approximation for solving boundary value problems. *Comput. Mech.*, vol. 30, pp. 396–409.

Lin, H.; Atluri, S. (2000): Meshless local petrov-galerkin (mlpg) method for convection-diffusion problems. *CMES Comput. Model. Eng. Sci.*, vol. 1, pp. 45–60.

Lin, H.; Atluri, S. (2001): The meshless local petrov-galerkin (mlpg) method for solving incompressible navier-stokes equations. *CMES Comput. Model. Eng. Sci.*, vol. 2, pp. 117–142.

Ling, L.; Opfe, R.; Schaback, R. (2006): Results on meshless collocation techniques. *Eng. Anal. Bound. Elem.*, vol. 30, pp. 247–253.

Ling, L.; Schaback, R. (2009): An improved subspace selection algorithm for meshless collocation methods. *Int. J. Numer. Meth. Engng.*, vol. 80, pp. 1623–1639.

Mai-Duy, N.; Tran-Cong, T. (2002): Mesh-free radial basis function network methods with domain decomposition for approximation of functions and numerical solution of poisson equation. *Eng. Anal. Bound. Elem.*, vol. 26, pp. 133–156.

Mohammadi, M.; Mokhtari, R. (2011): Solving the generalized regularized long wave equation on the basis of a reproducing kernel space. *J. Comput. Appl. Math.*, vol. 235, pp. 4003–4014.

Mohammadi, M.; Mokhtari, R. (2013): A new algorithm for solving nonlinear schrödinger equation in the reproducing kernel space. *Iranian J. Sci. Tech.*, vol. 37, pp. 523–546.

Mohammadi, M.; Mokhtari, R. (2014): A reproducing kernel method for solving a class of nonlinear systems of pdes. *Math. Model. Anal.*, vol. 19, pp. 180–198.

Mohammadi, M.; Mokhtari, R.; Isfahani, F. T. (2014): Solving an inverse problem for a parabolic equation with a nonlocal boundary condition in the reproducing kernel space. *Iranian J. Numer. Anal. Optimization.*, vol. 4, pp. 57–76.

Mohammadi, M.; Mokhtari, R.; Panahipour, H. (2013): A galerkin-reproducing kernel method: application to the 2d nonlinear coupled burgers' equations. *Eng. Anal. Bound. Elem.*, vol. 37, pp. 1642–1652.

Mohammadi, M.; Mokhtari, R.; Panahipour, H. (2014): Solving two parabolic inverse problems with a nonlocal boundary condition in the reproducing kernel space. *Appl. Comput. Math.*, vol. 13, pp. 91–106.

Mohammadi, M.; Mokhtari, R.; Schaback, R. (2014): A meshless method for solving the 2d brusselator reaction-diffusion system. *CMES: Comput. Model. Eng. Sci.*, vol. 101, pp. 113–138.

Mokhtari, R.; Isfahani, F. T.; Mohammadi, M. (2012): Reproducing kernel method for solving nonlinear differential-difference equations. *Abstr. Appl. Anal.*, vol. 2012, pp. 1–10.

Muller, S.; Schaback, R. (2009): A newton basis for kernel spaces. *J. Approx. Theory.*, vol. 161, pp. 645–655.

Pazouki, M.; Schaback, R. (2011): Bases for kernel-based spaces. *J. Comput. Appl. Math.*, vol. 236, pp. 575–588.

Pugh, S. (1995): Finite element approximations of burgers' equation.

Scahabck, R. (2011): *Kernel-based meshless methods*. Lecture Note, Gottingen. http://num.math.unigoettingen.de/schaback/teaching/AV2.pdf.

Schaback, R. (1995): Error estimates and condition numbers for radial basis function interpolation. *Adv. Comput. Math.*, vol. 3, pp. 251–264.

Schaback, R. (2015): A computational tool for comparing all linear pde solvers. *Adv. Comput. Math.*, vol. 41, pp. 333–355.

Wendland, H. (2005): *Scattered Data Approximation*. Cambridge University Press.