

Article

# Variable Shape Parameter Strategy in Local Radial Basis Functions Collocation Method for Solving the 2D Nonlinear Coupled Burgers' Equations

Hananeh Nojavan, Saeid Abbasbandy \*  and Tofiqh Allahviranloo

Department of Mathematics, Science and Research Branch, Islamic Azad University, Tehran 14778, Iran; nojavan.hananeh@gmail.com (H.N.); allahviranloo@yahoo.com (T.A.)

\* Correspondence: abbasbandy@yahoo.com; Tel.: +98-(912)-130-5326

Academic Editor: Athanasios Tzavaras

Received: 31 May 2017; Accepted: 12 July 2017; Published: 21 July 2017

**Abstract:** This study aimed at investigating a local radial basis function collocation method (LRBFCM) in the reproducing kernel Hilbert space. This method was, in fact, a meshless one which applied the local sub-clusters of domain nodes for the approximation of the arbitrary field. For time-dependent partial differential equations (PDEs), it would be changed to a system of ordinary differential equations (ODEs). Here, we intended to decrease the error through utilizing variable shape parameter (VSP) strategies. This method was an appropriate way to solve the two-dimensional nonlinear coupled Burgers' equations comprised of Dirichlet and mixed boundary conditions. Numerical examples indicated that the variable shape parameter strategies were more efficient than constant ones for various values of the Reynolds number.

**Keywords:** local meshless method; variable shape parameter (VSP); reproducing kernel space; 2D nonlinear coupled Burgers' equations

## 1. Introduction

Contrary to conventional numerical methods in solving the partial differential equations (PDEs), meshless methods [1], it is not essential to utilize meshes. Collocation methods are, in fact, meshless and easy to program. In addition, they allow some kinds of approaches to solve the PDEs. Considering the translation of kernels as trial functions, meshless collocation in asymmetric and symmetric forms is described in [2–4]. It is highly successful, since the arising linear systems are easy to produce, leading to such good accuracy with the beneficial range of computational expenses. Furthermore, it has been newly proven [5] that the symmetric collocation [2,3] utilizing kernel basis is optimal along all linear PDE solvers using the same input data. By following this utilization of kernels, we can solve the PDEs. An overview of kernel methods prior to the year 2006 is presented in [6], while their recent variations are in [7–12] and the related references.

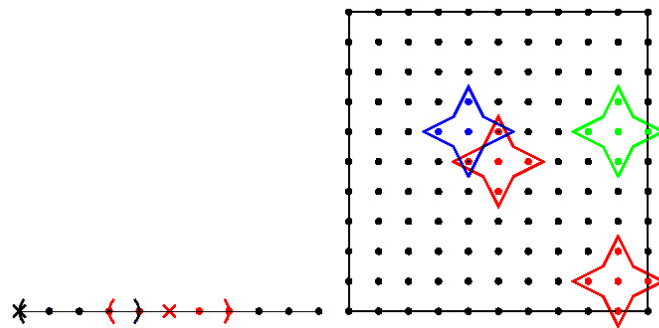
The selection of a shape parameter in a radial or kernel basis function has been a challengeable topic for some decades (see [13] and references over there). There are many experimental observations on the treatment of kernel-based methods under scaled (shape) parameter. In addition, there are optimization methods attempting to provide harmony between bad conditions and small errors.

There are some strategies in the literature to select shape parameters. Variable shape parameter (VSP) strategies can be compared with constant shape parameter (CSP) strategies. Many mathematicians utilize the CSPs in the radial basis function (RBF) approximations [14,15] due to their easy analysis in comparison with the VSPs; however, there are many findings from a broad applications set [16–18], indicating the advantages of using the VSPs. A number of strategies to choose a VSP would be investigated in this work.

Meshless kernel-based approaches are on the basis of a fixed spatial interpolation for time-dependent PDEs. By the means of method of lines, they can be converted to a system of the ordinary differential equations (ODEs) in time [19,20].

In general, the global collocation methods consider the whole domain. Although this method is simple to implement, the obtained collocation matrix is ill-conditioned, especially for large-scale problems.

Therefore, various localized meshless methods have been recommended in the literature to solve this problem (see [21,22] and the references therein). A main idea behind the local RBF collocation method is the utilization of the local sub-clusters of domain nodes (see Figure 1), called local domains of influence, with the local RBFs for the approximation of fields. In other words, for the approximation of function in any nodes of domain, we consider the local sub-clusters of domain nodes containing that node and specified number of nearest neighboring nodes of domain nodes. With the chosen influence domain, an approximation function is considered as a sum of weighted local RBFs with VSP. After that, the collocation approach is utilized to determine weights. In the following, all essential differential operators can be made via any operator on the approximation function. The most important benefit of utilizing the local method would be that the overlapping influence domain leads to many small matrices for every center node, rather than a large collocation matrix. As a result, sparse global derivative matrices would be achieved. Therefore, less computer storage and flops are required by this method. In principle, it is probable to use both uniform and random nodal arrangements in the method implementation owing to the approach meshless feature, but the accuracy wise efficiency of the uniform nodal arrangement is better than random nodes. One of the advantages of using uniform nodal arrangement is that small spatial derivatives and collocation matrices (of the size of sub-domain that is  $5 \times 5$  in the current case) corresponding to every stencil require to be computed just once. This stores a considerable amount of the CPU time and also memory.



**Figure 1.** The uniform node arrangement and the schematics of the local domains of influence in the interior, near boundary and corner points using  $ns = 5$ , 1D (left), and 2D (right), where  $\times$  denotes the center node in 1D.

In this work, by considering the 2D nonlinear coupled Burgers' equations, we try to show the validation of the proposed method. Indeed, we consider the Dirichlet and mixed boundary conditions. The accuracy, stability and efficiency are considered for the high Reynolds number,  $Re$ . We can find the applications and some numerical methods for this equation in [23–29]. In addition, the numerical investigation of the two-dimensional coupled Burgers' equations can be found in [30]. The 2D nonlinear coupled Burgers' equations are considered by a implicit finite-difference scheme in [31], the element-free characteristic method in [32], the variational multiscale element-free Galerkin method in [33], the Chebyshev pseudospectral method in [34], the global RBF method in [24], and the local RBF collocation method in [35].

The rest of the paper is organized as follows. In Section 2, a useful summary of the kernel-based trial functions is provided. In Section 3, the recommended local reproducing kernel method is given. In Section 4, the model is solved numerically and conclusions are provided in the last section.

## 2. Kernel-Based Trial Functions

**Definition 1.** Let  $\Gamma \subset \mathbb{R}^d$  be an arbitrary nonempty set. A function  $K : \Gamma \times \Gamma \rightarrow \mathbb{R}$  is called (real) kernel on  $\Gamma$ .

**Definition 2.** Let  $H$  be a real Hilbert space of function  $f : \Omega \rightarrow \mathbb{R}$ . A function  $K : \Omega \times \Omega \rightarrow \mathbb{R}$  is called reproducing kernel for  $H$  if [36] :

- (1)  $K(x, \cdot) \in H$  for all  $x \in \Omega$ ,
- (2)  $\langle f, K(x, \cdot) \rangle_H = f(x)$  for all  $x \in \Omega, f \in H$ .

**Definition 3.** A kernel  $K$  is symmetric, if  $K(x, y) = K(y, x)$  holds for all  $x, y \in \Gamma$ .

**Definition 4.** A kernel  $K$  is called semi-positive definite, if:

$$\sum_{i,j=1}^n \epsilon_i \epsilon_j K(x_i, x_j) \geq 0, \quad (1)$$

for any finite set of points  $X = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$  and any real numbers  $\epsilon_1, \dots, \epsilon_n$ . Furthermore, the function  $K$  is called positive definite on  $\Gamma$  if the quadratic form (1) is zero only for  $X = 0$ .

We consider a smooth symmetric positive definite kernel  $K : \Gamma \times \Gamma \rightarrow \mathbb{R}$  on the spatial domain  $\Gamma$ . With each kernel, there is a reproducing “native” Hilbert space:

$$\mathcal{N}_K = \overline{\text{span}\{K(x, \cdot) \mid x \in \Gamma\}}$$

of functions defined on  $\Gamma$  in the sense:

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

where the inner product is related to the property of the kernel by:

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

For scattered nodes  $x_1, \dots, x_n \in \mathbb{R}^d$ , the translates  $K_j(x) = K(x_j, x)$  are the trial functions, and we intend to begin our work with them. Owing to the smoothness and explicitness of the kernel  $K$ , take derivatives with respect to both arguments. Therefore, we can achieve cheap derivatives of the  $K_j$ . If the kernel is translation-invariant on  $\mathbb{R}^d$ , we have:

$$K(x, y) = \phi(x - y), \quad \forall x, y \in \mathbb{R}^d.$$

One of the most important kernels with significant properties is the radial kernel [37]. The radial kernel can be defined as:

$$K(x, y) = \phi(r), \quad r = \|x - y\|_2, \quad x, y \in \mathbb{R}^d;$$

for a scalar function:

$$\phi : [0, \infty) \rightarrow \mathbb{R},$$

the function  $\phi$  is called a RBF.

The most important examples are the Whittle–Matern kernels  $r^\nu K_\nu(r)$ , where  $\nu = m - d/2$ ,  $r = \|x - y\|_2$ ,  $x, y \in \mathbb{R}^d$ , reproducing in the Sobolev space  $W_2^m(\mathbb{R}^d)$  for  $m > d/2$ , and  $K_\nu$  is the modified Bessel function of the second kind [38].

### 2.1. Constant Scaled Kernels

Kernels on  $\mathbb{R}^d$  can be scaled by a positive factor  $c$  by examining the new kernel:

$$K_c(x, y) = K\left(\frac{x}{c}, \frac{y}{c}\right) \quad \forall x, y \in \mathbb{R}^d.$$

Moreover, the constant scaled translation-invariant and radial kernels on  $\mathbb{R}^d$  can be defined as:

$$K_c(x, y) = \phi\left(\frac{x - y}{c}\right) \quad \forall x, y \in \mathbb{R}^d,$$

$$K_c(x, y) = \phi\left(\frac{r}{c}\right), \quad r = \|x - y\|_2, \quad x, y \in \mathbb{R}^d,$$

respectively. Large  $c$  increases the condition number of kernel matrices, and small  $c$  indicates sharp peaks, leading to approximating functions imperfectly. The selection of shape parameter  $c$  is regarded as a problem existing for over two decades (see [13,17,39] and the references therein), playing a crucial role in finding the numerical solution of the PDEs (see [40]).

### 2.2. Variable Shape Parameter

In many cases, it has been mentioned that VSP strategies would generate more valid results in comparison with CSP. A negative outcome of utilizing a variable shape is that the system matrix is not symmetric anymore. Up to now, the choice of the optimal value of the shape parameter remains an open question; no mathematical theory has been developed up to now to determine its optimal value.

If the kernel  $K$  is radial, i.e.,  $K(x, y) = \phi(r^2)$ , then the variably scaled radial kernels on  $\mathbb{R}^d$  can be defined as:

$$K_c(x, x_j) = \phi\left(\left(\frac{r_j}{c_j}\right)^2\right), \quad r_j = \|x - x_j\|_2, \quad x, x_j \in \mathbb{R}^d,$$

where  $c_j$  is the shape parameter corresponding to the  $j$ th center. For example, the Whittle–Matern kernel with VSP is given by:

$$\left(\left(\frac{r_j}{c_j}\right)^2\right)^\nu K_\nu\left(\left(\frac{r_j}{c_j}\right)^2\right), \quad r_j = \|x - x_j\|_2, \quad x, x_j \in \mathbb{R}^d.$$

In this paper, we have proposed some strategies to choose the VSP:

- strategy 1 :  $c_j = (c_{\min} + (\frac{c_{\max} - c_{\min}}{n-1})(j-1))^{1/2}$  for  $j = 1, 2, \dots, n$ ,
- strategy 2 :  $c_j = (c_{\min} + (\frac{c_{\max} - c_{\min}}{n-1})(j-1))^{-1/4}$  for  $j = 1, 2, \dots, n$ ,
- strategy 3 :  $c_j = (c_{\min}^2 (\frac{c_{\max}^2}{c_{\min}^2})^{\frac{(j-1)}{(n-1)}})^{1/2}$  for  $j = 1, 2, \dots, n$ ,
- strategy 4 :  $c_j = (c_{\min} + (c_{\max} - c_{\min}) \exp(-j))^{-1}$  for  $j = 1, 2, \dots, n$ ,
- strategy 5 :  $c_j = c_{\min} + (c_{\max} - c_{\min}) \sin(j)$  for  $j = 1, 2, \dots, n$ .

Note that,  $c_{\min}$  and  $c_{\max}$  are positive parameters and denote the minimum and maximum of  $c_j$ , respectively.

In [41,42], the linear VSP is in the following form :

$$c_j = c_{\min} + \left(\frac{c_{\max} - c_{\min}}{n-1}\right)j \quad \text{for } j = 1, 2, \dots, n.$$

Here, we modified the linear shape parameter formula, and considered  $c_j$  as a new VSP by strategies 1 and 2. Strategies 3 and 5 are known as exponential and trigonometric VSPs, respectively [41,43].

In this work, new VSP  $c_j$  is proposed in strategy 4.

The approximation function via radial kernel can be written as follows:

$$f(x) = \sum_{j=1}^n \lambda_j K_c(x, x_j), \quad (2)$$

for a set of distinct nodal points  $\{x_1, x_2, \dots, x_n\} \subseteq \Gamma \subset \mathbb{R}^d$  and  $f_1, f_2, \dots, f_n$  such that  $f_i \in \mathbb{R}$ ,  $i = 1, 2, \dots, n$ . Then, the unknown parameter  $\lambda_j$ , which is independent of  $r_j$ , can be found by putting each node  $x_i$ ,  $i = 1, 2, \dots, n$  into Equation (2) and solving the following system of linear algebraic equations:

$$A\Lambda = F,$$

where:

$$A = (K_c(x_i, x_j))_{1 \leq i, j \leq n}, \quad \Lambda = (\lambda_i, 1 \leq i \leq n)^T, \quad F = (f_i, 1 \leq i \leq n)^T.$$

### 3. Numerical Method

In this section, the proposed local reproducing kernel method is introduced. Considering the time-dependent PDE:

$$\mathcal{L}u(x, t) = f(x, t), \quad x \in \Gamma, t \in (0, T], \quad (3)$$

with the initial condition:

$$\mathcal{I}u(x, 0) = u_0(x), \quad x \in \bar{\Gamma}, \quad (4)$$

and Dirichlet or Neumann boundary conditions:

$$u(x, t) = g^D(x, t), \quad x \in D \subset \partial\Gamma, t \in [0, T], \quad (5)$$

$$\frac{\partial u}{\partial n}(x, t) = g^N(x, t), \quad x \in \mathcal{N} \subset \partial\Gamma, t \in [0, T]. \quad (6)$$

Suppose  $\mathcal{L}$  is a differential operator,  $\mathcal{I}$  is a linear operator,  $u \in H$ ,  $f \in F$ , where  $H$  and  $F$  are Hilbert spaces of functions on  $\bar{\Gamma}$  and we assume that the problems (3)–(6) are well-posed. Let us consider the discretization points  $x_i$ ,  $1 \leq i \leq n$ , and a symmetric positive definite kernel  $K : \Gamma \times \Gamma \rightarrow \mathbb{R}$ . Let us reorder the points successively into points  $\{X_I \cup X_D \cup X_N\}$ , where  $X_I = \{x_1, \dots, x_{z_1}\}$  is the set of interior points,  $X_D = \{x_{z_1+1}, \dots, x_{z_1+z_2}\}$  is the set of Dirichlet boundary points, and  $X_N = \{x_{z_1+z_2+1}, \dots, x_n\}$  is the set of Neumann boundary points,  $z_1$  and  $z_2$  are the number of interior and Dirichlet boundary points. For each  $x_i$  ( $i = 1, \dots, n$ ), we consider a stencil  $S_i = \{x_k^{[i]}\}_{k=1}^{n_s}$ , which contains the center  $x_i$  and its  $n_s - 1$  nearest neighboring points and forms the radial basis  $N_1^{[i]}, N_2^{[i]}, \dots, N_{n_s}^{[i]}$  corresponding to these points. To approximate the solution  $u(x, t)$  over  $S_i$ , we use a linear combination as follows:

$$u^{[i]}(x, t) = \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) N_j^{[i]}(x), \quad x \in S_i. \quad (7)$$

In matrix form, we have:

$$U^{[i]} = N^{[i]} * \alpha^{[i]}(t),$$

which gives:

$$\alpha^{[i]}(t) = N^{-1[i]} * U^{[i]}, \quad (8)$$

where:

$$\begin{aligned} U^{[i]} &= \left( u^{[i]}(x_1^{[i]}, t), u^{[i]}(x_2^{[i]}, t), \dots, u^{[i]}(x_{n_s}^{[i]}, t) \right)^T, \\ \alpha^{[i]} &= \left( \alpha_1^{[i]}(t), \alpha_2^{[i]}(t), \dots, \alpha_{n_s}^{[i]}(t) \right)^T, \\ N^{[i]} &= \left( N_j^{[i]}(x_k^{[i]}) \right)_{1 \leq k \leq n_s, 1 \leq j \leq n_s}. \end{aligned}$$

The derivative of the approximate solution can also be approximated at the center locations by applying a linear differential operator  $L$  to the local interpolation (7). Depending on the problem at hand,  $L$  will be either a single derivative operator or a linear combination of derivative operators. The equation:

$$L(u^{[i]}(x, t)) = \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) L(N_j^{[i]}(x)), \quad x \in S_i,$$

gives:

$$L(u^{[i]}(x^{[i]}, t)) = L(N^{[i]}) * N^{-1[i]} * U^{[i]},$$

where:

$$L(N^{[i]}) = \left( L(N_j^{[i]}(x^{[i]})) \right)_{1 \leq j \leq n_s}.$$

Now, we write the PDE (3) at a point  $x_i, i = 1, \dots, z_1$  as follows:

$$\mathcal{L}u(x_i, t) = f(x_i, t),$$

then:

$$\mathcal{L}u^{[i]}(x_i, t) = f(x_i, t).$$

Thus:

$$\mathcal{L}\left(\sum_{j=1}^{n_s} \alpha_j^{[i]}(t) N_j^{[i]}(x_i)\right) = f(x_i, t), \quad i = 1, \dots, z_1. \quad (9)$$

Now, let:

$$\begin{aligned} U &= (U_I, U_D, U_N)^T, \\ U_I &= (u(x_i, t), 1 \leq i \leq z_1), \\ U_D &= (u(x_i, t), z_1 + 1 \leq i \leq z_1 + z_2), \\ U_N &= (u(x_i, t), z_1 + z_2 + 1 \leq i \leq n). \end{aligned}$$

By the Dirichlet boundary (5), we have:

$$U_D = (g^D(x_i, t), z_1 + 1 \leq i \leq z_1 + z_2), \quad (10)$$

and by the Neumann boundary (6):

$$\sum_{j=1}^{n_s} \alpha_j^{[i]}(t) \frac{\partial N_j^{[i]}}{\partial \mathbf{n}}(x_i) = g^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n.$$

Hence:

$$N_{\mathbf{n}}^{[i]} * N^{-1[i]} * U^{[i]} = g^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n, \quad (11)$$

in which:

$$N_{\mathbf{n}}^{[i]} = \left( \frac{\partial N_j^{[i]}}{\partial \mathbf{n}}(x_i), j = 1, \dots, n_s \right).$$

Now, let  $I_i$  be a vector that contains the indices of center  $x_i$  and its  $n_s - 1$  nearest neighboring points.

We consider the  $(n - z_1 - z_2) \times n$  sparse matrix  $W$  as follows:

$$W(i - z_1 - z_2, I_i) = N_{\mathbf{n}}^{[i]} * N^{-1[i]}, \quad i = z_1 + z_2 + 1, \dots, n.$$

Therefore, Equation (11) leads to:

$$W * U = (g^{\mathcal{N}}(x_i, t), z_1 + z_2 + 1 \leq i \leq n)^T.$$

The unknown vector  $U_{\mathcal{N}}$  will be considered in terms of the unknown vector  $U_I$  by solving the following equations:

$$\begin{aligned} & W(:, z_1 + z_2 + 1 : n) * U_{\mathcal{N}} \\ &= (g^{\mathcal{N}}(x_i, t), z_1 + z_2 + 1 \leq i \leq n)^T - W(:, 1 : z_1) * U_I - W(:, z_1 + 1 : z_1 + z_2) * U_D. \end{aligned} \quad (12)$$

Furthermore, the initial condition (4) leads to:

$$\sum_{j=1}^{n_s} \alpha_j^{[i]}(0) \mathcal{I}(N_j^{[i]}(x_i)) = u_0(x_i), \quad i = 1, \dots, z_1.$$

Let  $D$  be an  $z_1 \times n$  sparse matrix as follows:

$$D(i, I_i) = \mathcal{I}(N^{[i]}) * N^{-1[i]}, \quad i = 1, \dots, z_1,$$

where:

$$\mathcal{I}(N^{[i]}) = \left( \mathcal{I}(N_j^{[i]}(x_i)) \right)_{1 \leq j \leq n_s}.$$

Then, we have:

$$\begin{aligned} & D(:, 1 : z_1) * U_I(0) \\ &= (u_0(x_i), 1 \leq i \leq z_1)^T - D(:, z_1 + 1 : z_1 + z_2) * U_D(0) - D(:, z_1 + z_2 + 1 : n) * U_{\mathcal{N}}(0). \end{aligned} \quad (13)$$

By considering Equations (8), (10) and (12), Equation (9) leads to the following ODEs:

$$\tilde{\mathcal{L}}(U_I(t)) = \mathcal{F}(U_I(t)),$$

with the initial conditions (13), where  $\tilde{\mathcal{L}}$  is a linear differential operator and  $\mathcal{F}$  is an operator, which may have some global sparse matrices from local contribution.

#### 4. Validation of the Method

Let us consider the following system:

$$\begin{cases} U_t(X, t) = -U(X, t)U_x(X, t) - V(X, t)U_y(X, t) + \frac{1}{Re}\Delta U(X, t), \\ V_t(X, t) = -V(X, t)V_y(X, t) - U(X, t)V_x(X, t) + \frac{1}{Re}\Delta V(X, t) \end{cases} \quad X \in \Gamma \subset \mathbb{R}^2, \quad t \in (0, T], \quad (14)$$

$$\begin{cases} (U(X, t), V(X, t)) = (f^D(X, t), g^D(X, t)) \\ (\frac{\partial U}{\partial n}(X, t), \frac{\partial V}{\partial n}(X, t)) = (f^N(X, t), g^N(X, t)) \end{cases} \quad \begin{matrix} X \in D \subseteq \partial\Gamma, & t \in [0, T], \\ X \in \mathcal{N} \subseteq \partial\Gamma, & t \in [0, T], \end{matrix} \quad (15)$$

$$\begin{cases} U(X, 0) = U_0(X), \\ V(X, 0) = V_0(X), \end{cases} \quad (16)$$

where  $X = (x, y) \in \bar{\Gamma}$ ,  $Re$  is the Reynolds number,  $V_0, U_0, f^D, g^D, f^N$ , and  $g^N$  are known functions,  $\Gamma \subset \mathbb{R}^2$  is the domain set,  $\partial\Gamma$  is the boundary of the domain set  $\Gamma$ ,  $\Delta$  is the Laplace operator,  $U$  and  $V$  are unknown functions. We consider the set of points  $\{X_I \cup X_D \cup X_N\}$  for discretization equations, where  $X_I = \{X_1, \dots, X_{z_1}\}$  is the set of interior points,  $X_D = \{X_{z_1+1}, \dots, X_{z_1+z_2}\}$  is the set of Dirichlet boundary points, and  $X_N = \{X_{z_1+z_2+1}, \dots, X_n\}$  is the set of Neumann boundary points,  $z_1$  and  $z_2$  are the number of interior and Dirichlet boundary points. For each  $X_i$  ( $i = 1, \dots, n$ ), we consider a stencil  $S_i = \{X_k^{[i]}\}_{k=1}^{n_s}$ , which contains the center  $X_i$  and its  $n_s - 1$  nearest neighboring points and form the radial basis  $N_1^{[i]}, N_2^{[i]}, \dots, N_{n_s}^{[i]}$  corresponding to these points. Let us consider the symmetric positive definite kernel  $K : \Gamma \times \Gamma \rightarrow \mathbb{R}$ , and try to find the functions  $U(\cdot, t) | [0, T] \rightarrow N_K$ , and  $V(\cdot, t) | [0, T] \rightarrow N_K$ , where:

$$\begin{aligned} U^{[i]}(X, t) &= \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) N_j^{[i]}(X), & X \in S_i, \\ V^{[i]}(X, t) &= \sum_{j=1}^{n_s} \beta_j^{[i]}(t) N_j^{[i]}(X), & X \in S_i. \end{aligned}$$

Here,  $N_j(X)$  are the RBFs corresponding to the kernel  $K$ , which is reproducing in the native Hilbert space  $N_K$ . Thus, we have:

$$\begin{cases} \alpha^{[i]}(t) = \mathbf{N}^{-1[i]} * \mathbf{U}^{[i]}, \\ \beta^{[i]}(t) = \mathbf{N}^{-1[i]} * \mathbf{V}^{[i]}, \end{cases} \quad (17)$$



where:

$$\begin{aligned}
 \alpha^{[i]}(t) &= (\alpha_j^{[i]}(t)), 1 \leq j \leq n_s)^T, \\
 \beta^{[i]}(t) &= (\beta_j^{[i]}(t)), 1 \leq j \leq n_s)^T, \\
 \mathbf{N}^{[i]} &= (N_j^{[i]}(X_k^{[i]}))_{1 \leq k \leq n_s, 1 \leq j \leq n_s}', \\
 \mathbf{U}^{[i]} &= (\mathbf{U}_I, \mathbf{U}_D, \mathbf{U}_N)^T, \\
 \mathbf{V}^{[i]} &= (\mathbf{V}_I, \mathbf{V}_D, \mathbf{V}_N)^T, \\
 \mathbf{U}_I &= (U(X_j, t), 1 \leq j \leq z_1), \\
 \mathbf{U}_D &= (U(X_j, t), z_1 + 1 \leq j \leq z_1 + z_2), \\
 \mathbf{U}_N &= (U(X_j, t), z_1 + z_2 + 1 \leq j \leq n), \\
 \mathbf{V}_I &= (V(X_j, t), 1 \leq j \leq z_1), \\
 \mathbf{V}_D &= (V(X_j, t), z_1 + 1 \leq j \leq z_1 + z_2), \\
 \mathbf{V}_N &= (V(X_j, t), z_1 + z_2 + 1 \leq j \leq n).
 \end{aligned}$$

We now write the PDE (14) at the interior points  $X_i$  ( $i = 1, \dots, n_i$ ) as follows:

$$\begin{aligned}
 U_t^{[i]}(X_i, t) = -U^{[i]}(X_i, t) \left( \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) \frac{\partial N_j^{[i]}}{\partial x}(X_i) \right) - V^{[i]}(X_i, t) \left( \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) \frac{\partial N_j^{[i]}}{\partial y}(X_i) \right) \\
 + \frac{1}{\text{Re}} \sum_{j=1}^{n_s} \alpha_j^{[i]}(t) \Delta N_j^{[i]}(X_i), \quad (18)
 \end{aligned}$$

$$\begin{aligned}
 V_t^{[i]}(X_i, t) = -V^{[i]}(X_i, t) \left( \sum_{j=1}^{n_s} \beta_j^{[i]}(t) \frac{\partial N_j^{[i]}}{\partial y}(X_i) \right) - U^{[i]}(X_i, t) \left( \sum_{j=1}^{n_s} \beta_j^{[i]}(t) \frac{\partial N_j^{[i]}}{\partial x}(X_i) \right) \\
 + \frac{1}{\text{Re}} \sum_{j=1}^{n_s} \beta_j^{[i]}(t) \Delta N_j^{[i]}(X_i). \quad (19)
 \end{aligned}$$

With the aid of Label (17), we have:

$$\begin{cases} \mathbf{U}_t^{[i]} = -\mathbf{U}^{[i]} * (\mathbf{N}_x^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{U}^{[i]}) - \mathbf{V}^{[i]} * (\mathbf{N}_y^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{U}^{[i]}) + \frac{1}{\text{Re}} (\Delta \mathbf{N}^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{U}^{[i]}), \\ \mathbf{V}_t^{[i]} = -\mathbf{V}^{[i]} * (\mathbf{N}_y^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{V}^{[i]}) - \mathbf{U}^{[i]} * (\mathbf{N}_x^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{V}^{[i]}) + \frac{1}{\text{Re}} (\Delta \mathbf{N}^{[i]} * \mathbf{N}^{-1[i]} * \mathbf{V}^{[i]}), \end{cases} \quad (20)$$

where:

$$\begin{aligned}
 \mathbf{N}_x^{[i]} &= \left( \frac{\partial N_j^{[i]}}{\partial x}(x_i), j = 1, \dots, n_s \right), \\
 \mathbf{N}_y^{[i]} &= \left( \frac{\partial N_j^{[i]}}{\partial y}(x_i), j = 1, \dots, n_s \right).
 \end{aligned}$$

Let  $I_i$  be a vector that contains the indices of center  $X_i$  and its  $n_s - 1$  nearest neighboring points. We consider the  $z_1 \times n$  sparse matrices  $D_1$ ,  $D_2$  and  $D_3$  as follows:

$$\begin{aligned}D_1(i, I_i) &= N_x^{[i]} * N^{-1[i]}, \\D_2(i, I_i) &= N_y^{[i]} * N^{-1[i]}, \\D_3(i, I_i) &= \Delta N^{[i]} * N^{-1[i]}.\end{aligned}$$

Then, Equation (14) leads to:

$$\begin{cases} \tilde{\mathbf{U}}_t = -\tilde{\mathbf{U}} * (D_1 * \mathbf{U}) - \tilde{\mathbf{V}} * (D_2 * \mathbf{U}) + \frac{1}{\text{Re}} D_3 * \mathbf{U}, \\ \tilde{\mathbf{V}}_t = -\tilde{\mathbf{V}} * (D_2 * \mathbf{V}) - \tilde{\mathbf{U}} * (D_1 * \mathbf{V}) + \frac{1}{\text{Re}} D_3 * \mathbf{V}, \end{cases} \quad (21)$$

where:

$$\begin{aligned}\mathbf{U} &= (U(X_1, t), \dots, U(X_n, t))^T, \\ \tilde{\mathbf{U}} &= (U(X_1, t), \dots, U(X_{z_1}, t))^T, \\ \mathbf{V} &= (V(X_1, t), \dots, V(X_n, t))^T, \\ \tilde{\mathbf{V}} &= (V(X_1, t), \dots, V(X_{z_1}, t))^T,\end{aligned}$$

and  $*$  denotes the pointwise product between two matrices or vectors.

The Dirichlet boundary conditions imply that:

$$\begin{cases} \mathbf{U}_D = (f^D(X_j, t)), \quad z_1 + 1 \leq j \leq z_1 + z_2, \\ \mathbf{V}_D = (g^D(X_j, t)), \quad z_1 + 1 \leq j \leq z_1 + z_2. \end{cases} \quad (22)$$

With the Neumann boundary conditions, we have:

$$\begin{aligned}\sum_{j=1}^{n_s} \alpha_j(t) \frac{\partial N_j}{\partial n}(x_i) &= f^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n, \\ \sum_{j=1}^{n_s} \beta_j(t) \frac{\partial N_j}{\partial n}(x_i) &= g^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n,\end{aligned}$$

where:

$$\begin{aligned}\frac{\partial \mathbf{N}}{\partial n} * \mathbf{N}^{-1} * \mathbf{U} &= (f^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n), \\ \frac{\partial \mathbf{N}}{\partial n} * \mathbf{N}^{-1} * \mathbf{V} &= (g^{\mathcal{N}}(x_i, t), \quad z_1 + z_2 + 1 \leq i \leq n),\end{aligned}$$

and:

$$\frac{\partial \mathbf{N}}{\partial n} = \left( \frac{\partial N_j}{\partial n}(X_i) \right)_{z_1+z_2+1 \leq i \leq n, 1 \leq j \leq n}.$$

Suppose:

$$\mathbf{W} = \frac{\partial \mathbf{N}}{\partial n} * \mathbf{N}^{-1},$$

hence the vector  $\mathbf{U}_{\mathcal{N}}$  and  $\mathbf{V}_{\mathcal{N}}$  are as:

$$\begin{cases} \mathbf{W}(:, z_1 + z_2 + 1 : n) * \mathbf{U}_{\mathcal{N}} \\ = (f^{\mathcal{N}}(X_i, t), z_1 + z_2 + 1 \leq i \leq n)^T - \mathbf{W}(:, 1 : z_1) * \mathbf{U}_I - \mathbf{W}(:, z_1 + 1 : z_1 + z_2) * \mathbf{U}_D, \\ \mathbf{W}(:, z_1 + z_2 + 1 : n) * \mathbf{V}_{\mathcal{N}} \\ = (g^{\mathcal{N}}(X_i, t), z_1 + z_2 + 1 \leq i \leq n)^T - \mathbf{W}(:, 1 : z_1) * \mathbf{V}_I - \mathbf{W}(:, z_1 + 1 : z_1 + z_2) * \mathbf{V}_D. \end{cases} \quad (23)$$

By substituting Labels (22) and (23) in (21), we obtain the system of the ODEs with the initial conditions:

$$\begin{cases} \tilde{\mathbf{U}}_I(0) = \tilde{\mathbf{U}}_{I0}, \\ \tilde{\mathbf{V}}_I(0) = \tilde{\mathbf{V}}_{I0}, \end{cases}$$

where:

$$\begin{aligned} \tilde{\mathbf{U}}_{I0} &= (U_0(X_j), 1 \leq j \leq z_1)^T, \\ \tilde{\mathbf{V}}_{I0} &= (V_0(X_j), 1 \leq j \leq z_1)^T. \end{aligned}$$

## 5. Numerical Results

The results of our scheme for the numerical solution of two problems with Dirichlet and mixed boundary conditions have been presented. We take the Matern kernel due to strong convergence rate with the RBF parameter  $\nu = m - d/2 = 2$ , and RBF scale  $c$ , i.e., we work with the kernel:

$$K(x, y) = \left( \frac{\|x - y\|_2}{c} \right)^2 K_2 \left( \frac{\|x - y\|_2}{c} \right),$$

which is reproducing in the Hilbert space  $W_2^3(\mathbb{R}^2)$ .

In addition, we consider  $c_{min} = 1$  and  $c_{max} = 10$  in variable shape strategies for all experiments in examples.

In the implementation of this technique, we have also used Legendre points:

$$x_i = -1 + \left( \frac{2(i-1)}{n-1} \right), \quad i = 1, \dots, n,$$

belonging to the interval  $[-1, 1]$  that can be easily transferred to the interval  $[a, b]$  by the transformation  $y = \frac{b-a}{2}x + \frac{a+b}{2}$ , and uniform points:

$$x_i = a + (i-1)h, \quad h = \frac{b-a}{n-1}, \quad i = 1, \dots, n.$$

The ODE solver ode113 of MATLAB (R2014b, MathWorks, Natick, MA, USA) is used to solve the final ODE system (21). The accuracy of the numerical results, is measured by the maximum absolute error defined as :

$$L_{\infty} = \max_{1 \leq j \leq n} |u_j - \tilde{u}|,$$

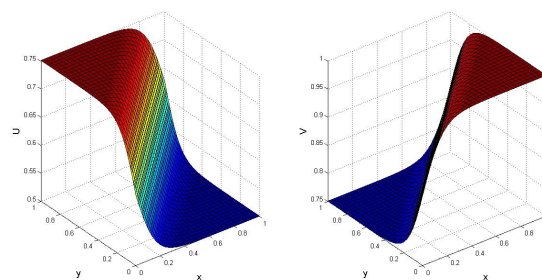
where  $u$  and  $\tilde{u}$  represent the exact and approximate solutions, respectively.

**Example 1.** Take problems (14)–(16). The initial and Dirichlet boundary conditions can be chosen by considering the following solutions, which are stated in [35]:

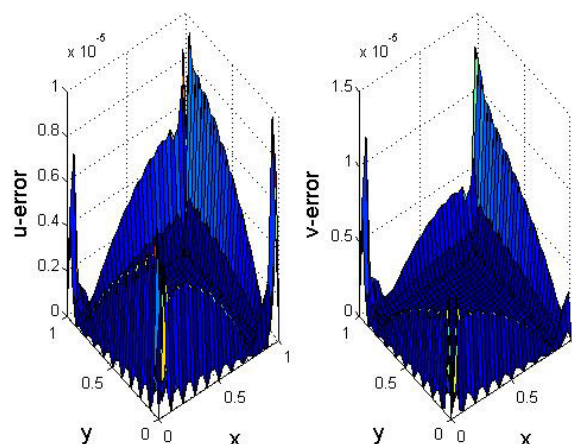
$$U(X, t) = \frac{3}{4} - \frac{1}{4 \left( 1 + \exp((-4x + 4y - t)(\frac{Re}{32})) \right)},$$

$$V(X, t) = \frac{3}{4} + \frac{1}{4 \left( 1 + \exp((-4x + 4y - t)(\frac{Re}{32})) \right)}.$$

Large values of  $Re$  lead to sharp gradients in the solutions as shown in Figure 2. The solutions of singularly perturbed problems exhibit sharp boundary or /and interior layers, where the solutions have a steep gradient. Analytical and numerical aspects of singularly perturbed problems are considered by Das et al. [44]. Figures 3 and 4 show absolute error distributions at time  $T = 2$  for  $Re = 10$  with strategy 3 for different number of points. These figures show, if we increase nodes, absolute error decreases. The absolute error distributions at time  $T = 2$  for  $Re=100$  with CSP are shown in Figures 5–7 with a different number of points showing the absolute error distributions at time  $T = 2$  for  $Re=100$ . It can be seen that the error decreases with an increasing the number of points. Figures 8–12 present absolute error distributions at time  $T = 2$  for  $Re=100$  with different shape strategies. Figures 13 presents absolute error distributions at time  $T = 2$  for  $Re=1000$  by CSP. Figures 14–17 show absolute error distributions at time  $T = 2$  for  $Re=1000$  with different shape strategies.



**Figure 2.** Numerical solution of  $U$  and  $V$  at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 1 (Test problem 1).



**Figure 3.** Absolute error graphs at time  $T = 2$  with  $Re = 10$ ,  $n = 441$ , strategy 3 (test problem 1).

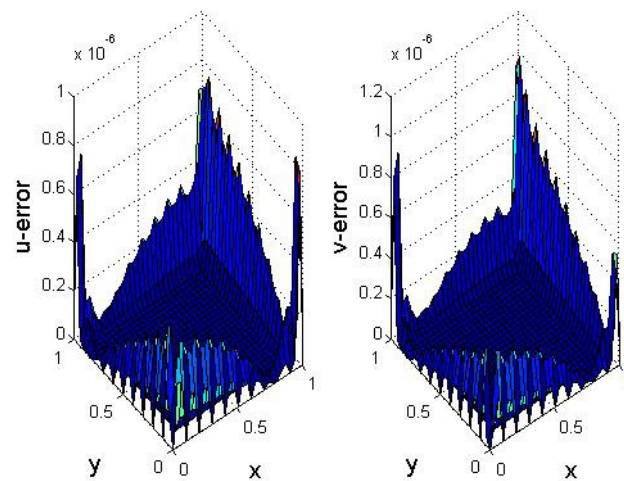


Figure 4. Absolute error graphs at time  $T = 2$  with  $Re = 10$ ,  $n = 1681$ , strategy 3 (test problem 1).

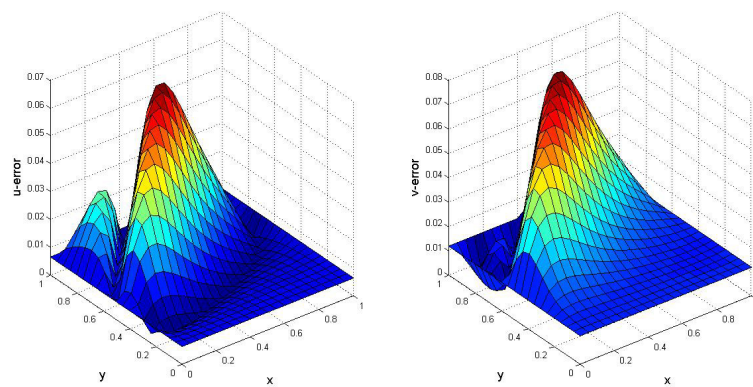


Figure 5. Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ ,  $c = 10$  (test problem 1).

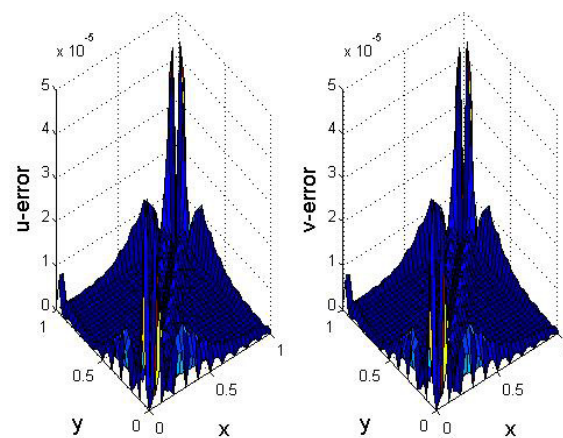


Figure 6. Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 1 (test problem 1).

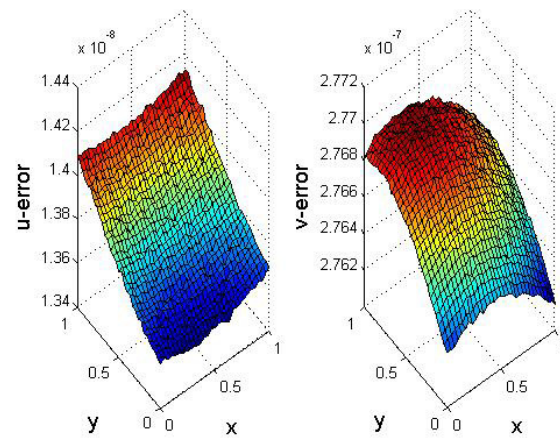


Figure 7. Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 3721$ , strategy 1 (test problem 1).

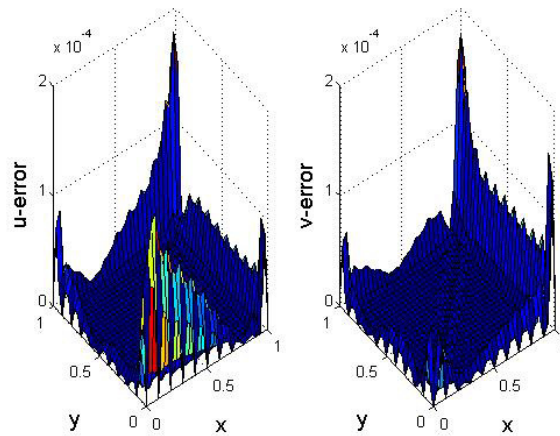


Figure 8. Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 2 (test problem 1).

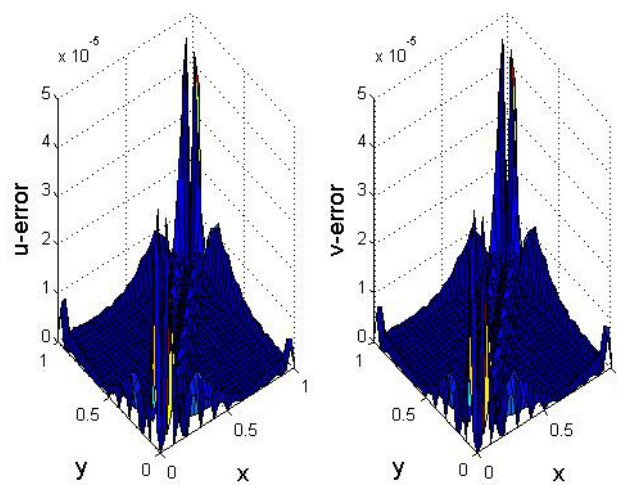
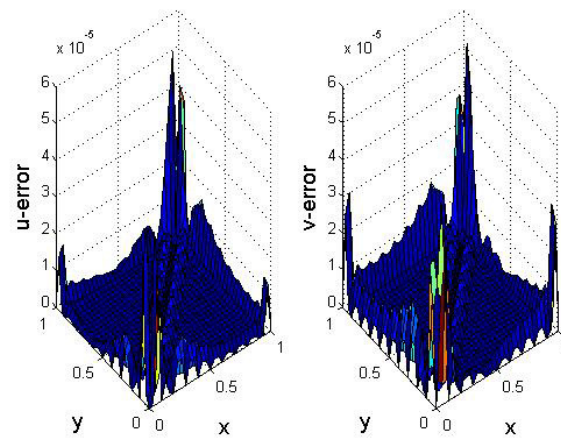
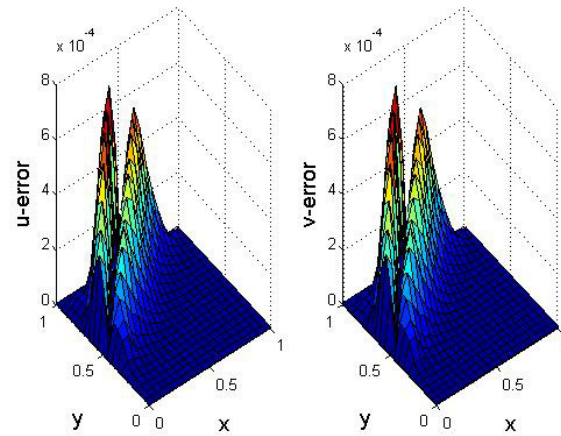


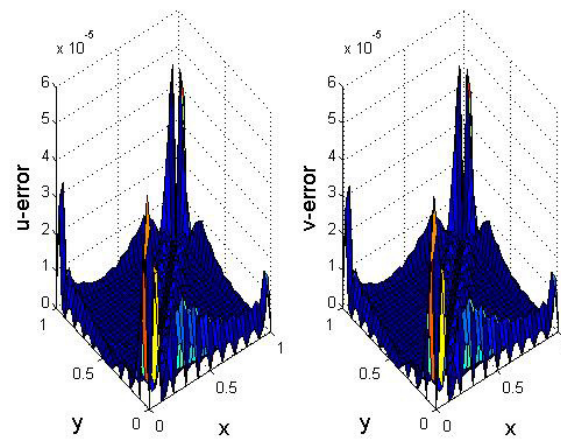
Figure 9. Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 3 (test problem 1).



**Figure 10.** Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 4 (test problem 1).



**Figure 11.** Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 4 (legend points) (test problem 1).



**Figure 12.** Absolute error graphs at time  $T = 2$  with  $Re = 100$ ,  $n = 1681$ , strategy 5 (test problem 1).



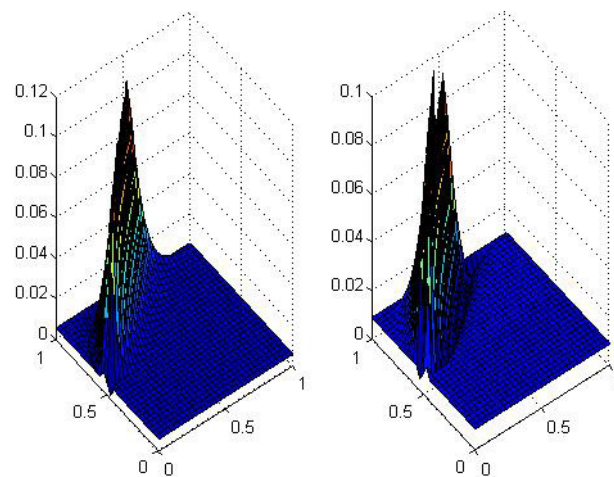


Figure 13. Absolute error graphs at time  $T = 2$  with  $Re = 1000$ ,  $n = 1681$ ,  $c = 10$  (test problem 1).

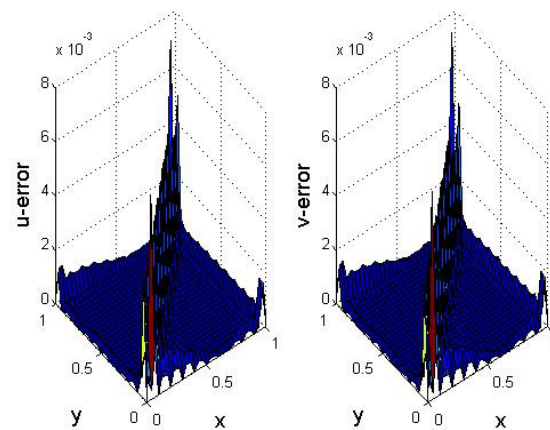


Figure 14. Absolute error graphs at time  $T = 2$  with  $Re = 1000$ ,  $n = 1681$ , strategy 2 (test problem 1).

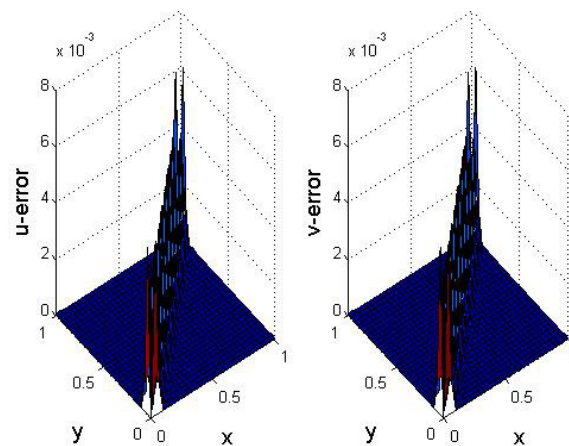


Figure 15. Absolute error graphs at time  $T = 2$  with  $Re = 1000$ ,  $n = 1681$ , strategy 3 (test problem 1).



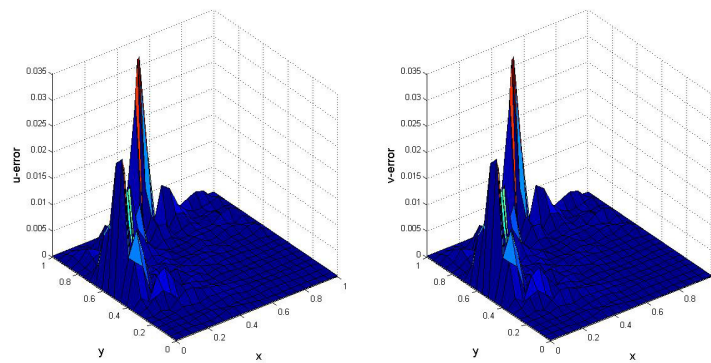


Figure 16. Absolute error graphs at time  $T = 2$  with  $Re = 1000$ ,  $n = 1681$ , strategy 4 (test problem 1).

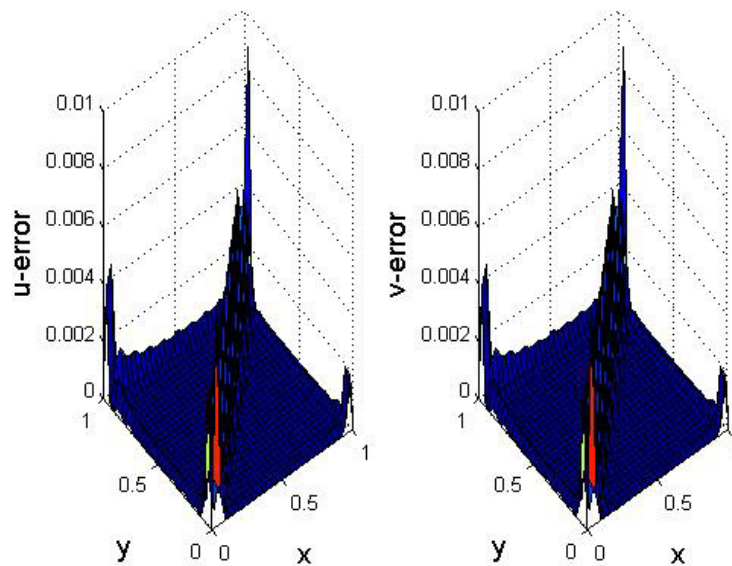


Figure 17. Absolute error graphs at time  $T = 2$  with  $Re = 1000$ ,  $n = 1681$ , strategy 5 (test problem 1).

In Tables 1 and 2, performance of the method is compared with the other methods [24,27,31,35] at selected points. It can be noted that the numerical results obtained from the proposed method are fairly accurate.

Table 1. Numerical results for  $U$  at time  $T = 2$  with  $Re = 100$ ,  $n = 441$ .

$(x,y)$	(0.1,0.1)	(0.3,0.3)	(0.5,0.5)	(0.3,0.7)	(0.1,0.9)	(0.5,0.9)
Exact	0.500482	0.500482	0.500482	0.555675	0.744256	0.555675
Strategy 4	0.500490	0.500514	0.500528	0.554754	0.744165	0.555365
[35]	0.500470	0.500441	0.500414	0.554805	0.744197	0.554489
[24]	0.50035	0.50042	0.50046	0.55609	0.74409	0.55604
[31]	0.49983	0.49977	0.49973	0.55429	0.74340	0.55413
[27]	0.50012	0.50042	0.50041	0.55413	0.74416	0.55637

**Table 2.** Numerical results for  $V$  at time  $T = 2$  with  $Re = 100$ ,  $n = 441$ .

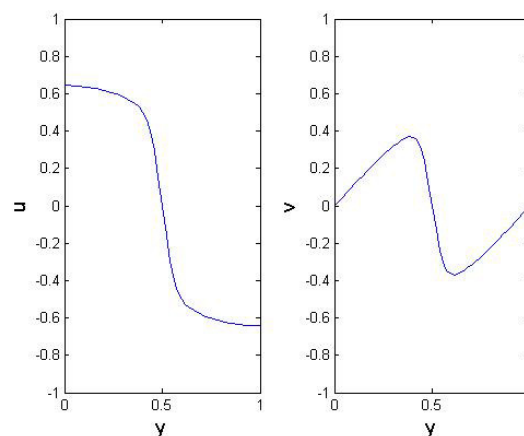
$(x,y)$	(0.1,0.1)	(0.3,0.3)	(0.5,0.5)	(0.3,0.7)	(0.1,0.9)	(0.5,0.9)
Exact	0.999518	0.999518	0.999518	0.944325	0.755744	0.944325
Strategy 4	0.999537	0.999582	0.999617	0.945206	0.755867	0.944679
[35]	0.999530	0.999559	0.999586	0.945195	0.755803	0.945511
[24]	0.99936	0.99951	0.99958	0.94387	0.75592	0.94392
[31]	0.99826	0.99861	0.99821	0.94409	0.75500	0.94441
[27]	0.99946	0.99938	0.99941	0.94387	0.75558	0.94345

**Example 2.** Consider problems (14)–(16) with the following initial and mixed boundary conditions given by [32]:

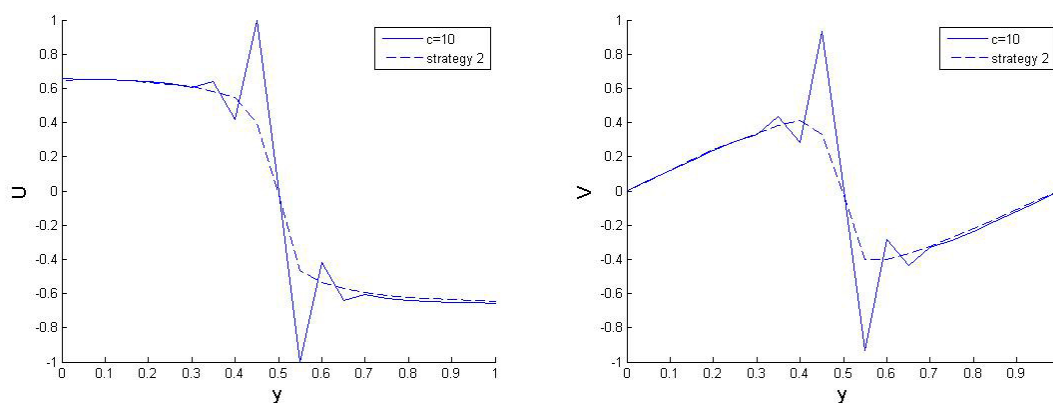
$$\begin{cases} U(x,y,0) = \sin(\pi x) \cos(\pi y), \\ V(x,y,0) = \cos(\pi x) \sin(\pi y), \\ \begin{cases} U(0,y,t) = U(1,y,t) = 0, \\ V(x,0,t) = V(x,1,t) = 0, \\ \frac{\partial U}{\partial n}(x,0,t) = \frac{\partial U}{\partial n}(x,1,t) = 0, \\ \frac{\partial V}{\partial n}(0,y,t) = \frac{\partial V}{\partial n}(1,y,t) = 0. \end{cases} \end{cases}$$

We will compare the obtained solution in our proposed method with the results described in [11,35] because there is not the exact solution. Fake oscillations have been observed by using finite element method (FEM), finite-difference method (FDM), element free Galerkin method [32] and Galerkin-reproducing kernel method [11]. In [35], an adaptive upwind technique has been innovated to avoid wiggles in the recommended local RBF collocation method. As mentioned in [11], even a very fine grid cannot get rid of the oscillatory behavior caused by a sharp gradient. To omit the wiggles and defeat instabilities for  $Re = 1000$ , we have used RBFs with CSP as shown in Figure 18.

The solution is smooth and near the front we do not have any instability for small  $Re$  and, therefore, no VSP is required. In our case, for  $Re = 10,000$ , Figure 19 shows the oscillatory behavior. To avoid these wiggles, we have used VSP (strategy 2) in order to stabilize the solution near the sharp front.



**Figure 18.** Numerical results at time  $T = 0.4$ ,  $x = 0.5$ , with  $Re = 1000$ ,  $c = 10$ ,  $n = 441$ , on the left  $U$  and on the right  $V$ . (test problem 2)



**Figure 19.** Numerical results at time  $T = 0.4$ ,  $x = 0.5$ , with  $Re = 10000$ ,  $n = 441$  Legendre points, on the left  $U$  and on the right  $V$ . (test problem 2)

## 6. Conclusions

Six shape parameter strategies were compared in this study. One strategy used a constant shape, while the other five used a different value of the shape parameter at each center. Results show that VSPs can improve the condition number and the solution accuracy. The approach is successfully applied to solve the 2D nonlinear coupled Burgers' equations with Dirichlet boundary conditions for high  $Re$ . Owing to Neumann boundary conditions, instabilities appear near the sharp gradient without any special filtering technique for  $Re = 1000$ ,  $10,000$  for mixed boundary conditions. These fake oscillations were also observed using the FDM, FEM and element free Galerkin methods. In [35], an adaptive upwind technique was devised to avoid these wiggles for  $Re = 1000$ . To get rid of the oscillatory behavior, we proposed constant and VSP for  $Re = 1000$ ,  $10,000$ , respectively.

**Acknowledgments:** The authors are very grateful to reviewers for carefully reading this paper and for their comments and suggestions which have improved the paper.

**Author Contributions:** All authors contributed equally and significantly to the study and preparation of the article. They have read and approved the final manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M.; Krysl, P. Meshless methods: An overview and recent developments. *Comput. Methods Appl. Mech. Eng.* **1996**, *139*, 3–47.
2. Franke, C.; Schaback, R. Convergence order estimates of meshless collocation methods using radial basis functions. *Adv. Comput. Math.* **1998**, *8*, 381–399.
3. Franke, C.; Schaback, R. Solving partial differential equations by collocation using radial basis functions. *Appl. Math. Comput.* **1998**, *93*, 73–82.
4. Kansa, E.J. Application of Hardy's multiquadric interpolation to hydrodynamics. In Proceedings of the Society for Computer Simulation (SCS) Multiconference, San Diego, CA, USA, 23–25 January 1986; Volume 4, pp. 111–117.
5. Schaback, R. A computational tool for comparing all linear PDE solvers. *Adv. Comp. Math.* **2015**, *41*, 333–355.
6. Schaback, R.; Wendland, H. Kernel techniques: From machine learning to meshless methods. *Acta Numer.* **2006**, *15*, 543–639.
7. Abbasbandy, S.; Azarnavid, B.; Alhuthali, M.S. A shooting reproducing kernel Hilbert space method for multiple solutions of nonlinear boundary value problems. *J. Comput. Appl. Math.* **2015**, *279*, 293–305.

8. Azarnavid, B.; Parvaneh, F.; Abbasbandy, S. Picard-Reproducing kernel Hilbert space method for solving generalized singular nonlinear Lane-Emden type equations. *Math. Model. Anal.* **2015**, *20*, 754–767.
9. Hon, Y.; Schaback, R. Solvability of partial differential equations by meshless kernel methods. *Adv. Comput. Math.* **2008**, *28*, 283–299.
10. Lee, C.F.; Ling, L.; Schaback, R. On convergent numerical algorithms for unsymmetric collocation. *Adv. Comput. Math.* **2009**, *30*, 339–354.
11. Mohammadi, M.; Mokhtari, R.; Panahipour, H. A Galerkin-reproducing kernel method: Application to the 2D nonlinear coupled Burgers' equations. *Eng. Anal. Bound. Elem.* **2013**, *37*, 1642–1652.
12. Mohammadi, M.; Mokhtari, R.; Schaback, R. A meshless method for solving the 2D Brusselator reaction-diffusion system. *Comput. Model. Eng. Sci.* **2014**, *101*, 113–138.
13. Bozzini, M.; Lenarduzzi, L.; Rossini, M.; Schaback, R. Interpolation with variably scaled kernels. *IMA J. Numer. Anal.* **2015**, *35*, 199–219.
14. Hardy, R.L. Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **1971**, *76*, 1905–1915.
15. Huang, C.S.; Lee, C.F.; Cheng, A.D. Error estimate, optimal shape factor, and high precision computation of multiquadric collocation method. *Eng. Anal. Bound. Elem.* **2007**, *31*, 614–623.
16. Kansa, E.J.; Aldredge, R.C.; Ling, L. Numerical simulation of two-dimensional combustion using mesh-free methods. *Eng. Anal. Bound. Elem.* **2009**, *33*, 940–950.
17. Sarra, S.A.; Sturgill, D. A random variable shape parameter strategy for radial basis function approximation methods. *Eng. Anal. Bound. Elem.* **2009**, *33*, 1239–1245.
18. Golbabai, A.; Rabiei, H. Hybrid shape parameter strategy for the RBF approximation of vibrating systems. *Int. J. Comput. Math.* **2012**, *89*, 2410–2427.
19. Dehghan, M.; Mohammadi, V. The method of variably scaled radial kernels for solving two-dimensional magnetohydrodynamic (MHD) equations using two discretizations: The Crank-Nicolson scheme and the method of lines (MOL). *Comput. Math. Appl.* **2015**, *70*, 2292–2315.
20. Dereli, Y.; Schaback, R. The meshless kernel-based method of lines for solving the equal width equation. *Appl. Math. Comp.* **2013**, *219*, 5224–5232.
21. Mavrič, B.; Šarler, B. Local radial basis function collocation method for linear thermoelasticity in two dimensions. *Int. J. Numer. Methods Heat Fluid Flow* **2015**, *25*, 1488–1510.
22. Sarra, S. A local radial basis function method for advection-diffusion-reaction equations on complexly shaped domains. *Appl. Math. Comp.* **2012**, *218*, 9853–9865.
23. Burger, J.M. A mathematical model illustrating the theory of turbulence. *Adv. Appl. Mech.* **1948**, *1*, 171–199.
24. Ali, A.; Islam, S.U.; Haq, S. A computational meshfree technique for the numerical solution of the two-dimensional coupled Burgers' equations. *Int. J. Comput. Methods Eng. Sci. Mech.* **2009**, *10*, 406–412.
25. Nee, J.; Duan, J. Limit set of trajectories of the coupled viscous Burgers' equations. *Appl. Math. Lett.* **1998**, *11*, 57–61.
26. Perko, J.; Šarler, B. Weight function shape parameter optimization in meshless methods for non-uniform grids. *Comput. Model. Eng. Sci.* **2007**, *19*, 55–68.
27. Younga, D.; Fana, C.; Hua, S.; Alturi, S. The Eulerian-Lagrangian method of fundamental solutions for two-dimensional unsteady Burgers' equations. *Eng. Anal. Bound. Elem.* **2009**, *32*, 395–412.
28. Fletcher, J.D. Generating exact solutions of the two-dimensional Burgers' equations. *Int. J. Numer. Methods Fluids* **1983**, *3*, 213–216.
29. Mokhtari, R.; Mohammadi, M. New exact solutions to a class of coupled nonlinear PDEs. *Int. J. Nonlinear Sci. Numer. Simul.* **2009**, *10*, 779–796.
30. Fletcher, C.A.J. A comparison of finite element and finite difference solution of the one and two dimensional Burgers' equations. *J. Comput. Phys.* **1983**, *51*, 159–188.
31. Bahdir, A.R. A fully implicit finite difference scheme for two dimensional Burgers' equations. *Appl. Math. Comput.* **2003**, *137*, 131–137.
32. Zhang, X.H.; Ouyang, J.; Zhang, L. Element-free characteristic Galerkin method for Burgers' equation. *Eng. Anal. Bound. Elem.* **2009**, *33*, 356–362.
33. Zhang, L.; Ouyang, J.; Wang, X.; Zhang, X. Variational multiscale element-free Galerkin method for 2D Burgers' equation. *J. Comput. Phys.* **2010**, *29*, 7147–7161.

34. Zhang, W.; Ahang, C.; Xi, G. An explicit Chebyshev pseudospectral multigrid method for incompressible Navier-Stokes equations. *Comput. Fluids* **2010**, *39*, 178–188.
35. Šarler, B.; Vertnik, R.; Kosec, G. Radial basis function collocation method for the numerical solution of the two-dimensional transient nonlinear coupled Burgers' equations. *Appl. Math. Model.* **2012**, *36*, 1148–1160.
36. Wendland, H. Scattered data approximation. In *Cambridge Monograph on Applied and Computational Mathematics*; Cambridge University Press: Cambridge, UK, 2005.
37. Buhmann, M.D. *Radial Basis Functions*; Cambridge University Press: Cambridge, UK, 2004.
38. Schaback, R. Kernel-Based Meshless Methods. Available online: <http://num.math.uni-goettingen.de/schaback/teaching/Appverf-II.pdf> (access on 18 July 2007).
39. Sanyasiraju, Y.; Satyanarayana, C. On optimization of the RBF shape parameter in a grid-free local scheme for convection dominated problems over non-uniform centers. *Appl. Math. Model.* **2013**, *37*, 7245–7272.
40. Dehghan, M.; Mohammadi, V. The numerical solution of Fokker–Planck equation with radial basis functions (RBFs) based on the meshless technique of Kansa's approach and Galerkin method. *Eng. Anal. Bound. Elem.* **2014**, *47*, 38–63.
41. Kansa, E.J. Multiquadrics—A scattered data approximation scheme with applications to computational fluid dynamics I: Surface approximations and partial derivative estimates. *Comput. Math. Appl.* **1990**, *19*, 127–145.
42. Kansa, E.J. Multiquadrics—A scattered data approximation scheme with applications to computational fluid dynamics II: Solutions to parabolic hyperbolic, and elliptic partial differential equations. *Comput. Math. Appl.* **1990**, *19*, 147–161.
43. Xiang, S.; Wang, K.M.; Ai, Y.T.; Sha, Y.D.; Shi, H. Trigonometric variable shape parameter and exponent strategy for generalized multiquadric radial basis function approximation. *Appl. Math. Model.* **2012**, *36*, 1931–1938.
44. Das, P.; Natesan, S. Adaptive mesh generation for singularly perturbed fourth-order ordinary differential equations. *J. Comput. Math.* **2015**, *92*, 562–578.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).