



# Article On the Approximation of a Nonlinear Biological Population Model Using Localized Radial Basis Function Method

# Marjan Uddin \*<sup>(D)</sup>, Hazrat Ali and Muhammad Taufiq

Department of Basic Sciences, University of Engineering and Technology, Peshawar 25000, Pakistan; Hazratali@uetpeshawar.edu.pk (H.A.); taufiq@uetpeshawar.edu.pk (M.T.)

\* Correspondence: marjan@uetpeshawar.edu.pk

Received: 11 April 2019; Accepted: 21 May 2019; Published: 23 May 2019



**Abstract:** A localized radial basis function meshless method is applied to approximate a nonlinear biological population model with highly satisfactory results. The method approximates the derivatives at every point corresponding to their local support domain. The method is well suited for arbitrary domains. Compared to the finite element and element free Galerkin methods, no integration tool is required. Four examples are demonstrated to check the efficiency and accuracy of the method. The results are compared with an exact solution and other methods available in literature.

Keywords: local RBF; biological population model; RK4 method; multi-quadric RBF

MSC: Primary 47A15; Secondary 46A32, 47D20

# 1. Introduction

Biological population models have attracted many researchers in the last few years. Consider the nonlinear biological population model [1]

$$\rho_t(\mathbf{x},t) = \rho_{xx}^2(\mathbf{x},t) + \rho_{yy}^2(\mathbf{x},t) + \sigma(\rho), \ \mathbf{x} = (x,y) \in \Omega \subseteq \mathbb{R}^d, \ d = 1,2,$$
(1)

where  $\sigma$  is a function of  $\rho(\mathbf{x}, t)$  subject to the initial condition  $\rho(\mathbf{x}, 0)$ , the solution  $\rho(\mathbf{x}, t)$  denotes the density of population in a region  $\Gamma$ , and  $\sigma$  represent the population supply due to births and deaths. The fields  $\rho(\mathbf{x}, t)$ , *V* (diffusion velocity) and  $\sigma$  must obey the following population balance law for a regular subregion of region  $\Gamma$  for all time *t*.

$$\frac{d}{dt} \int_{\mathbb{R}} \rho dV + \int_{\partial \mathbb{R}} \rho V \cdot \mathbf{n} \, dV + \int_{\mathbb{R}} \sigma dV \tag{2}$$

where **n** denotes the outward unit normal to the boundary  $\partial \mathbb{R}$  of  $\mathbb{R}$ . Physically, the equation describes that rate of population directly supplied to  $\mathbb{R}$  is equal to the sum of the rate of change of population and the rate at which individuals leave  $\mathbb{R}$  across the boundary [1–3]. Several methods have been applied to consider the uniqueness, existence and regularity of these models.

In recent work, the biological population model (BPM) has been numerically approximated by various methods [4–6]. Meshless procedures have attracted researchers because they can be applied to a complex shaped domain with high dimensional problems. These methods include the radial basis function (RBF) [7–9], smooth particle hydrodynamics methods (SPH) [10], reproducing kernel particle method (RKPM) [11], element free Galerkin method (EFG) [11], and meshless local Petrov Galerkin method (MLPG) [12].

Since from discovery of mutiquadrics by Hardy [13], the RBFs (radial basis function) attracted researchers. Thin plate splines were proposed by Douchen [14]. Edwards Kansa for the first time used RBFs to solve PDEs [15], which greatly improved the numerical solution of PDEs. Presently RBFs are applied to solve a lot of mathematical models arising in engineering and science. During the past decades, robust kernel based methods have been developed by Atluri [12], Buhumann [16], Wendland [17], Fasshaur [18,19] and Levesely [20] and many others [21–25]. The convergence theory of the RBFs method was proved by Schaback [26]. The importance of RBFs for solving PDES is easy to implement, robustness and applicability to complicated and multi dimensional problems.

To overcome the problems occurring during global RBFs, partition of unity, domain decomposition method and greedy algorithms have been discovered [26–28]. Another alternative is the implementation of local RBFs (LRBFs) [29–31]. In LRBFs, small domains centred at each points are used to give rise to differentiation weights. In the present scenario, we have applied local RBFs to solve the nonlinear biological model.

#### 2. Description of the Method

### 2.1. Spatial Approximation

The multivariate data interpolation problem involves recovering an unknown function  $\rho : \mathbb{R}^2 \to \mathbb{R}$ from M functional values  $\{\rho(\mathbf{x}_1), ..., \rho(\mathbf{x}_M)\} \subset \mathbb{R}^2$ , where the centres  $\mathbf{x}_1, ..., \mathbf{x}_M \in \Omega$  and  $\Omega \subset \mathbb{R}^2$  is bounded domain and the centres are taken anywhere in the domain  $\Omega$ . In local RBF approximation techniques, at every centre  $\mathbf{x}_i \in \Omega$ , the local interpolant

$$\rho(\mathbf{x}_j, t) = \sum_{\mathbf{x}_k \in \Omega_j} \eta_k(t) \phi(\|\mathbf{x}_j - \mathbf{x}_k\|),$$
(3)

where  $\eta^j = [\eta_1, ..., \eta_m]$  is expansion coefficients and  $||\mathbf{x}_j - \mathbf{x}_k||$  is the Euclidean distance between  $\mathbf{x}_j$  and  $\mathbf{x}_k$ . Here,  $\phi(r)$  is a kernel function (Multiquadrics) defined for  $r \ge 0$  and  $\Omega_j \subset \Omega$  is the local support domain for each node  $\mathbf{x}_j$  and have *m* nearest centres with centre  $\mathbf{x}_j$ . The *M* number of  $m \times m$  linear systems is

$$\mathbf{P}^{j} = \mathbf{N}^{j} \eta^{j}, \quad j = 1, 2, ..., M,$$
 (4)

where matrix entries of  $\mathbf{N}^{j}$  are  $a_{kj}^{j} = \phi(||\mathbf{x}_{k} - \mathbf{x}_{j}||)$ , where  $\mathbf{x}_{k}$  and  $\mathbf{x}_{j} \in \Omega_{j}$ . The matrix  $\mathbf{N}^{j}$  is interpolation matrix corresponding to centre  $x_{j}$ . Every system is solved for unknown coefficients corresponding to each local domain  $\Omega_{j}$ . Applying the differential operator  $\mathcal{L}$  to Equation (3)

$$\mathcal{L}\rho(\mathbf{x}_{j},t) = \sum_{\mathbf{x}_{k}\in\Omega_{i}}\eta_{k}(t)\mathcal{L}\phi(\|\mathbf{x}_{i}-\mathbf{x}_{j}\|).$$
(5)

The expression in (5) can be written as element wise product of two vectors,

$$\mathcal{L}\rho(\mathbf{x}_{j},t) = \mathbf{Z}^{j} \cdot \boldsymbol{\eta}^{j},\tag{6}$$

where  $\eta^{j}$  is the  $m \times 1$  vector of unknown coefficients, and  $\mathbf{Z}^{j}$  is  $1 \times m$  vector with entries

$$\mathbf{Z}^{j} = \mathcal{L}\phi(\|\mathbf{x}_{j} - \mathbf{x}_{k}\|), \quad x_{k} \in \Omega_{j}.$$
(7)

To eliminate the unknown coefficients, Equation (4)

$$\eta^j = (\mathbf{N}^j)^{-1} \mathbf{P}^j,\tag{8}$$

on substituting the values of  $\eta^j$  from (8) in (6) we have ,

$$\mathcal{L}\rho(\mathbf{x}_{j},t) = \mathbf{Z}^{j}(\mathbf{N}^{j})^{-1}\mathbf{P}^{j} = \mathbf{w}^{j}\mathbf{P}^{j},\tag{9}$$

where,

$$\mathbf{w}^j = \mathbf{Z}^j (\mathbf{N}^j)^{-1},\tag{10}$$

gives weight to centre  $x_j$ . For all centres, it gives

$$\mathcal{L}\rho = \mathbf{W}\mathbf{P},\tag{11}$$

where **W** is  $M \times M$  sparse differentiation matrix, each row has and M - m zeros elements, where *m* is the stencil size in each local domain  $\Omega_i$ .

#### 2.2. Temporal Approximation

After spatial local RBF approximation, we obtained the following system of ODEs

$$\frac{d\rho}{dt} = F(\rho). \tag{12}$$

In our case  $F(\rho) = WP$ . For discretization any built-in Matlab ODEs solver can be used, such as ode113 or ode45, and  $\rho_0$  has been taken as initial solution vector. Note that ode45 uses an explicit Runge–Kutta (4,5) formula [32]. For computing  $\rho(t_n)$ , needs solution at the preceding time  $\rho(t_{n-1})$ . For ODEs computation, the fourth-order Runge–Kutta method has been taken and the time step  $\delta t$  is taken manually [33]. The method of lines approach is the solution of system of ODEs using finite difference formula (FD) in *t*. The rule of thumb says if all the eigenvalues of the spatial discretization operator scaled using  $\delta t$ , lies in the stability region of the time discretization operator, the method of lines will be stable.

$$r_{1} = F(\rho^{n}),$$

$$r_{2} = F(\rho^{n} + \frac{\delta t}{2}r_{1}),$$

$$r_{3} = F(\rho^{n} + \frac{\delta t}{2}r_{2}),$$

$$r_{4} = F(\rho^{n} + \delta tr_{3}), \rho^{n+1} = \rho^{n} + \frac{\delta t}{6}(r_{1} + 2r_{2} + 2r_{3} + r_{4}).$$
(13)

In addition we may use some other efficient methods for time integrations such as exponential integrator schemes [34,35].

#### 2.3. Choosing Optimal Shape Parameter

A large number of kernel functions such as Multiquadrics, Thin Plate Splines, Quadratics and Inverse Mutiquadrics can be used. We used Multiquadrics  $\phi(r) = \sqrt{1 + (rc)^2}$  with a shape parameter c. The numerical solution of the problem also depends on the parameter c. Finding most suitable value of shape parameter analytically for various RBFs is still an open problem. A large number of papers are available for finding good value of shape parameter (see for example [36] and the references therein). The matrix condition number (MCN) is used to check the sensitivity of a linear system and accuracy of the solution. Good conditioning results require small values of shape parameter and large separation distance between centres. Both results cannot occur at the same time, which is known as the uncertainty principle [37]. The smallest error occurs when system condition number is kept in range of Algorithm 1.

#### Algorithm 1: Optimal shape parameter.

```
Start: MCN = 1;

10^{13} < MCN < 10^{15}

while MCN < MCN_{min} and MCN > MCN_{max} do

\begin{vmatrix} U, S, V = svd(\mathbf{N}^i); \\ MCN = max(S)/min(S); \\ if <math>MCN < MCN_{min} then

\mid c = c - dc;

else

\mid c = c + dc;

end

c (OptimalShape) = c;

end
```

## 3. Numerical Experiments

This section is devoted to demonstrate the accuracy and performance of LRBF method by solving the three test problems of the population model (1). The accuracy is tested in terms of maximum absolute error (MAE) and root mean squared error (RMSE) defined by the following equations

$$MAE = \max_{1 \le i \le M} \| \rho(\mathbf{x}_{j}, t) - \widehat{\rho}(\mathbf{x}_{j}, t) \|,$$
(14)

$$RMSE = \sqrt{\frac{1}{M} \Sigma_{i=1}^{M} (\rho(\mathbf{x}_{j}, t) - \widehat{\rho}(\mathbf{x}_{j}, t))^{2}},$$
(15)

where  $\rho(\mathbf{x}_i, t)$  and  $\hat{\rho}(\mathbf{x}_i, t)$  are the exact and the approximate solutions, respectively.

#### 3.1. Example 1

We consider the biological population model (1)

$$\rho_t(\mathbf{x},t) = \rho_{xx}^2(\mathbf{x},t) + \rho_{yy}^2(\mathbf{x},t) + h^{\alpha}\rho(\mathbf{x},t)(1-h^{\beta}\rho(\mathbf{x},t)), \ \mathbf{x} = (x,y) \in [0\ 1]^2, \tag{16}$$

with the initial condition

$$o(\mathbf{x},0) = \sqrt{xy},\tag{17}$$

where  $h = \frac{1}{2}$ ,  $\alpha = 1$  and h = 0 with exact solution [6]

$$\rho(\mathbf{x},t) = e^t \sqrt{xy}.\tag{18}$$

The present problem is solved by the proposed method in the square domain  $\Omega = [0, 1]^2$  for various number of nodes in the global domain  $\Omega$  and local sub-domains  $\Omega_i$ . For time integration, the RK4 method with time step  $\delta t = 0.0001$  is used. The sparsity of differentiation matrix W and distribution of nodes in the global and local sub-domains corresponding to an interior and boundary points are shown in Figure 1. Figure 2 shows the point wise error and Figure 3 shows the problem solution  $\rho(x, 0.5)$  at different times. The results for M = 400 total number of nodes in the global domain, different stencil size *ns* of local sub-domain, matrix condition number (MCN) and shape parameter are shown in Table 1. The comparison of our method is made with the results in [5] in Table 2, which shows that our results are more accurate and stable.



**Figure 1.** (a) Sparsity of matrix *W*, when ns = 30 and M = 400, and (b) centre distribution in domain  $[0, 1]^2$  corresponding to Example 1.



**Figure 2.** (a) Nodes *M* against MAE and RMSE norms for ns = 20, and (b) stability plots of our numerical scheme corresponding to Example 1.



**Figure 3.** (a) Numerical solution  $\rho(x, 0.5)$  at different times *t*, and (b) point wise error when t = 1, ns = 20 and  $\delta = 0.0001$  when M = 400 corresponding to Example 1.

**Table 1.** Numerical results when N = 400 for domain  $[0, 1]^2$  (Example 1).

| ns | MCN  | MAE  | RMSE   | C-Time   | Shape (c)   |
|----|--|--|--|--|---|
| 5  | $3.099\times10^{14}$                         | $1.931\times 10^{-5}$  | $8.336 	imes 10^{-6}$  | 0.950  | 0.010   |
| 11 | $1.241 \times 10^{13}$                       | $1.915	imes10^{-5}$  | $7.935 	imes 10^{-6}$  | 0.858  | 0.520   |
| 21 | $7.732 	imes 10^{13}$                        | $1.215 	imes 10^{-5}$  | $7.643	imes10^{-6}$  | 2.319  | 1.160   |
| 49 | $2.929	imes10^{14}$                          | $4.109	imes10^{-5}$  | $1.269 	imes 10^{-5}$  | 1.869  | 1.990   |
| 60 | $2.941	imes10^{14}$                          | $2.725 	imes 10^{-5}$  | $1.121 	imes 10^{-5}$  | 2.504  | 2.150   |
| 80 | $5.096 	imes 10^{14}$                        | $2.961 	imes 10^{-5}$  | $1.287 	imes 10^{-5}$  | 4.199  | 2.320   |
|    | <i>ns</i><br>5<br>11<br>21<br>49<br>60<br>80 | $\begin{array}{c ccc} \textit{ns} & \textbf{MCN} \\ \hline 5 & 3.099 \times 10^{14} \\ 11 & 1.241 \times 10^{13} \\ 21 & 7.732 \times 10^{13} \\ 49 & 2.929 \times 10^{14} \\ 60 & 2.941 \times 10^{14} \\ 80 & 5.096 \times 10^{14} \\ \end{array}$ | $\begin{array}{c cccc} \textit{ns} & \textbf{MCN} & \textbf{MAE} \\ \hline 5 & 3.099 \times 10^{14} & 1.931 \times 10^{-5} \\ 11 & 1.241 \times 10^{13} & 1.915 \times 10^{-5} \\ 21 & 7.732 \times 10^{13} & 1.215 \times 10^{-5} \\ 49 & 2.929 \times 10^{14} & 4.109 \times 10^{-5} \\ 60 & 2.941 \times 10^{14} & 2.725 \times 10^{-5} \\ 80 & 5.096 \times 10^{14} & 2.961 \times 10^{-5} \\ \end{array}$ | $\begin{array}{c ccccc} ns & MCN & MAE & RMSE \\ \hline 5 & 3.099 \times 10^{14} & 1.931 \times 10^{-5} & 8.336 \times 10^{-6} \\ 11 & 1.241 \times 10^{13} & 1.915 \times 10^{-5} & 7.935 \times 10^{-6} \\ 21 & 7.732 \times 10^{13} & 1.215 \times 10^{-5} & 7.643 \times 10^{-6} \\ 49 & 2.929 \times 10^{14} & 4.109 \times 10^{-5} & 1.269 \times 10^{-5} \\ 60 & 2.941 \times 10^{14} & 2.725 \times 10^{-5} & 1.121 \times 10^{-5} \\ 80 & 5.096 \times 10^{14} & 2.961 \times 10^{-5} & 1.287 \times 10^{-5} \end{array}$ | $\begin{array}{c ccccccccccccccccccccccccccccccccccc$ |

**Table 2.** Numerical solution of  $\rho(x, y, t)$  at x = 0.5, y = 0.5 compared with exact solution and Zhang et al. [5] (Example 1).

| Number of Nodes = $13 \times 13$ | <i>t</i> = 0.1 | t = 0.2 | t = 0.3 | t = 0.4 | t = 0.5 |
|----------------------------------|----------------|---------|---------|---------|---------|
| Present Solution                 | 1.2628         | 1.1426  | 1.0339  | 0.9355  | 0.8465  |
| Zhang et al. Solution [5]        | 1.2574         | 1.1382  | 1.0302  | 0.9325  | 0.844   |
| Exact Solution                   | 1.2628         | 1.1426  | 1.0339  | 0.9355  | 0.8465  |

#### 3.2. Example 2

The biological population model defined by (1) with the initial condition

$$\rho(x,y,0) = \frac{1}{4}\sqrt{2x^2 + 2y^2 + y + 5},\tag{19}$$

with the parameters values  $h = \frac{1}{96}$ ,  $\alpha = -1$ , h = 48 and  $\beta = 1$ . The exact solution of this problem given in [6] is

$$\rho(x,y,t) = \frac{1}{4}\sqrt{2x^2 + 2y^2 + y + \frac{t}{3} + 5}.$$
(20)

The problem is solved in the square domain  $\Omega = [0, 1]^2$ , the numerical scheme is advanced in time with step size  $\delta t = 0.0001$ . Figure 4 compares the exact and numerical solutions and Figure 5 shows the problem solution  $\rho(x, 0.5)$  at different times.



**Figure 4.** (a) Exact against (b) numerical solutions, when t = 1,  $\delta t = 0.0001$ , ns = 20, M = 400, corresponding to Example 2.



**Figure 5.** Numerical solution at different times *t* for  $\rho(x, 0.5)$ , corresponding to Example 2.

#### 3.3. Example 3

The biological population model defined by (1) with initial condition

$$\rho(x, y, 0) = e^{\frac{x+y}{3}},\tag{21}$$

with the values of parameters h = -1,  $\alpha = 1$ ,  $h = \frac{-8}{9}$  and  $\beta = 1$ . The exact solution [6] is given by

$$\rho(x, y, t) = e^{\frac{x+y}{3}-t}.$$
(22)

The results are obtained by the present local kernel based method in the domain  $[0.1]^2$ . Figure 6 compares the exact and numerical solution and Figure 7 shows the point wise error and the model solution  $\rho(x, 0.5)$  at different times. The results for different *M* and *ns* are obtained and the matrix condition number MCN and shape parameter *c* are shown in Table 3. The comparison of the present method is made with [5] in Table 4.



**Figure 6.** (a) Exact against (b) numerical solutions, when t = 1,  $\delta t = 0.0001$ , ns = 20, M = 400 corresponding to Example 3.



**Figure 7.** (a) Point wise error, when t = 1, ns = 20 and  $\delta t = 0.0001$ , and (b) numerical solution, corresponding to Example 3.

|              | ns | MCN                    | MAE                  | RMSE                 | C-Time  | Shape (c) |
|--------------|----|------------------------|----------------------|----------------------|---------|-----------|
| RK4 (MQ RBF) | 5  | $3.099\times 10^{14}$  | $1.35 	imes 10^{-4}$ | $8.85 	imes 10^{-5}$ | 0.14538 | 0.01      |
|              | 10 | $5.136	imes10^{13}$    | $6.93	imes10^{-5}$   | $4.70	imes10^{-5}$   | 0.76946 | 0.43      |
|              | 15 | $4.549 	imes 10^{13}$  | $6.90	imes10^{-5}$   | $4.75 	imes 10^{-5}$ | 0.77773 | 0.85      |
|              | 25 | $1.367	imes10^{14}$    | $7.50 	imes 10^{-5}$ | $5.47	imes10^{-5}$   | 0.88658 | 1.41      |
|              | 50 | $3.225 	imes 10^{14}$  | $7.34	imes10^{-5}$   | $4.67	imes10^{-5}$   | 1.26577 | 1.99      |
|              | 70 | $3.558 \times 10^{14}$ | $6.92 	imes 10^{-5}$ | $4.72 	imes 10^{-5}$ | 1.57271 | 2.26      |
|              | 80 | $5.096 	imes 10^{14}$  | $7.53 	imes 10^{-5}$ | $4.53	imes10^{-5}$   | 1.74231 | 2.32      |

**Table 3.** Numerical results corresponding to Example 3 in  $[0, 1]^2$ .

**Table 4.** Numerical solution of  $\rho(x, y, t)$  at x = 0.5, y = 0.5 compared with exact solution and Zhang et al. [5] (Example 3).

| Number of Nodes = 13×13                       | <i>t</i> = 0.1   | <i>t</i> = 0.2   | t = 0.3          | t = 0.4          | <i>t</i> = 0.5    |
|---|------------------|------------------|------------------|------------------|-------------------|
| Present Solution<br>Zhang et al. Solution [5] | 1.2628<br>1.2574 | 1.1426<br>1.1382 | 1.0339<br>1.0302 | 0.9355<br>0.9325 | $0.8465 \\ 0.844$ |
| Exact Solution                                | 1.2628           | 1.1426           | 1.0339           | 0.9355           | 0.8465            |

## 3.4. Example 4

In this problem the present method is applied over irregular domains for example Ameoba and Cassini like shaped domains for solving same model discussed in Example 1 above. The parametric equations of such types irregular boundaries  $\partial \Omega$  are defined as

$$\partial \Omega = \{(x,y) | x = r(\theta) \cos(\theta), y = r(\theta) \sin(\theta), 0 \le \theta < 2\pi\},$$
(23)

where for Ameoba-shaped boundary  $r(\theta)$  is defined by

$$r(\theta) = e^{\sin(\theta)} \sin^2(2\theta) + e^{\cos\theta} \cos^2(2\theta), \tag{24}$$

while for Cassini-shaped boundary  $r(\theta)$  is defined by

$$r(\theta) = \sqrt[3]{\cos(4\theta) + \sqrt{\frac{18}{5} - \sin^2(4\theta)}},$$
 (25)

The irregular domains with inner and boundary stencils are shown in Figure 8. The exact and numerical solutions corresponding to Ameoba-like domain with 100 boundary points and Cassini-shape domain with M = 220 are shown in Figures 9 and 10 and Table 5, respectively.

| Ameoba |                       |                       | Cassini              |                       |  |  |
|--------|-----------------------|-----------------------|----------------------|-----------------------|--|--|
| ns     | MAE                   | RMSE                  | MAE                  | RMSE                  |  |  |
| 5      | $9.40	imes10^{-4}$    | $3.52 	imes 10^{-4}$  | $1.63 	imes 10^{-4}$ | $1.02 	imes 10^{-4}$  |  |  |
| 15     | $2.40	imes10^{-4}$    | $2.21	imes10^{-4}$    | $7.33	imes10^{-4}$   | $4.32 	imes 10^{-4}$  |  |  |
| 30     | $3.20 	imes 10^{-3}$  | $9.12 	imes 10^{-3}$  | $6.36	imes10^{-4}$   | $3.82 	imes 10^{-4}$  |  |  |
| 45     | $1.78	imes10^{-4}$    | $1.51 	imes 10^{-4}$  | $2.51	imes10^{-4}$   | $1.56	imes10^{-4}$    |  |  |
| 60     | $1.20 \times 10^{-4}$ | $2.12 \times 10^{-4}$ | $4.45 	imes 10^{-4}$ | $2.22 \times 10^{-4}$ |  |  |

**Table 5.** MAE and RMSE for Ameoba and Cassini-shaped domains at t = 1 with different stencil sizes.



Figure 8. (a) Ameoba and (b) Cassini-shaped domain with interior (red) and boundary (green) stencil.



(a)

**Figure 9.** (a) Exact against (b) numerical solution of Ameoba-shaped domain when t = 1,  $\delta t = 0.0001$ , ns = 10, M = 291 to Example 4.



**Figure 10.** (a) Exact against (b) numerical solution of Cassini-shaped domain when t = 1,  $\delta t = 0.0001$ , ns = 10, M = 220 to Example 4.

## 4. Concluding Remarks

In the present paper, the nonlinear biological population model has been numerically simulated using localized RBF method. The multiquadric RBF has been used as the radial kernel. The shape parameter has been chosen using the uncertainty principle. The nonlinear equation is converted to a system of ODEs, which is then advanced in time by the RK4 method. The numerical results are compared with the benchmark problems. The present numerical scheme is more accurate and stable as compared to other numerical methods for solving the biological model considered in this work.

Author Contributions: Investigation, M.U.; Methodology, H.A. and M.T.

Acknowledgments: This work is supported by HEC Pakistan under grant: 112-25237-2PSI-127.

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

## References

- 1. Gurtin, M.E.; MacCamy, R.C. On the diffusion of biological populations. *Math. Biosci.* **1977**, *33*, 35–49. [CrossRef]
- 2. Gurney, W.; Nisbet, R. The regulation of inhomogeneous populations. *J. Theor. Biol.* **1975**, *52*, 441–457. [CrossRef]
- 3. Lu, Y.-G. Hölder estimates of solutions of biological population equations. *Appl. Math. Lett.* **2000**, *13*, 123–126. [CrossRef]
- 4. Shakeri, F.; Dehghan, M. Numerical solution of a biological population model using He's variational iteration method. *Comput. Math. Appl.* **2007**, *54*, 1197–1209. [CrossRef]
- 5. Zhang, L.; Deng, Y.; Liew, K. An improved element-free Galerkin method for numerical modeling of the biological population problems. *Eng. Anal. Bound. Elem.* **2014**, *40*, 181–188. [CrossRef]
- 6. Cheng, R.; Zhang, L.; Liew, K.-M. Modeling of biological population problems using the element-free kp-Ritz method. *Appl. Math. Comput.* **2014**, 227, 274–290. [CrossRef]
- 7. Golbabai, A.; Seifollahi, S. Numerical solution of the second kind integral equations using radial basis function networks. *Appl. Math. Comput.* **2006**, 174, 877–883. [CrossRef]
- 8. Golbabai, A.; Seifollahi, S. Radial basis function networks in the numerical solution of linear integro-differential equations. *Appl. Math. Comput.* **2007**, *188*, 427–432. [CrossRef]
- 9. Chen, W. New RBF collocation methods and kernel RBF with applications. In *Meshfree Methods for Partial Differential Equations*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 75–86.
- 10. Monaghan, J.J. An introduction to SPH. Comput. Phys. Commun. 1988, 48, 89–96. [CrossRef]
- 11. Gu, L. Moving kriging interpolation and element-free Galerkin method. *Int. J. Numer. Methods Eng.* **2003**, 56, 1–11. [CrossRef]
- 12. Atluri, S.N.; Zhu, T. A new meshless local Petrov–Galerkin (MLPG) approach in computational mechanics. *Comput. Mech.* **1998**, 22, 117–127. [CrossRef]
- 13. Hardy, R.L. Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **1971**, *76*, 1905–1915. [CrossRef]
- 14. Duchon, J. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. In *Constructive Theory of Functions of Several Variables*; Springer: Berlin/Heidelberg, Germany, 1977; pp. 85–100.
- 15. 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. [CrossRef]
- 16. Buhmann, M.D. *Radial Basis Functions: Theory and Implementations;* Cambridge University Press: Cambridge, UK, 2003; Volume 12.
- 17. Wendland, H. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.* **1995**, *4*, 389–396. [CrossRef]
- 18. Fasshauer, G.E. Solving partial differential equations by collocation with radial basis functions. In *Proceedings of Chamonix;* Vanderbilt University Press: Nashville, TN, USA, 1996; Volume 1997, pp. 1–8.
- 19. Fasshauer, G.E. Solving differential equations with radial basis functions: Multilevel methods and smoothing. *Adv. Comput. Math.* **1999**, *11*, 139–159. [CrossRef]

- 20. Brown, D.; Ling, L.; Kansa, E.; Levesley, J. On approximate cardinal preconditioning methods for solving PDEs with radial basis functions. *Eng. Anal. Bound. Elem.* **2005**, *29*, 343–353. [CrossRef]
- 21. Uddin, M.; Minullah, Z.; Ali, A. On the approximation of rapidly oscillatory Hankel transform via radial kernels. *Dolomites Res. Notes Approx.* **2018**, *11*, 1.
- 22. Uddin, M.; Ali, H.; Ali, A. Kernel-based local meshless method for solving multi-dimensional wave equations in irregular domain. *Comput. Model. Eng. Sci.* 2015, 107, 463–479.
- 23. Uddin, M.; Ali, H. The space–time kernel-based numerical method for Burgers' equations. *Mathematics* **2018**, *6*, 212. [CrossRef]
- 24. Bianca, C.; Pappalardo, F.; Pennisi, M.; Ragusa, M. Persistence analysis in a Kolmogorov-type model for cancer-immune system competition. *AIP Conf. Proc.* **2013**, *1558*, 1797–1800.
- 25. Barbagallo, A.; Ragusa, M.A. On Lagrange duality theory for dynamics vaccination games. *Ricerche di Matematica* **2018**, *67*, 969–982. [CrossRef]
- 26. Schaback, R.; Wendland, H. Adaptive greedy techniques for approximate solution of large RBF systems. *Numer. Algorithms* **2000**, *24*, 239–254. [CrossRef]
- 27. Shcherbakov, V.; Larsson, E. Radial basis function partition of unity methods for pricing vanilla basket options. *Comput. Math. Appl.* **2016**, *71*, 185–200. [CrossRef]
- 28. Hon, Y.; Schaback, R.; Zhou, X. An adaptive greedy algorithm for solving large RBF collocation problems. *Numer. Algorithms* **2003**, *32*, 13–25. [CrossRef]
- 29. Kosec, G.; Sarler, B. Local RBF collocation method for Darcy flow. Comput. Model. Eng. Sci. 2008, 25, 197.
- 30. Shu, C.; Ding, H.; Chen, H.; Wang, T. An upwind local RBF-DQ method for simulation of inviscid compressible flows. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 2001–2017. [CrossRef]
- Fan, C.-M.; Chien, C.-S.; Chan, H.-F.; Chiu, C.-L. The local RBF collocation method for solving the double-diffusive natural convection in fluid-saturated porous media. *Int. J. Heat Mass Transf.* 2013, 57, 500–503. [CrossRef]
- 32. Dormand, J.R.; Prince, P.J. A family of embedded Runge–Kutta formulae. J. Comput. Appl. Math. 1980, 6, 19–26. [CrossRef]
- 33. Uddin, M. RBF-PS scheme for solving the equal width equation. *Appl. Math. Comput.* **2013**, 222, 619–631. [CrossRef]
- 34. Nie, Q.; Zhang, Y.-T.; Zhao, R. Efficient semi-implicit schemes for stiff systems. J. Comput. Phys. 2006, 214, 521–537. [CrossRef]
- 35. Beylkin, G.; Keiser, J.M.; Vozovoi, L. A new class of time discretization schemes for the solution of nonlinear PDEs. *J. Comput. Phys.* **1998**, 147, 362–387. [CrossRef]
- 36. Uddin, M. On the selection of a good value of shape parameter in solving time-dependent partial differential equations using RBF approximation method. *Appl. Math. Model.* **2014**, *38*, 135–144. [CrossRef]
- 37. Schaback, R. Error estimates and condition numbers for radial basis function interpolation. *Adv. Comput. Math.* **1995**, *3*, 251–264. [CrossRef]



© 2019 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/).