

Implementation of modified cubic UAT tension B-spline DQM for Numerical approximation of 1D and 2D Reaction-Diffusion System

Mamta Kapoor,
 Department of Mathematics,
 Lovely Professional University,
 Punjab, India

Varun Joshi,
 Department of Mathematics,
 Lovely Professional University,
 Punjab, India

Received: January 14, 2021. Revised: March 20, 2021. Accepted: March 26, 2021. Published: April 5, 2021.

Abstract—In present paper, a new approach, “modified cubic UAT tension B-spline DQM” has been developed to solve the 1D and 2D Reaction-Diffusion system numerically. The modified cubic UAT tension B-spline is used as basis function, to find the required weighting coefficients. The resulting system of ODE has been solved by SSP-RK43 scheme. The proposed scheme is checked by five test examples.

Keywords— Uniform Algebraic Trigonometric (UAT) tension B-spline, 1D and 2D Reaction-Diffusion system, Differential quadrature method, SSP-RK 43 scheme.

I. INTRODUCTION

1D non-linear Reaction Diffusion system of equations is as follows:

$$u_t = a_1 u_{xx} + h_1(u, v) \quad (1)$$

$$v_t = a_2 v_{xx} + h_2(u, v) \quad (2)$$

with Dirichlet or Neumann Boundary conditions in the computational domain $[a, b]$. Where $u(x, t)$ and $v(x, t)$ are the real valued functions, h_1 and h_2 are the arbitrary constants.

2D Reaction Diffusion Brusselator system of the non-linear system of partial differential equations is as follows

$$u_t = \mu + u^2 v + [\lambda + 1]u + \beta[u_{xx} + u_{yy}] \quad (3)$$

$$v_t = \lambda u - u^2 v + \beta [v_{xx} + v_{yy}] \quad (4)$$

Where $x \in [0, L]$ and $y \in [0, L]$

Where $u(x, y, t)$ and $v(x, y, t)$ are the provided in 2D region R^2 closed by the curve C along with

Initial conditions:

$$u(x, y, 0) = h_1(x, y) \quad (5)$$

and

$$v(x, y, 0) = h_2(x, y) \quad (6)$$

and Neumann Boundary conditions on boundary ∂C are defined by the lines $x = 0, x = L, y = 0$ and $y = L$.

$$u_x(0, y, t) = u_x(L, y, t) = 0, t \geq 0 \quad (7)$$

$$u_y(x, 0, t) = u_y(x, L, t) = 0, t \geq 0 \quad (8)$$

$$v_x(0, y, t) = v_x(L, y, t) = 0, t \geq 0 \quad (9)$$

$$v_y(x, 0, t) = v_y(x, L, t) = 0, t \geq 0 \quad (10)$$

Where λ, μ, β are the given constants. $h_1(x, y), h_2(x, y)$ are the prescribed functions. Formulation of reaction-Diffusion systems has an important role in the study of the traits of chemical and biological processes. Reaction-Diffusion systems are very crucial in the chemical processes, which are related to a series of patterns like chemical exchange reactions, pattern formation in the biological phenomena etc. One of the crucial Reaction-Diffusion model is considered as Brusselator model, which represents the working process of the chemical Reaction-Diffusion system along with non-linear oscillatory pattern.

Brusselator model was developed by Brusselator school of Prigogine [1-4]. Basically, the analytic solutions are cumbersome to find for such models and therefore it is interesting problem from the numerical aspect. The literature regarding the numerical approximation of reaction-Diffusion systems has enriched a lot in the previous decades. Researchers have paid a lot of attention in the development of better numerical regimes for the solution of Reaction-Diffusion systems, like numerical approximation of reaction-Diffusion Brusselator model by decomposition method [5, 6], FD approach [7], collocation with radial basis function [8], dual-reciprocity boundary element scheme [9], Homotop Perturbation Method [10], Runge-Kutta-Chebyshev method [11] etc. Dehghan and Abbaszadeh [12] gave the variation multi scale element free Galerkin and Local discontinuous Galerkin scheme for getting the solution of 2D Brusselator Reaction-Diffusion system with cross diffusion and without cross diffusion. Gumel et al. [13], Sun et al. [14] Kumar et al. [15] also generated some techniques for solving the Brusselator model. Jiwari et al. [16, 17] elaborated two approaches for 2D Reaction-Diffusion Brusselator model. Jiwari et al. [16,17] confirmed that for the small values of

diffusion coefficient, the steady state solution will converge to the equilibrium point.

Bellman and his associates [18] firstly introduced DQM to solve the differential equations. Thereafter Quan and Chang [19, 20] gave the improvised concept regarding DQM. In DQM, different test functions are used to generate novel regimes, so that the required weighting coefficients can be determined to approximate the partial derivatives of the considered function. B-spline functions, cubic B-spline, modified cubic B-spline, trigonometric B-spline, exponential B-spline, modified exponential B-spline, Sinc function, Lagrange polynomial function, Barycentric Lagrange interpolation function, Legendre polynomial, quartic B-spline, quintic B-spline and many more [21-23, 24, 25] test functions have been used to develop the novel regimes based upon DQM.

In present paper a new method “Modified UAT tension B-spline DQM” is developed to get the numerical approximation of 1D and 2D Reaction-Diffusion systems. In this scheme modified cubic UAT tension B-spline is implemented in the traditional DQM to fetch the weighting coefficients. This method transforms the Reaction-Diffusion system in to the system of ordinary differential equations, which is tackled by using the SSP-RK43 scheme. In order to check the efficiency and accuracy of this scheme, five test problems are considered.

II. NUMERICAL METHODOLOGY

[Modified cubic UAT tension B-spline Differential Quadrature Method]

UAT tension B-spline with order-4 is defined as follows:

$$\begin{aligned}
 & \mathbf{UATB}_{i,4}(\mathbf{x}) = \\
 & \begin{cases} \frac{\delta_{i,3}\delta_{i,2}}{\tau\sin(\tau h)} \left[(x - x_{i-2}) - \frac{\sin[\tau(x-x_{i-2})]}{\tau} \right], & [\mathbf{x}_{i-2}, \mathbf{x}_{i-1}] \\ \delta_{i,3} \left[\frac{\delta_{i,2}}{\tau\sin(\tau h)} \left\{ (x_{i-1} - x_{i-2}) - \frac{\sin[\tau(x_{i-1}-x_{i-2})]}{\tau} \right\} + (x - x_{i-1}) \right. \\ \left. - \frac{\delta_{i,2}}{\tau\sin(\tau h)} \left\{ (x - x_{i-1}) - \frac{1}{\tau} (\sin(\tau(x - x_i)) + \sin(\tau(x_i - x))) \right\} \right. \\ \left. - \frac{\delta_{i+1,2}}{\tau\sin(\tau h)} \left\{ (x - x_{i-1}) - \frac{\sin(\tau(x-x_{i-1}))}{\tau} \right\} \right] \\ \frac{\delta_{i+1,3}\delta_{i+1,2}}{\tau\sin(\tau h)} \left\{ (x - x_{i-1}) - \frac{\sin(\tau(x-x_{i-1}))}{\tau} \right\}, & [\mathbf{x}_{i-1}, \mathbf{x}_i] \\ 1 - \frac{\delta_{i,3}\delta_{i+1,2}}{\tau\sin(\tau h)} \left\{ (x_{i+1} - x) + \frac{\sin(\tau(x-x_{i+1}))}{\tau} \right\} \\ - \delta_{i+1,3} \left[\frac{\delta_{i+1,2}}{\tau\sin(\tau h)} \left\{ (x_i - x_{i-1}) - \frac{\sin(\tau(x_i-x_{i-1}))}{\tau} \right\} \right. \\ \left. + (x - x_i) - \frac{\delta_{i+1,2}}{\tau\sin(\tau h)} \left\{ (x - x_i) - \frac{(\sin(\tau(x-x_{i+1})) - \sin(\tau(x_i-x_{i+1})))}{\tau} \right\} \right. \\ \left. - \frac{\delta_{i+2,2}}{\tau\sin(\tau h)} \left\{ (x - x_i) - \frac{\sin(\tau(x-x_i))}{\tau} \right\} \right], & [\mathbf{x}_i, \mathbf{x}_{i+1}] \\ \frac{\delta_{i+1,3}\delta_{i+2,2}}{\tau\sin(\tau h)} \left[(x_{i+2} - x) + \frac{\sin(\tau(x-x_{i+2}))}{\tau} \right], & [\mathbf{x}_{i+1}, \mathbf{x}_{i+2}] \\ 0, & \text{else} \end{cases} \quad (11)
 \end{aligned}$$

Table I: Value of UAT tension B-spline at different node points.

	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$UATB_i(x)$	0	a_1	a_2	a_3	0
$UATB'_i(x)$	0	a_4	0	a_5	0

In order to improve the results, modified cubic UAT tension B-spline can be implemented, in a way so that the obtained system of matrix will become “Diagonally dominant” [24]. Where by using following set of equations improvised values can be obtained.

$$\begin{aligned}
 MUATB_1(x) &= UATB_1(x) + 2UATB_0(x) \\
 MUATB_2(x) &= UATB_2(x) - UATB_0(x) \\
 MUATB_j(x) &= UATB_j(x), \quad (j = 3, 4, 5, \dots, N - 2) \\
 MUATB_{N-1}(x) &= UATB_{N-1}(x) - UATB_{N+1}(x) \\
 MUATB_N(x) &= UATB_N(x) + 2UATB_{N+1}(x)
 \end{aligned} \quad (12)$$

A. Determination of weighting coefficients by method (UAT tension B-spline based DQM)

$$MUATB_k^{(1)}(x_i) = \sum_{j=1}^n p_{ij}^{(1)} MUATB_k(x_j) \quad (13)$$

(Where $i = 1, 2, 3, \dots, n$) and ($k = 1, 2, 3, \dots, n$).

From above set of equation at grid point x_i and for the values of $k = 1, 2, 3, \dots, n$, following tridiagonal system of algebraic equations will be obtained:

$$\mathbf{A} \vec{p}^{(1)}[i] = \vec{R}[i], \text{ where } i = 1, 2, 3, \dots, n$$

$$\begin{aligned}
 & \mathbf{A} \\
 & = \\
 & \begin{pmatrix} a_2 + 2a_3 & a_3 & & & & \\ a_1 - a_3 & a_2 & a_3 & & & \\ & a_1 & a_2 & a_3 & & \\ & \vdots & & \ddots & & \\ & & & & a_1 & a_2 & a_3 & 0 \\ & & & & \dots & a_1 & a_2 & a_3 - a_1 \\ & & & & & & a_1 & a_2 + 2a_1 \end{pmatrix}
 \end{aligned}$$

$$\vec{p}^{(1)}[i] = \begin{pmatrix} p_{i,1}^{(1)} \\ p_{i,2}^{(1)} \\ p_{i,3}^{(1)} \\ \vdots \\ p_{i,N-1}^{(1)} \\ p_{i,N}^{(1)} \end{pmatrix}$$

and

$$\vec{R}[1] = \begin{pmatrix} 2a_5 \\ a_4 - a_5 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix}$$

$$\vec{R}[2] = \begin{pmatrix} a_5 \\ 0 \\ a_4 \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix},$$

.....

$$\vec{R}[n] = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ a_5 - a_4 \\ 2a_4 \end{pmatrix}$$

So with the help of above-mentioned equations 2nd and higher order weighting coefficients can be easily obtained [26].

$$c_{ij}^{(r)} = r \left[c_{ij}^{(1)} c_{ii}^{(r-1)} - \frac{c_{ij}^{(r-1)}}{x_i - x_j} \right] \text{ for } i \neq j \quad (14)$$

Where $i = 1, 2, 3, \dots, N$ and $r = 2, 3, 4, \dots, N_1 - 1$

$$c_{ii}^{(r)} = - \sum_{j=1, j \neq i}^N c_{ij}^{(r)} \text{ for } i = j \quad (15)$$

Similarly the weighting coefficients $d_{ij}^{(r)}$ for second or higher order derivatives can be obtained by following formula [26],

$$d_{ij}^{(r)} = r \left[d_{ij}^{(1)} d_{ii}^{(r-1)} - \frac{d_{ij}^{(r-1)}}{y_i - y_j} \right] \text{ for } i \neq j \quad (16)$$

Where $i = 1, 2, 3, \dots, N$ and $r = 2, 3, 4, \dots, N_2 - 1$

$$d_{ii}^{(r)} = - \sum_{j=1, j \neq i}^N d_{ij}^{(r)} \text{ for } i = j \quad (17)$$

B. Implementation of scheme:

DQM approximations, in order to evaluate the weighting coefficients are provided as,

$$u_x^{(r)}(x_i) = \sum_{j=1}^n a_{ij}^{(r)} u(x_j) \text{ and } v_x^{(r)}(x_i) = \sum_{j=1}^n a_{ij}^{(r)} v(x_j) \quad (18)$$

$$u_y^{(r)}(x_i) = \sum_{j=1}^n b_{ij}^{(r)} u(x_j) \text{ and } v_y^{(r)}(x_i) = \sum_{j=1}^n b_{ij}^{(r)} v(x_j) \quad (19)$$

$$u_x^{(1)}(x_i) = \sum_{j=1}^n a_{ij}^{(1)} u(x_j) \text{ and } v_x^{(1)}(x_i) = \sum_{j=1}^n a_{ij}^{(1)} v(x_j) \quad (20)$$

$$u_y^{(1)}(x_i) = \sum_{j=1}^n b_{ij}^{(1)} u(x_j) \text{ and } v_y^{(1)}(x_i) = \sum_{j=1}^n b_{ij}^{(1)} v(x_j) \quad (21)$$

$$u_x^{(2)}(x_i) = \sum_{j=1}^n a_{ij}^{(2)} u(x_j) \text{ and } v_x^{(2)}(x_i) = \sum_{j=1}^n a_{ij}^{(2)} v(x_j) \quad (22)$$

$$u_y^{(2)}(x_i) = \sum_{j=1}^n b_{ij}^{(2)} u(x_j) \text{ and } v_y^{(2)}(x_i) = \sum_{j=1}^n b_{ij}^{(2)} v(x_j) \quad (23)$$

C. Implementation over 1D Reaction-Diffusion System:

$$\frac{du_i}{dt} = a_1 \sum_{j=1}^n a_{ij}^{(2)} u(x_j) + h_1(u_i, v_i) \quad (24)$$

$$\frac{dv_i}{dt} = a_1 \sum_{j=1}^n a_{ij}^{(2)} v(x_j) + h_2(u_i, v_i) \quad (25)$$

D. Implementation over 2D Reaction-Diffusion system:

$$\frac{du_{ij}}{dt} = \mu + u_{ij}^2 v_{ij} + (\lambda + 1)u_{ij} + \beta [\sum_{j=1}^n a_{ij}^{(2)} u(x_j) + \sum_{j=1}^n b_{ij}^{(2)} u(x_j)] \quad (26)$$

$$\frac{dv_{ij}}{dt} = \lambda u_{ij} - u_{ij}^2 v_{ij} + \beta [\sum_{j=1}^n a_{ij}^{(2)} v(x_j) + \sum_{j=1}^n b_{ij}^{(2)} v(x_j)] \quad (27)$$

III. NUMERICAL EXAMPLES AND DISCUSSION

In Figures 1 and 2 Numerical approximation of U and V are provided graphically at the mentioned time levels. In Table II, Numerical U and V are provided at the time levels t = 1, 2 and 3 respectively. In Figure 3 and 4, Numerical U and V are provided graphically for N = 101, Δt = 0.0001, τ = 0.01 and k = 0.9. In Figures 5 and 6, Numerical U and V are graphically represented at the time levels t = 10, 20, 30, 40 and 50 respectively. In Table III, Numerical U and V are evaluated at t = 10, 20 and 30. In Figures 7 and 8 Numerical U and V are graphically presented at different time levels. In Table IV, Numerical values of U and V components is considered at t = 1 and t = 2. In Figures 9 and 10, Numerical

profiles of U and V components are provided at mentioned time levels. In Table V, Numerical U and V are given at t = 0.1 and 1.0. In Table VI, Numerical profiles are provided at time levels t = 5 and t = 10 respectively for N = 11, Delta t = 0.0001, $\tau = 1$, $\lambda = 1$, $\mu = 2$, $\beta = 0.002$.

• *1D RD SYSTEM*

Example 1:

Brusselator model [27] was proposed by Brussels school of Prigogine. present model represents the Hypo-theoretical tri-molecular natured reaction, having very important traits in the chemical science area. Problem is defined as follows [28], [29],

$$u_t = a_1 u_{xx} - (b_1 + 1)u + u^2 v + \eta_1 \quad (28)$$

$$v_t = a_2 v_{xx} + b_1 u - u^2 v \quad (29)$$

Computational domain : [0, 1]

Initial conditions:

$$u(x, 0) = 0.5 \text{ and } v(x, 0) = 1 + 5x \quad (30)$$

Boundary conditions:

Natural boundary conditions are considered

$$u(0, t) = 0, u(1, t) = 0, v(0, t) = 0, v(1, t) = 0 \quad (31)$$

Example 2:

Isothermal chemical system is given as [29], [30]:

$$u_t = u_{xx} - uv \quad (32)$$

$$v_t = v_{xx} - kv + uv \quad (33)$$

Computational Domain: [0, 200]

Initial conditions:

$$u(x, 0) = 1 \quad (34)$$

$$v(x, 0) = \exp(-x^2) \quad (35)$$

Boundary conditions:

$$\frac{\partial u}{\partial x}(0, t) = 0, u(200, t) = 1, \frac{\partial v}{\partial x}(0, t) = 0, v(200, t) = 0 \quad (36)$$

• *2D RD SYSTEM:*

Example 3:

In present example, Brusselator system (3) and (4) is considered along with the Neumann boundary conditions (7)-(10) and the following initial conditions [31, 32].

Initial Conditions:

$$u(x, y, 0) = 0.5 + y, v(x, y, 0) = 1 + 5x \quad (37)$$

Example 4:

In this example non-linear PDE related to the Brusselator system (3) and (4) is considered with the Neumann boundary conditions (7)-(10) along with the following initial conditions [32, 33].

Initial Conditions:

$$u(x, y, 0) = 0.5x^2 - \frac{1}{3}x^3, v(x, y, 0) = 0.5y^2 - \frac{1}{3}y^3 \quad (38)$$

Example 5:

Considered Brusselator system (3) and (4) along with the Neumann boundary conditions (7)-(10) with the following initial conditions [32, 34].

Initial Conditions:

$$u(x, y, 0) = 2 + 0.25y, v(x, y, 0) = 1 + 0.8y \quad (39)$$

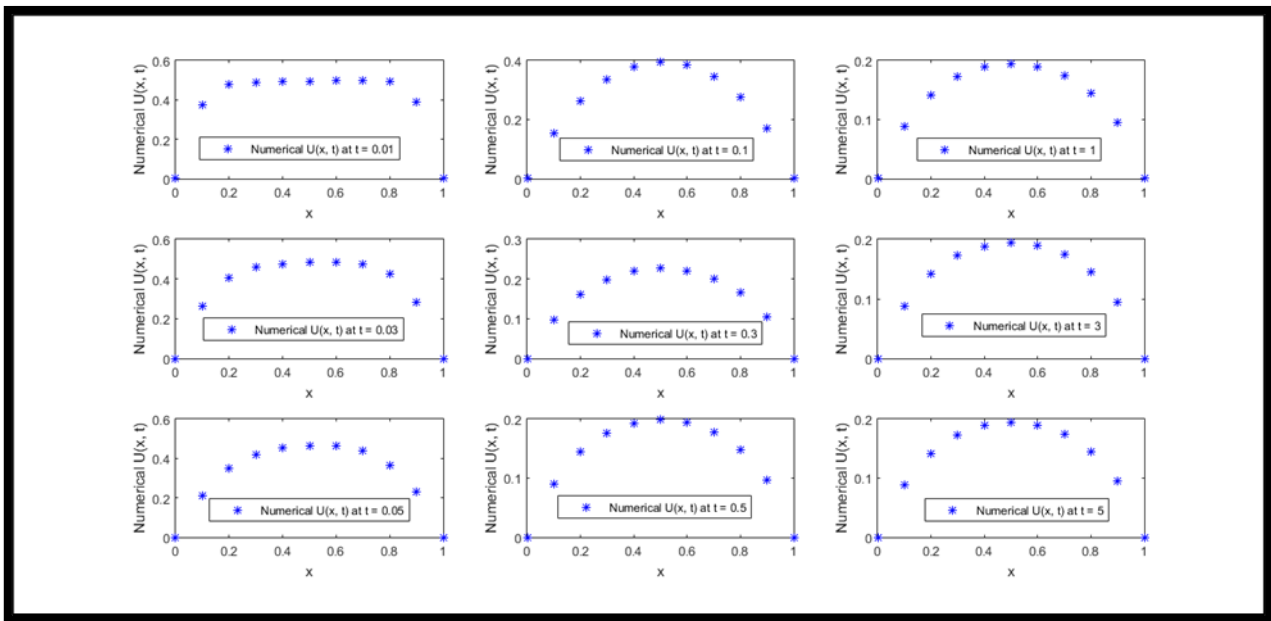


Figure 1: Numerical approximations of $U(x, t)$ at mentioned time levels for $N = 11$, $\tau = 0.01$, $a_1 = 10^{-4}$, $a_2 = 10^{-4}$, $\eta_1 = 1$, $b_1 = 3.4$

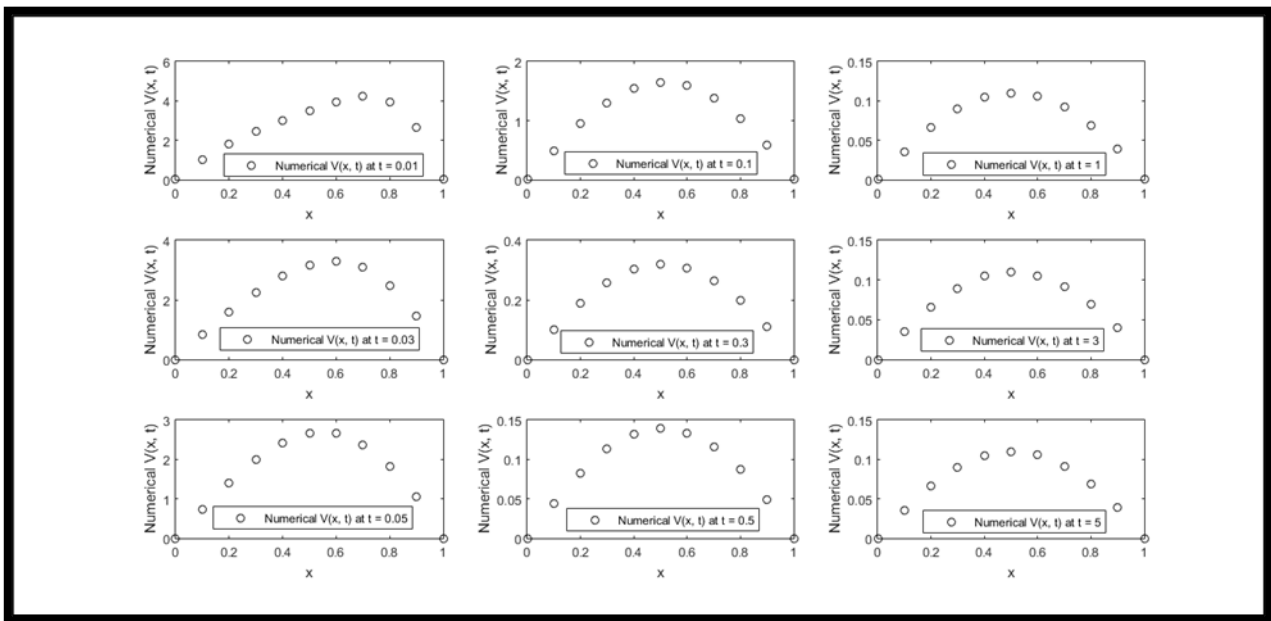


Figure 2: Numerical approximations of $V(x, t)$ at mentioned time levels for $N = 11$, $\tau = 0.01$, $a_1 = 10^{-4}$, $a_2 = 10^{-4}$, $\eta_1 = 1$, $b_1 = 3.4$

Table II: Numerical values of $U(x, t)$ at time levels $t = 1$, $t = 2$ and $t = 3$ respectively for $N = 21$, $\Delta t = 0.0001$, $\tau = 0.001$, $a_1 = a_2 = 10^{-2}$, $\eta_1 = 2$, $b_1 = 3.5$

x	Numerical U	Numerical V	Numerical U	Numerical V	Numerical U	Numerical V
	$t = 1$		$t = 2$		$t = 3$	
0.05	0.1	0.036	0.1	0.036	0.1	0.036

0.1	0.1779	0.0707	0.1778	0.0706	0.1778	0.0706
0.15	0.2376	0.103	0.2376	0.1029	0.2376	0.1029
0.2	0.2832	0.1323	0.2832	0.1322	0.2832	0.1322
0.25	0.3177	0.1579	0.3177	0.1578	0.3177	0.1578
0.3	0.3434	0.1794	0.3433	0.1793	0.3433	0.1793
0.35	0.3618	0.1965	0.3618	0.1963	0.3618	0.1963
0.4	0.3743	0.209	0.3743	0.2088	0.3743	0.2088
0.45	0.3816	0.2166	0.3815	0.2164	0.3815	0.2164
0.5	0.3841	0.2193	0.3841	0.2191	0.3841	0.2191
0.55	0.3821	0.2172	0.382	0.217	0.382	0.217
0.6	0.3753	0.2101	0.3752	0.2099	0.3752	0.2099
0.65	0.3634	0.1982	0.3633	0.198	0.3633	0.198

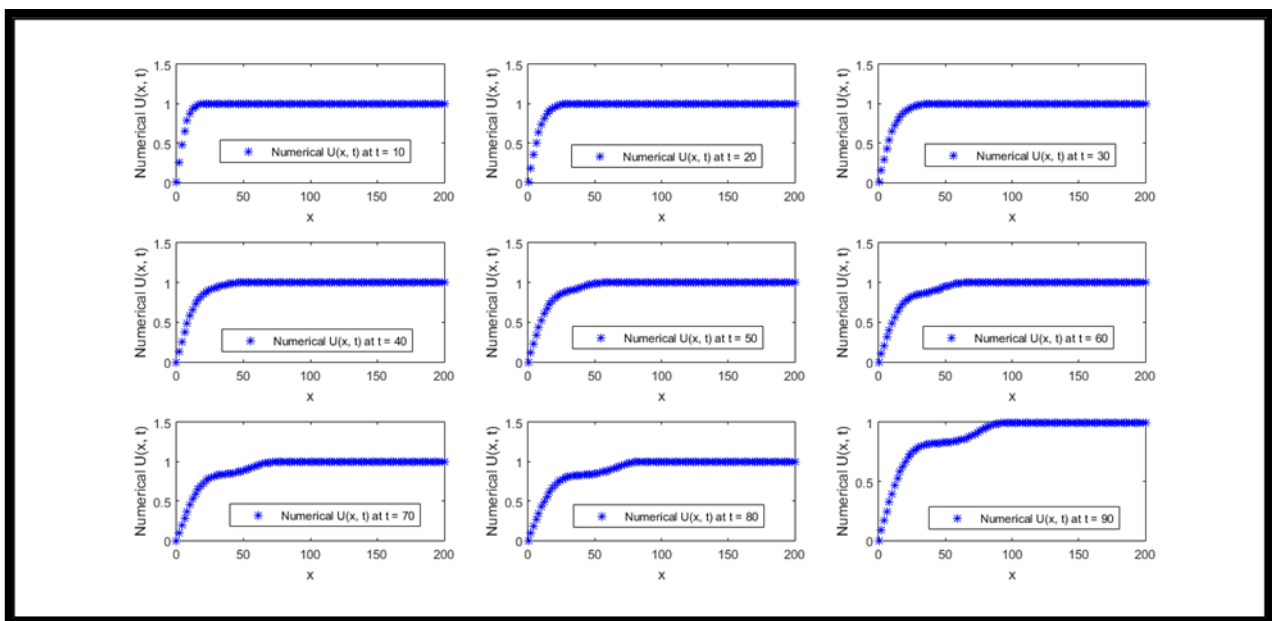


Figure 3: Numerical $U(x, t)$ at the mentioned time levels for $N = 101$, $\Delta t = 0.0001$, $\tau = 0.01$ and $k = 0.9$.

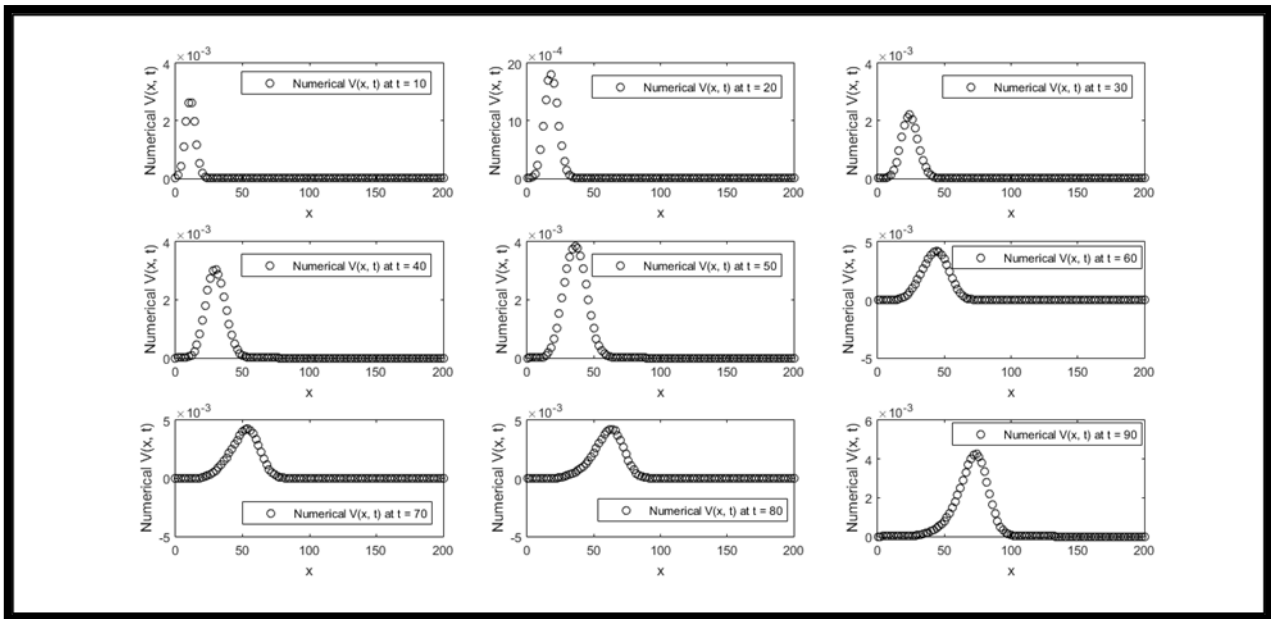


Figure 4: Numerical $V(x, t)$ at the mentioned time levels for $N = 101$, $\Delta t = 0.0001$, $\tau = 0.01$ and $k = 0.9$.

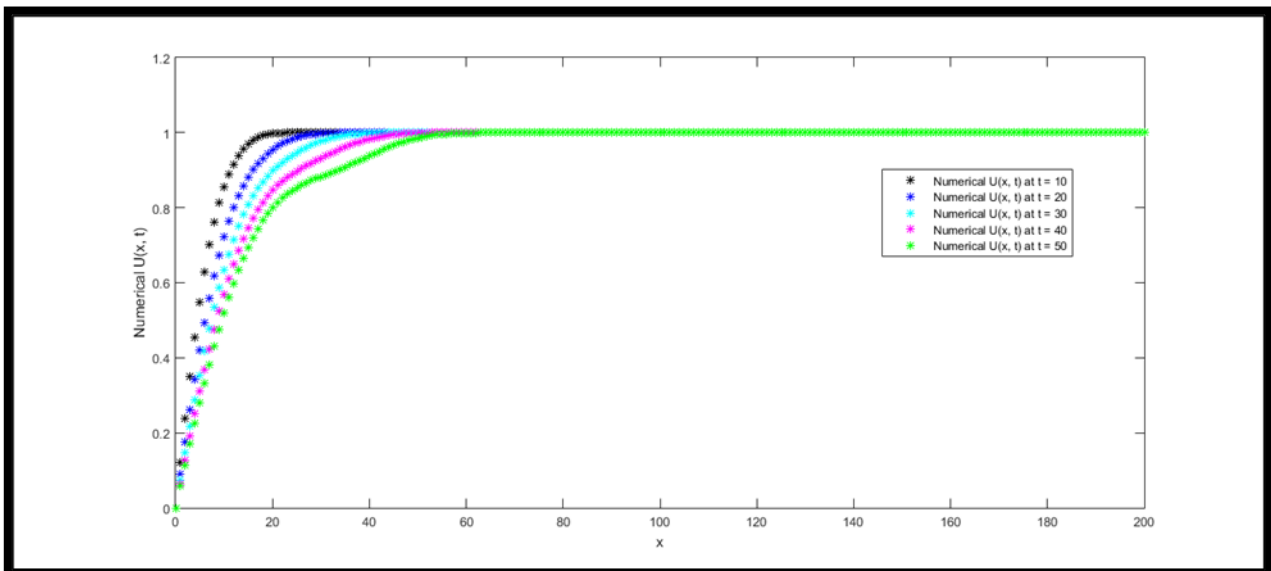


Figure 5: Numerical solutions of $U(x, t)$ at $t = 10, 20, 30, 40$ and 50 respectively for $N = 201$, $\Delta t = 0.0001$, $\tau = 0.01$ and $k = 0.9$

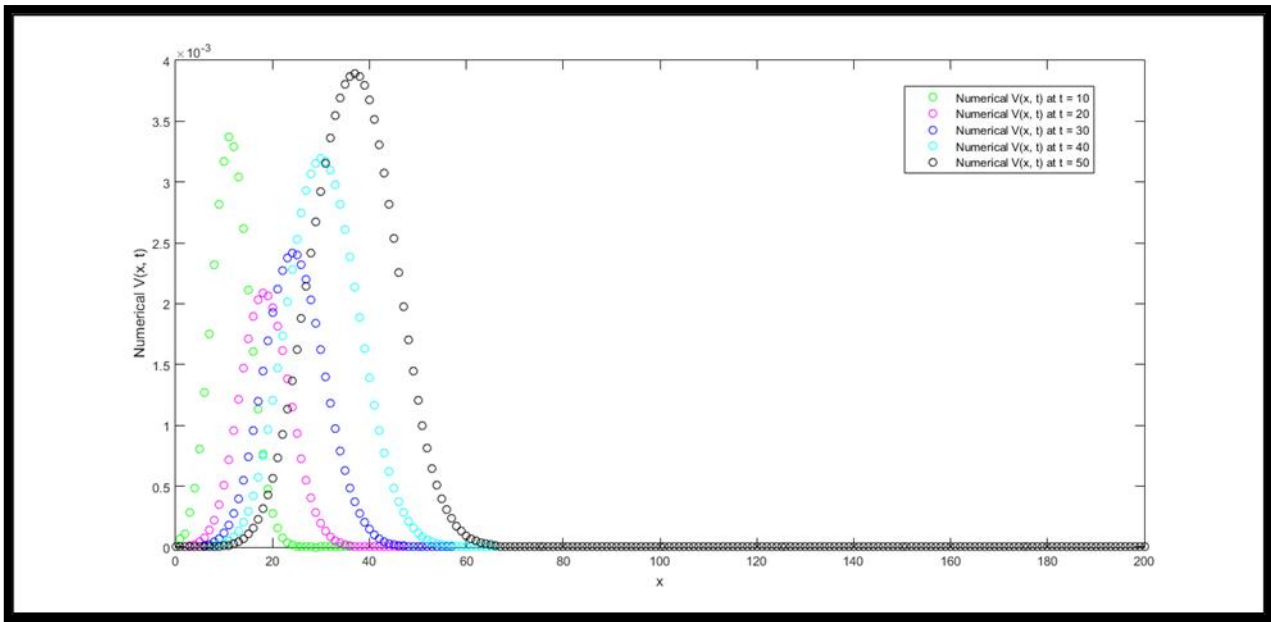


Figure 6: Numerical solutions of $V(x, t)$ at $t = 10, 20, 30, 40$ and 50 respectively for $N = 201, \Delta t = 0.0001, \tau = 0.01$ and $k = 0.9$

Table III: Numerical values of $U(x, t)$ and $V(x, t)$ at time levels $t = 10, 20$ and 30 respectively for $N = 51, \Delta t = 0.0001, \tau = 0.01$ and $k = 0.9$

x	Numerical U	Numerical V	Numerical U	Numerical V	Numerical U	Numerical V
	$t = 10$		$t = 20$		$t = 30$	
4	0.471	0.0004	0.3534	0	0.2941	0
8	0.7769	0.0018	0.6321	0.0002	0.5437	0
12	0.9257	0.0025	0.8123	0.0009	0.7265	0.0003
16	0.9826	0.0012	0.9105	0.0017	0.8416	0.0009
20	0.9978	0.0002	0.9608	0.0016	0.9071	0.0018
24	0.9999	0	0.9857	0.0009	0.9458	0.0022
28	1	0	0.9961	0.0003	0.9711	0.0018

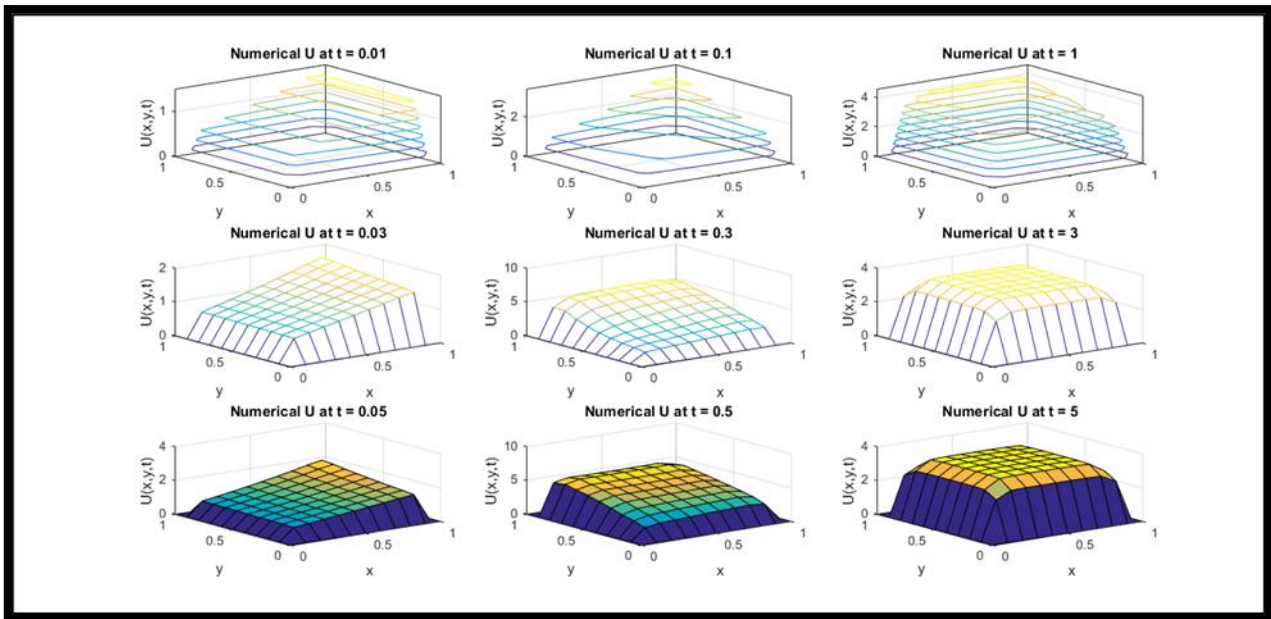


Figure 7: Numerical approximations of $U(x, y, t)$ at the different time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 3.4$ and $\beta = 0.002$

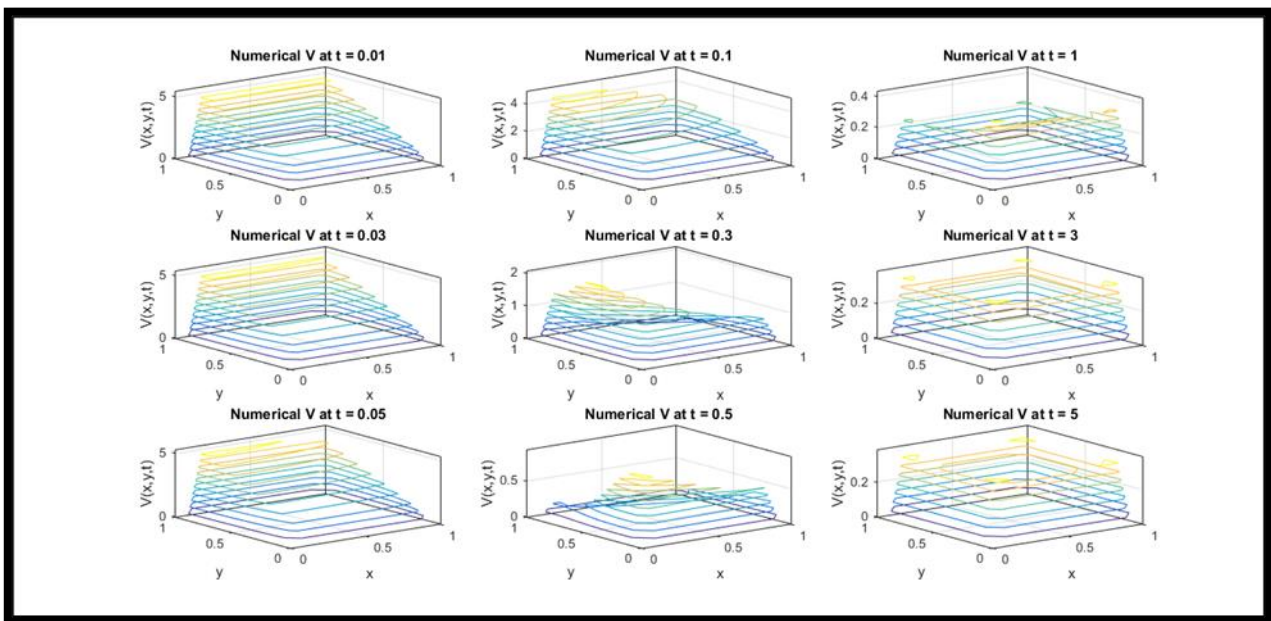


Figure 8: Numerical approximations of $V(x, y, t)$ at the different time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 3.4$ and $\beta = 0.002$

Table IV: Numerical values of $U(x, y, t)$ and $V(x, y, t)$ at time levels $t = 1$ and $t = 2$ respectively for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 3.4$ and $\beta = 0.002$

(x, y)	Numerical U	Numerical V	Numerical U	Numerical V
	$t = 1$		$t = 2$	
$(0.1, 0.3)$	2.763392	0.368878	2.806566	0.344901
$(0.2, 0.3)$	3.364048	0.302191	3.366181	0.2985
$(0.3, 0.5)$	3.597465	0.27674	3.465287	0.287846
$(0.3, 0.7)$	3.623658	0.274304	3.472867	0.286998
$(0.5, 0.7)$	4.020891	0.246011	3.609703	0.275433

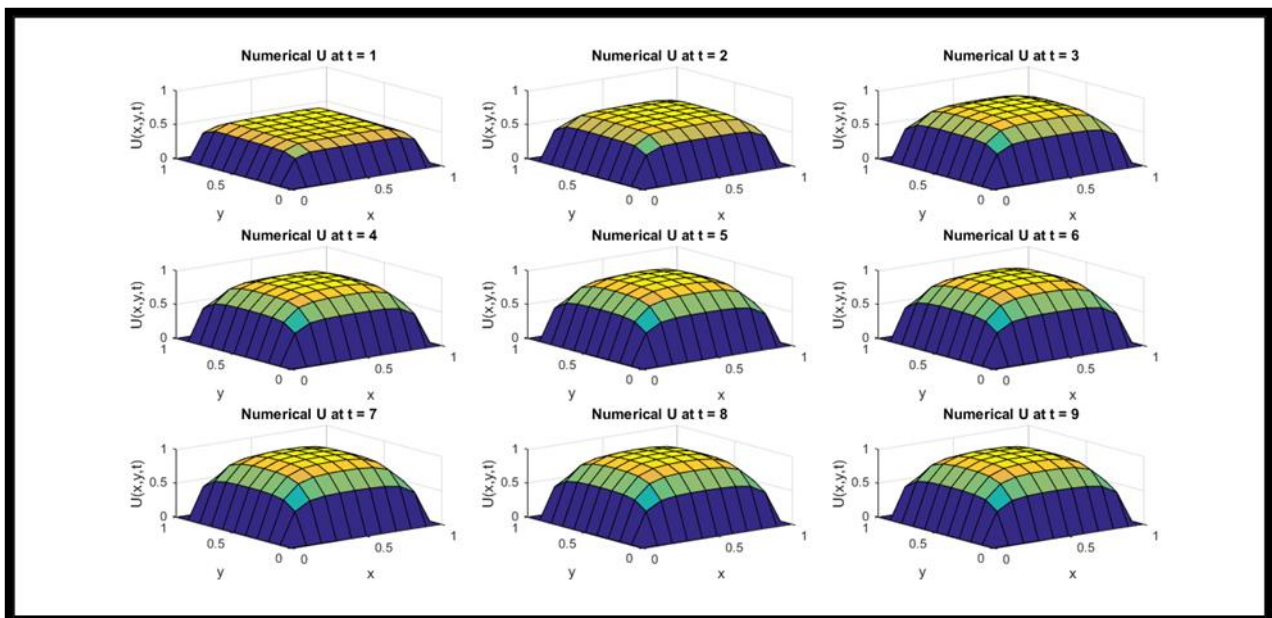


Figure 9: Numerical solutions of $U(x, y, t)$ at different time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 0.0001$, $\lambda = 0.5$, $\mu = 1$ and $\beta = 0.005$

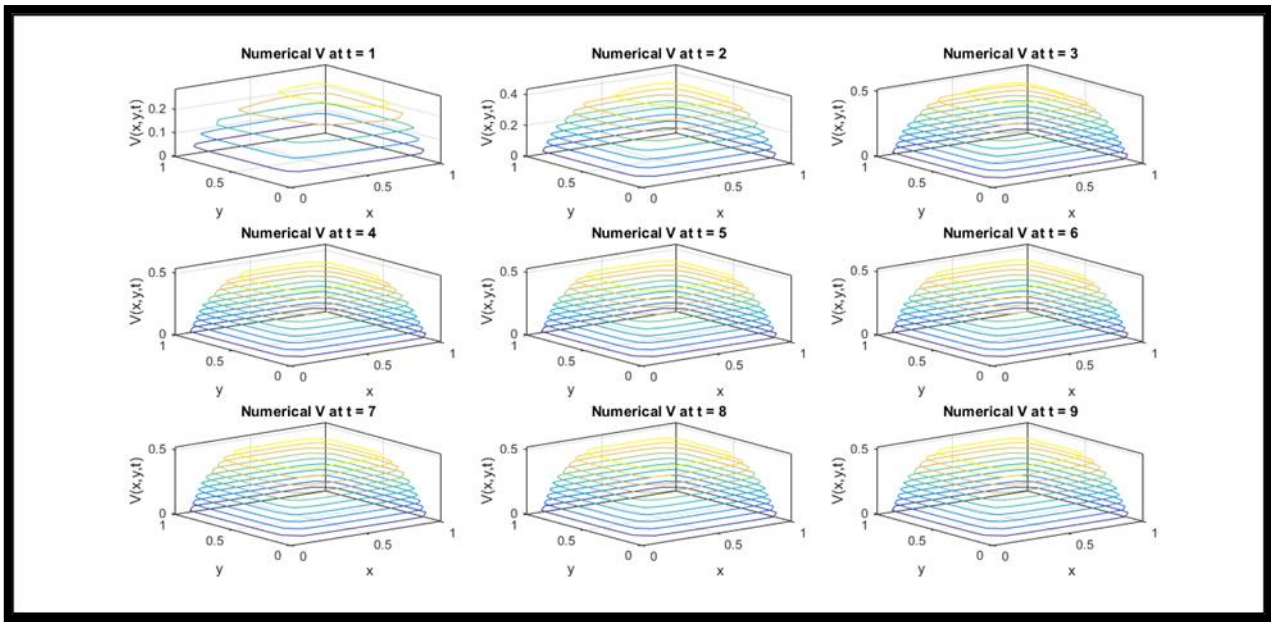


Figure 10: Numerical solutions of $V(x, y, t)$ at different time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 0.0001$, $\lambda = 0.5$, $\mu = 1$ and $\beta = 0.005$

Table V: Numerical approximations of $U(x, y, t)$ and $V(x, y, t)$ at $t = 0.1$ and 1.0 respectively for $N = 11$, $\Delta t = 0.0001$, $\tau = 0.01$, $\lambda = 0.5$, $\mu = 1$ and $\beta = 0.005$

(x, y)	Numerical V		Numerical U	
	$t = 0.1$		$t = 1.0$	
$(0.2, 0.3)$	0.108314	0.039536	0.528485	0.182735
$(0.3, 0.5)$	0.124121	0.087284	0.543129	0.231628
$(0.5, 0.3)$	0.16464	0.042389	0.550896	0.201343
$(0.7, 0.5)$	0.205336	0.091472	0.567449	0.248876
$(0.7, 0.8)$	0.206191	0.157643	0.564561	0.289206

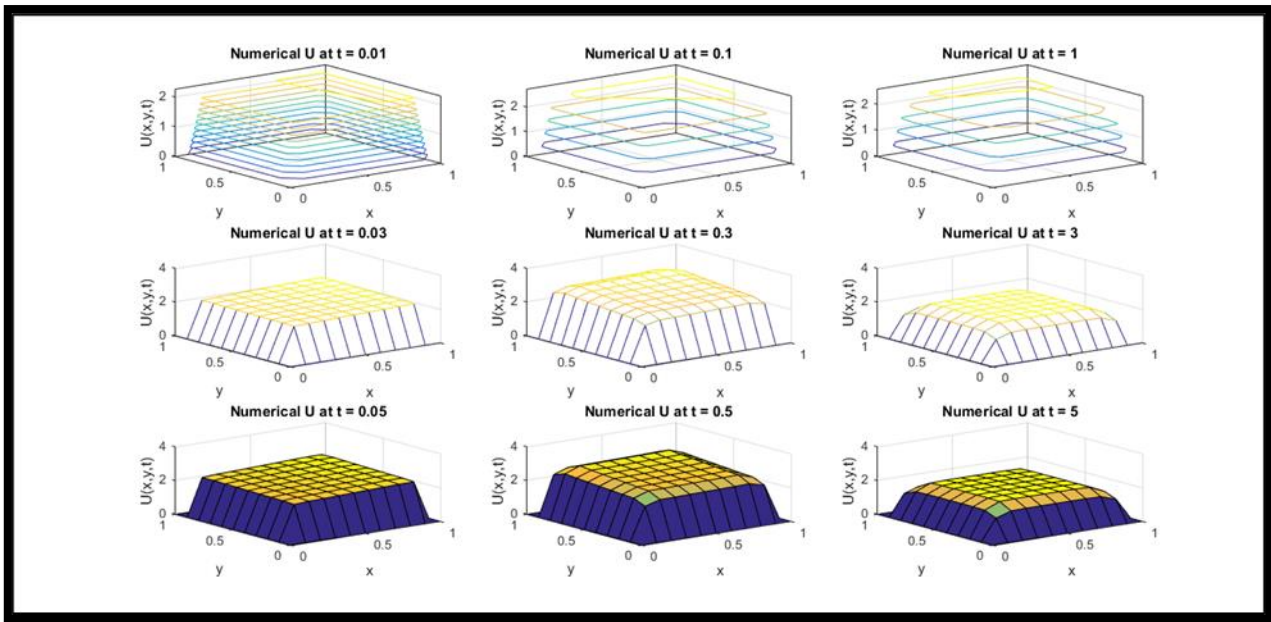


Figure 11: Numerical profiles of $U(x, y, t)$ at the mentioned time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 2$ and $\beta = 0.002$

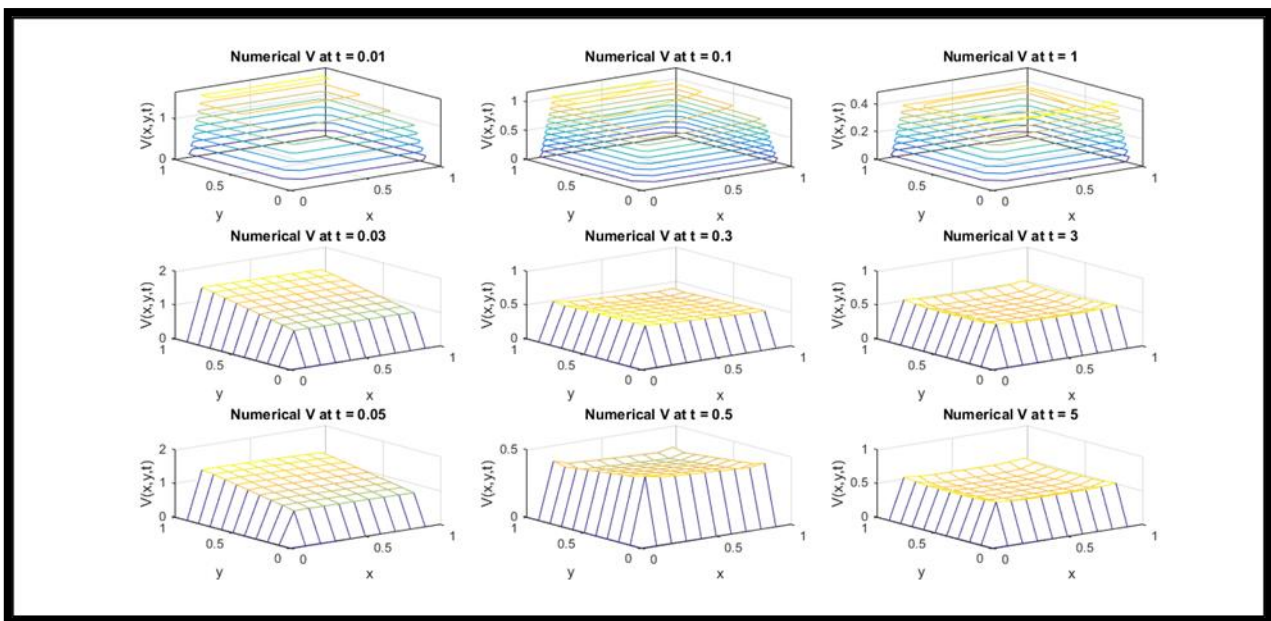


Figure 12: Numerical profiles of $V(x, y, t)$ at the mentioned time levels for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 2$ and $\beta = 0.002$

Table VI: Numerical profiles at time levels $t = 5$ and $t = 10$ respectively for $N = 11$, $\Delta t = 0.0001$, $\tau = 1$, $\lambda = 1$, $\mu = 2$, $\beta = 0.002$

(x, y)	Numerical U	Numerical V	Numerical U	Numerical V
	$t = 5$		$t = 10$	
$(0.1, 0.1)$	1.279069	0.607734	1.279639	0.60771
$(0.3, 0.3)$	1.997259	0.50023	1.99661	0.500535

(0.5, 0.3)	1.999235	0.499812	1.998251	0.500251
(0.5, 0.7)	1.999317	0.499776	1.998251	0.500251
(0.5, 0.8)	1.970645	0.514103	1.970136	0.514418

IV. CONCLUSION

In this paper, a novel regime, “modified cubic UAT tension B-spline DQM” is implemented to solve 1D and 2D Reaction-Diffusion system. Modified cubic UAT tension B-spline is used as the basis function. Obtained system of Ode is solved by SSP-RK43 scheme. Numerical approximation and its graphical results are provided in the form of tables and figures. This scheme is helpful while getting the numerical approximation of complex non-linear partial differential equations, mostly, where analytical solution are not available.

REFERENCES

- [1]. R. Lefever and G. Nicolis(1971). Chemical instabilities and sustained oscillations. *Journal of theoretical Biology*, 30(2), 267-284.
- [2]. G. Nicolis (1977). Self-organization in nonequilibrium systems. *Dissipative Structures to Order through Fluctuations*, 339-426.
- [3]. I. Prigogine and R. Lefever(1968). Symmetry breaking instabilities in dissipative systems. II. *The Journal of Chemical Physics*, 48(4), 1695-1700.
- [4]. J. J. Tyson (1973). Some further studies of nonlinear oscillations in chemical systems. *The Journal of Chemical Physics*, 58(9), 3919-3930.
- [5]. G. Adomian (1995). The diffusion-Brusselator equation. *Computers & Mathematics with Applications*, 29(5), 1-3.
- [6]. A. M. Wazwaz (2000). The decomposition method applied to systems of partial differential equations and to the reaction–diffusion Brusselator model. *Applied mathematics and computation*, 110(2-3), 251-264.
- [7]. E. H. Twizell, A. B. Gumel and Q. Cao (1999). A second-order scheme for the “Brusselator” reaction–diffusion system. *Journal of Mathematical Chemistry*, 26(4), 297-316.
- [8]. A. Ali and S. Haq (2010). A computational modeling of the behavior of the two-dimensional reaction–diffusion Brusselator system. *Applied Mathematical Modelling*, 34(12), 3896-3909.
- [9]. W. T. Ang (2003). The two-dimensional reaction–diffusion Brusselator system: a dual-reciprocity boundary element solution. *Engineering Analysis with Boundary Elements*, 27(9), 897-903.
- [10]. S. Kumar, Y. Khan and A. Yildirim (2012). A mathematical modeling arising in the chemical systems and its approximate numerical solution. *Asia-Pacific Journal of Chemical Engineering*, 7(6), 835-840.
- [11]. J. G. Verwer, W. H. Hundsdorfer and B. P. Sommeijer (1990). Convergence properties of the Runge-Kutta-Chebyshev method. *Numerische Mathematik*, 57(1), 157-178.
- [12]. M. Dehghan and M. Abbaszadeh (2016). Variational multiscale element free Galerkin (VMEFG) and local discontinuous Galerkin (LDG) methods for solving two-dimensional Brusselator reaction–diffusion system with and without cross-diffusion. *Computer Methods in Applied Mechanics and Engineering*, 300, 770-797.
- [13]. A. B. Gumel, W. F. Langford, E. H. Twizell and J. Wu (2000). Numerical solutions for a coupled non-linear oscillator. *Journal of Mathematical Chemistry*, 28(4), 325-340.
- [14]. M. Sun, Y. Tan and L. Chen (2008). Dynamical behaviors of the brusselator system with impulsive input. *Journal of mathematical chemistry*, 44(3), 637-649.
- [15]. S. Kumar, Y. Khan and A. Yildirim (2012). A mathematical modeling arising in the chemical systems and its approximate numerical solution. *Asia-Pacific Journal of Chemical Engineering*, 7(6), 835-840.
- [16]. R. C. Mittal and R. Jiwari (2011). Numerical solution of two-dimensional reaction–diffusion Brusselator system. *Applied mathematics and computation*, 217(12), 5404-5415.
- [17]. R. Jiwari and J. Yuan (2014). A computational modeling of two dimensional reaction–diffusion Brusselator system arising in chemical processes. *Journal of Mathematical Chemistry*, 52(6), 1535-1551.
- [18]. R. Bellman, B. G. Kashef and J. Casti (1972). Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations. *Journal of computational physics*, 10(1), 40-52.
- [19]. J. R. Quan and C. T. Chang (1989). New insights in solving distributed system equations by the quadrature method—I. Analysis. *Computers & Chemical Engineering*, 13(7), 779-788.
- [20]. J. R. Quan and C. T. Chang (1989). New insights in solving distributed system equations by the quadrature method—II. Numerical experiments. *Computers & Chemical Engineering*, 13(9), 1017-1024.
- [21]. A. Korkmaz and İ. Dağ (2011). Shock wave simulations using sinc differential quadrature method. *Engineering Computations*.

- [22]. A. Korkmaz and İ. Dag (2011). Polynomial based differential quadrature method for numerical solution of nonlinear Burgers' equation. *Journal of the Franklin Institute*, 348(10), 2863-2875.
- [23]. A. Korkmaz, A. M. Aksoy and İ. Dag (2011). Quartic B-spline differential quadrature method. *Int. J. Nonlinear Sci*, 11(4), 403-411.
- [24]. G. Arora and B. K. Singh (2013). Numerical solution of Burgers' equation with modified cubic B-spline differential quadrature method. *Applied Mathematics and Computation*, 224, 166-177.
- [25]. H. S. Shukla, M. Tamsir, V. K. Srivastava and J. Kumar (2014). Numerical solution of two dimensional coupled viscous Burger equation using modified cubic B-spline differential quadrature method. *AIP advances*, 4(11), 117134.
- [26]. C. Shu (2012). *Differential quadrature and its application in engineering*. Springer Science & Business Media.
- [27]. I. Prigogine and I. Stengers (1989). *Order out of Chaos*. New York: Bantam, 1984. Also: Nicolis, G. and Prigogine, I. *Exploring Complexity*.
- [28]. P. A. Zegeling and H. P. Kok (2004). Adaptive moving mesh computations for reaction-diffusion systems. *Journal of Computational and Applied Mathematics*, 168(1-2), 519-528.
- [29]. R. C. Mittal and R. Rohila (2016). Numerical simulation of reaction-diffusion systems by modified cubic B-spline differential quadrature method. *Chaos, Solitons & Fractals*, 92, 9-19.
- [30]. C. M. Garcia-Lopez and J. I. Ramos (1996). Linearized Θ -methods part II: Reaction-diffusion equations. *Computer methods in applied mechanics and engineering*, 137(3-4), 357-378.
- [31]. J. G. Verwer, W. H. Hundsdorfer and B. P. Sommeijer (1990). Convergence properties of the Runge-Kutta-Chebyshev method. *Numerische Mathematik*, 57(1), 157-178.
- [32]. R. C. Mittal and R. Jiwari (2011). Numerical solution of two-dimensional reaction-diffusion Brusselator system. *Applied mathematics and computation*, 217(12), 5404-5415.
- [33]. W. T. Ang (2003). The two-dimensional reaction-diffusion Brusselator system: a dual-reciprocity boundary element solution. *Engineering Analysis with Boundary Elements*, 27(9), 897-903.
- [34]. E. H. Twizell, A. B. Gumel and Q. Cao (1999). A second-order scheme for the "Brusselator" reaction-diffusion system. *Journal of Mathematical Chemistry*, 26(4), 297-316.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0
https://creativecommons.org/licenses/by/4.0/deed.en_US