Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 9(5): 13211–13223.
DOI: 10.3934/math.2024644
Received: 14 December 2023
Revised: 25 March 2024
Accepted: 29 March 2024
Published: 09 April 2024

Research article

Solving the reaction-diffusion Brusselator system using Generalized Finite Difference Method

Ángel García[1], **Francisco Ureña**[1,*] **and Antonio M. Vargas**[2]

[1] Departamento de Construcción y Fabricación, UNED, Madrid, Spain
[2] Departamento de Matemáticas Fundamentales, UNED, Madrid, Spain

* **Correspondence:** Email: furenaprieto@gmail.com.

Abstract: In this paper, we investigate the numerical solution of the Brusselator system using a meshless method. A numerical scheme is derived employing the formulas of the Generalized Finite Difference Method, and the convergence of the approximate solution to the exact solution is examined. In order to demonstrate the applicability and accuracy of the method, several examples are proposed.

Keywords: Brusselator; Generalized Finite Differences; meshless method; convergence
Mathematics Subject Classification: 35K57, 65N06

1. Introduction

In this paper we focus on the numerical discretization of the non-linear Brusselator system, which describes oscillatory chemical behavior in autocatalytic reactions . The Brusselator equations read as:

$$\begin{cases} \dfrac{\partial U}{\partial t} = U^2V - 2U + \dfrac{1}{4}\Delta U, & (x, y) \in \Omega,\ t > 0, \\ \dfrac{\partial V}{\partial t} = U - U^2V + \dfrac{1}{4}\Delta V, & (x, y) \in \Omega,\ t > 0, \end{cases} \tag{1.1}$$

where $x \in \Omega$, time $t > 0$, and U and V are the concentrations of the chemical species (the system is also valid to model biological species). Throughout this paper $\Omega \subset \mathbb{R}^2$ is a bounded domain with a regular boundary.

For its importance in biological or chemical processes we consider solving system (1.1) numerically using the meshless method of Generalized Finite Differences. Recently, several numerical methods have been used for solving reaction-diffusion Brusselator system. For instance, in [1] authors used the dual-reciprocity boundary element method (DRBEM) for the numerical solution. Also, a method of lines implemented as a meshless method based on spatial trial spaces spanned by the Newton basis functions have been applied in [2]. A non-standard finite difference method has been applied in [3], weighted residual methods have been used in [4] and the polynomial based differential quadrature method has been considered in [5] for solving the Brusselator system.

By the flexibility of our method due to the easy choice of the clouds of points and the steps of time and space, we derive a discretization of the system by means of the Generalized Finite Difference Method (GFDM) and we prove the convergence of the discrete solution to the analytical one. Several numerical examples on the applications of this meshless method over regular and irregular domains are presented in order to illustrate our results. Very recently, a high order accuracy of time discretization technique has been combined with the generalized finite difference method (see [6] and [7]).

The paper is organized as follows: In Section 2 we recall the analytical foundations of the GFDM and obtain the explicit formulas of the spatial derivatives. Section 3 is devoted to the analytical study of the explicit GFD scheme where we prove the main result of this section, enclosed in Theorem 3.1. In Section 4 we present several numerical examples over regular and irregular domains which show the applicability of the method. Finally, some conclusions are obtained in Section 5.

2. GFDM: Explicit formulae

The aim of this section is to obtain explicit linear expressions for the approximation of partial derivatives in the points (nodes) of the domain. First of all, an irregular grid or cloud of nodes is generated in the domain. For each one of the nodes of the domain, a star is defined as a set of selected nodes $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_s$, where $\mathbf{x}_0$ is denominated central node of star. In order to select the other nodes of star different criteria can be used, see [8,9].

Let $\mathbf{x}_0 = (x_0, y_0)$ be the central node and $h_i = x_i - x_0, k_i = y_i - y_0$, where (x_i, y_i) are the coordinates of the i^{th} node of the star. Let us put $U_0 = U(\mathbf{x}_0)$ and $U_i = U(\mathbf{x}_i)$, then by the Taylor series expansion, for $i = 1, ..., s$, we have

$$U_i = U_0 + h_i \frac{\partial U_0}{\partial x} + k_i \frac{\partial U_0}{\partial y} + \frac{1}{2}\left(h_i^2 \frac{\partial^2 U_0}{\partial x^2} + k_i^2 \frac{\partial^2 U_0}{\partial y^2} + 2h_i k_i \frac{\partial^2 U_0}{\partial x \partial y}\right) + \cdots . \tag{2.1}$$

Let us call

$$\mathbf{c_i}^T = \{h_i, k_i, \frac{h_i^2}{2}, \frac{k_i^2}{2}, h_i k_i\} \tag{2.2}$$

and the spatial derivatives

$$\boldsymbol{D_5}^T = \{\frac{\partial u_0}{\partial x}, \frac{\partial u_0}{\partial y}, \frac{\partial^2 u_0}{\partial x^2}, \frac{\partial^2 u_0}{\partial y^2}, \frac{\partial^2 u_0}{\partial x \partial y}\}. \tag{2.3}$$

If in (2.1) we don't consider the higher than second-order terms, we may obtain a second-order approximation of U_i, which we shall denote by u_i. We may then define:

$$\begin{aligned} B(u) = \sum_{i=1}^{s} [(u_0 - u_i) + h_i \frac{\partial u_0}{\partial x} + k_i \frac{\partial u_0}{\partial y} + \\ + \frac{1}{2}(h_i^2 \frac{\partial^2 u_0}{\partial x^2} + k_i^2 \frac{\partial^2 u_0}{\partial y^2} + 2h_i k_i \frac{\partial^2 u_0}{\partial x \partial y})]^2 \varphi_i^2, \end{aligned} \tag{2.4}$$

where $\varphi_i = \varphi(h_i, k_i)$ are positive symmetrical weighting functions which decrease in magnitude as the distance to the center increases. Some weighting functions such as potentials or exponentials can be used (see [10] for more details).

In order to minimize the error, the norm function given by (2.4) is minimized with respect to the partial derivatives and the following linear system is obtained:

$$\boldsymbol{A}(h_i, k_i, \varphi_i)\boldsymbol{D_5} = \boldsymbol{b}(h_i, k_i, \varphi_i, u_0, u_i). \tag{2.5}$$

By solving (2.5), the partial derivatives can be obtained as a function of the values $(u_0, u_i, h_i, k_i, \varphi_i)$.

Remark 2.1. *Matrix* $\boldsymbol{A}$ *defined in (2.5) is a positive definite matrix (see [10] for a complete proof) and the approximation of the spatial derivatives is of second order* $O(h_i^2, k_i^2)$.

If in accordance with [10] we define

$$\boldsymbol{A}^{-1} = \boldsymbol{Q}\boldsymbol{Q}^T, \tag{2.6}$$

then

$$\boldsymbol{D_5} = \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{b} \tag{2.7}$$

and (2.7) can be rewritten as

$$\boldsymbol{D_5} = -u_0\boldsymbol{Q}\boldsymbol{Q}^T\sum_{i=1}^{s}\varphi_i^2\boldsymbol{c_i} + \boldsymbol{Q}\boldsymbol{Q}^T\sum_{i=1}^{s}u_i\varphi_i^2\boldsymbol{c_i} \tag{2.8}$$

or

$$\boldsymbol{D_5} = \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{\Phi}(\boldsymbol{u} - u_0\mathbf{1}), \tag{2.9}$$

where $\mathbf{1} = \{1, 1, \cdots, 1\}^T$. Thus, spatial derivatives using GFD as in [11] are approximated by

$$\frac{\partial^2 u(\mathbf{x}_0, n\Delta t)}{\partial x^2} + \frac{\partial^2 u(\mathbf{x}_0, n\Delta t)}{\partial y^2} = -m_0 u_0^n + \sum_{i=1}^{s} m_i u_i^n + O(h_i^2, k_i^2), \tag{2.10}$$

Finally, time derivative is approximated as follows

$$\frac{\partial u}{\partial t}(\mathbf{x}_0, n\Delta t) = \frac{u(\mathbf{x}_0, (n+1)\Delta t) - u(\mathbf{x}_0, n\Delta t)}{\Delta t} + O(\Delta t).$$

3. GFD scheme and convergence

Let us consider a bounded domain $\Omega \subset \mathbb{R}^2$. Then, the GFD scheme for system (1.1) is:

$$\begin{cases} \dfrac{u_0^{n+1} - u_0^n}{\Delta t} = (u_0^n)^2 v_0^n - 2u_0^n + \dfrac{1}{4}\left[-m_0 u_0^n + \displaystyle\sum_{i=1}^{s} m_i u_i^n\right] + O(\Delta t, h_i^2, k_i^2), \\ \dfrac{v_0^{n+1} - v_0^n}{\Delta t} = u_0^n - (u_0^n)^2 v_0^n + \dfrac{1}{4}\left[-m_0 u_0^n + \displaystyle\sum_{i=1}^{s} m_i v_i^n\right] + O(\Delta t, h_i^2, k_i^2), \end{cases} \tag{3.1}$$

Thus,

$$\begin{cases} u_0^{n+1} = u_0^n \left[1 - 2\Delta t - \dfrac{m_0 \Delta t}{4} \right] + \Delta t \left[(u_0^n)^2\, v_0^n + \dfrac{\sum\limits_{i=1}^{s} m_i u_i^n}{4} \right] + O(\Delta t, h_i^2, k_i^2), \\ v_0^{n+1} = \Delta t \left[u_0^n - (u_0^n)^2\, v_0^n + \dfrac{\sum\limits_{i=1}^{s} m_i v_i^n}{4} \right] + v_0^n \left[1 - \dfrac{m_0 \Delta t}{4} \right] + O(\Delta t, h_i^2, k_i^2), \end{cases} \tag{3.2}$$

In order to prove the main result of the paper concerning the conditional convergence of the GFD scheme for solving system (1.1), we need the following basic results:

Lemma 3.1. *Let $A \in \mathfrak{M}_{n\times n}(\mathbb{R})$. If there exists some matrix norm such that $\|A\| < 1$, then*

$$\lim_{k\to\infty} A^k = \boldsymbol{0}.$$

Lemma 3.2. *Assuming $A \in \mathfrak{M}_{n\times n}(\mathbb{R})$, then the following are equivalent:*

(i) $\lim\limits_{k\to\infty} A^k = \boldsymbol{0}$,
(ii) $\rho(A) < 1$,

where $\rho(\cdot)$ stands for the spectral radius.

Our main result with respect to the proposed numerical scheme is as stated below

Theorem 3.1. *Let (U, V) be the exact solution to system (1.1). Then, the GFD explicit scheme given by (3.2) is convergent if the following holds:*

$$\Delta t \le \frac{1}{1 + \frac{m_0}{4} - v_0^n \left(u_0^n + U_0^n \right)}. \tag{3.3}$$

Proof.

Notice that, since U, V are the exact solution of system (1.1), they also solve the discrete equation (3.2). Let us denote by U_i^n the exact U-solution at time n and node i (respetively V_i^n). We take the difference between (3.2) and the same expression for the exact solution. We call $\tilde{u}_i^n = u_i^n - U_i^n$ (similarly for $\tilde{v}_i^n$) and notice the following relations:

$$\begin{aligned} (u_0^n)^2\, v_0^n - (U_0^n)^2\, V_0^n = & \\ & (u_0^n)^2\, v_0^n - (U_0^n)^2\, v_0^n + (U_0^n)^2\, v_0^n - (U_0^n)^2\, V_0^n = \\ & \tilde{u}_0^n\, (u_0^n + U_0^n)\, v_0^n + \tilde{v}_0^n\, (U_0^n)^2 . \end{aligned} \tag{3.4}$$

Then,

$$\begin{cases} \tilde{u}_0^{n+1} = \tilde{u}_0^n \left[1 - 2\Delta t - \dfrac{m_0 \Delta t}{4} - \Delta t v_0^n \left(u_0^n + U_0^n \right) \right] + \\ + \Delta t \dfrac{\sum\limits_{i=1}^{s} m_i u_i^n}{4} + \Delta t \tilde{v}_0^n \left(U_0^n \right)^2 + \mathcal{O}(\Delta t, h_i^2, k_i^2), \\ \tilde{v}_0^{n+1} = \Delta t \tilde{u}_0^n \left[1 - v_0^n \left(u_0^n + U_0^n \right) \right] + \tilde{v}_0^n \left[1 - \dfrac{m_0 \Delta t}{4} - \Delta t \left(U_0^n \right)^2 \right] + \\ + \Delta t \dfrac{\sum\limits_{i=1}^{s} m_i \tilde{v}_i^n}{4} + \mathcal{O}(\Delta t, h_i^2, k_i^2), \end{cases} \tag{3.5}$$

Let us call $\tilde{u} = \max_i \{ |\tilde{u}_i^n| \}$ (similarly for $\tilde{v}$). Then, we write (3.5) as

$$\begin{cases} \tilde{u}^{n+1} = \tilde{u}^n \left[\left| 1 - 2\Delta t - \dfrac{m_0 \Delta t}{4} - \Delta t v_0^n \left(u_0^n + U_0^n \right) \right| + \Delta t \dfrac{\sum\limits_{i=1}^{s} |m_i|}{4} \right] + \\ + \Delta t \tilde{v}^n \left(U_0^n \right)^2 + \mathcal{O}(\Delta t, h_i^2, k_i^2), \\ \tilde{v}^{n+1} = \Delta t \tilde{u}^n \left| 1 - v_0^n \left(u_0^n + U_0^n \right) \right| + \\ + \tilde{v}^n \left[\left| 1 - \dfrac{m_0 \Delta t}{4} - \Delta t \left(U_0^n \right)^2 \right| + \Delta t \dfrac{\sum\limits_{i=1}^{s} |m_i|}{4} \right] + \mathcal{O}(\Delta t, h_i^2, k_i^2), \end{cases} \tag{3.6}$$

Let us rewrite (3.6) as

$$\begin{pmatrix} \tilde{u}^{n+1} \\ \tilde{v}^{n+1} \end{pmatrix} \leq \begin{pmatrix} C_1 & \Delta t \left(U_0^n \right)^2 \\ \Delta t \left[\left| 1 - v_0^n \left(u_0^n + U_0^n \right) \right| \right] & C_2 \end{pmatrix} \begin{pmatrix} \tilde{u}^n \\ \tilde{v}^n \end{pmatrix}. \tag{3.7}$$

where

$$C_1 = \left| 1 - 2\Delta t - \frac{m_0 \Delta t}{4} - \Delta t v_0^n \left(u_0^n + U_0^n \right) \right| + \Delta t \frac{\sum\limits_{i=1}^{s} |m_i|}{4}$$

$$C_2 = \left| 1 - \frac{m_0 \Delta t}{4} - \Delta t \left(U_0^n \right)^2 \right| + \Delta t \frac{\sum\limits_{i=1}^{s} |m_i|}{4}.$$

Consider the matrix

$$\mathfrak{A} = \begin{pmatrix} C_1 & \Delta t \left(U_0^n \right)^2 \\ \Delta t \left[\left| 1 - v_0^n \left(u_0^n + U_0^n \right) \right| \right] & C_2 \end{pmatrix}, \tag{3.8}$$

Let us consider the matrix norm $N_1(\mathfrak{A}) = \max_{i=1,2}\{\sum_{j=1}^{2} |a_{ij}|\}$. From their definition it is clear that

$$N_1(\mathfrak{A}) = C_1 + \Delta t\,(U_0^n)^2\,. \tag{3.9}$$

Then,

$$C_1 + \Delta t\,(U_0^n)^2 < 1 \Rightarrow \left|1 - 2\Delta t - \frac{m_0\Delta t}{4} - \Delta t v_0^n\,(u_0^n + U_0^n)\right| < 1 - \Delta t\frac{\sum\limits_{i=1}^{s} |m_i|}{4} - \Delta t\,(U_0^n)^2\,, \tag{3.10}$$

inequality (3.10) is equivalent to

$$\begin{cases} -1 + \Delta t\dfrac{\sum\limits_{i=1}^{s} |m_i|}{4} + \Delta t\,(U_0^n)^2 < 1 - 2\Delta t - \dfrac{m_0\Delta t}{4} - \Delta t v_0^n\,(u_0^n + U_0^n)\,, \\ 1 - 2\Delta t - \dfrac{m_0\Delta t}{4} - \Delta t v_0^n\,(u_0^n + U_0^n) < 1 - \Delta t\dfrac{\sum\limits_{i=1}^{s} |m_i|}{4} - \Delta t\,(U_0^n)^2\,, \end{cases} \tag{3.11}$$

and inequalities (3.11) are equivalent to

$$\begin{cases} \Delta t < \dfrac{1}{1 + \frac{m_0}{4} - v_0^n\left(u_0^n + U_0^n\right)}, \\ \dfrac{\sum\limits_{i=1}^{s} |m_i|}{4} + (U_0^n)^2 + v_0^n\,(u_0^n + U_0^n) < 1 + \dfrac{m_0}{4}. \end{cases} \tag{3.12}$$

Applying Lemma 3.1, we have that $\lim_{k\to\infty} \mathfrak{A}^k = \mathbf{0}$. Now, by Lemma 3.2, this is equivalent to $\rho(\mathfrak{A}) < 1$, that is, the greatest absolute value of all eigenvalues of matrix $\mathfrak{A}$ is bounded by 1, which implies the convergence of the explicit scheme under the condition (3.3). □

4. Results

In this section we illustrate the application of the GFDM for solving the rection-diffusion Brusselator system given by (1.1). We test the method using the regular and irregular clouds of points of Figure 1.

In all the cases considered, the errors has been calculated according to the norms

$$\begin{cases} l_2 = \sqrt{\dfrac{\sum\limits_{i=1}^{NI} \left(sol(i) - exac(i)\right)^2}{NI}} \\ l_\infty = max|sol(i) - exac(i)| \end{cases} \tag{4.1}$$

where *sol*(i) is the GFD solution in node i, *exac*(i) is the exact value of the solution at node i, and NI is the number of nodes of the domain Ω.

For all the numerical results shown in the following sections, we choose Δt as stated, which is to say, the condition of convergence is chosen as the minimum value computated using formulaes Eq (3.3) for each star of the domain Ω. Notice that, therefore, each Δt may vary from one cloud of points to another since its value depends on the star of nodes. We choose $\Delta t = 0.001$, fulfilling the assumption made in Theorem 3.1. The distance criterion has been used, the number of nodes per star is 8 plus the central node and the weighting function is the inverse of the distance squared, $\varphi(d_i) = \frac{1}{d_i^2}$ with d_i being the distance from node i to the central node of the star.

We present two cases with different boundary conditions: Dirichlet and Neumann.

Case 1: Dirichlet boundary conditions

We consider for this first case the clouds of points 1 and 2 of Figure 1.

$$\begin{cases} \dfrac{\partial U}{\partial t} = U^2V - 2U + \dfrac{1}{4}\Delta U, & (x,y) \in \Omega,\ t > 0, \\ \dfrac{\partial V}{\partial t} = U - U^2V + \dfrac{1}{4}\Delta V, & (x,y) \in \Omega,\ t > 0, \\ U(x,y,0) = e^{-(x+y)},\ V(x,y,0) = e^{(x+y)}, & (x,y) \in \Omega. \end{cases} \tag{4.2}$$

By a direct check, the exact solution is (the Dirichlet boundary conditions are chosen so that the equation is fulfilled):

$$U(x,y,t) = e^{-\left(\frac{t}{2}+x+y\right)};\ V(x,y,t) = e^{\left(\frac{t}{2}+x+y\right)} \tag{4.3}$$

Table 1 shows the error norms in the clouds of points.

Table 1. Error norms l_2 and l_∞ for case 1 in clouds of nodes of the Figure 1.

Cloud of nodes	$l_2(U)$	$l_\infty(U)$	$l_2(V)$	$l_\infty(V)$
Cloud 1	6.7004×10^{-6}	1.7156×10^{-5}	8.7630×10^{-5}	1.7048×10^{-4}
Cloud 2	2.6218×10^{-4}	5.4143×10^{-4}	4.6584×10^{-4}	1.4191×10^{-4}

Figures 2 and 3 show the plots of the analytical, (U, V), and approximate, (u, v), solutions of case 1 in the clouds of points 1 and 2.

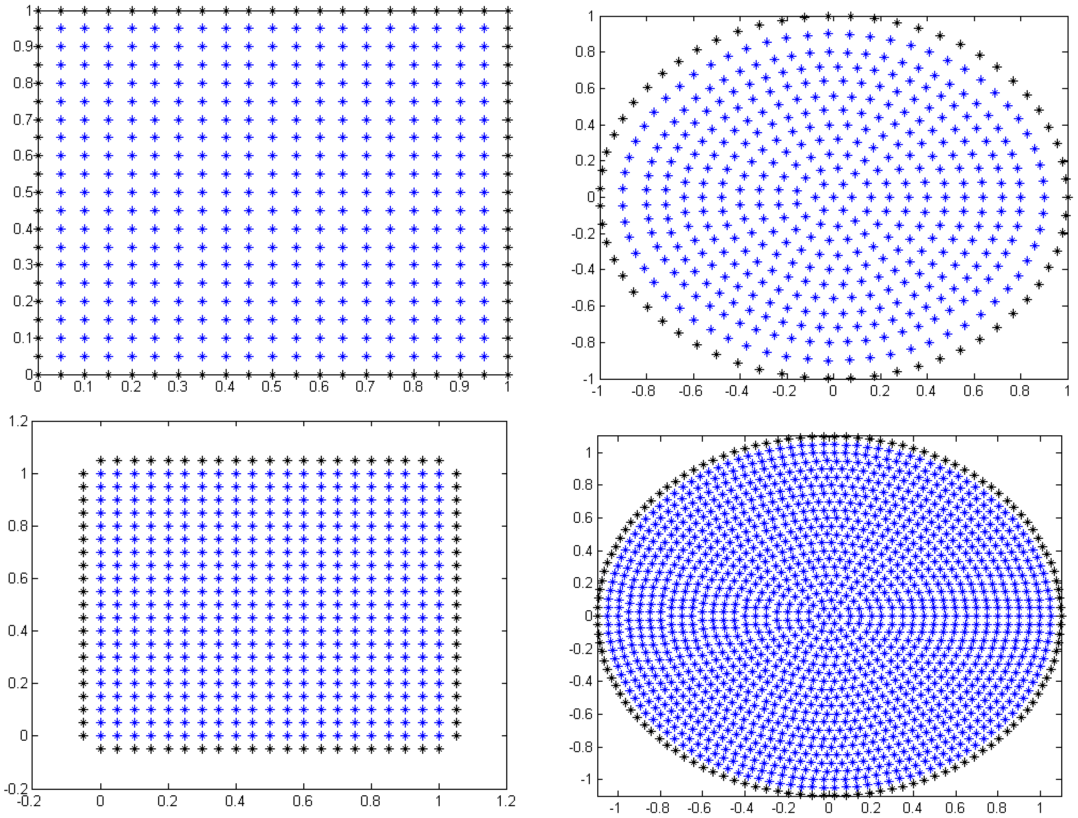

Figure 1. Clouds of points used: 1 to 4.

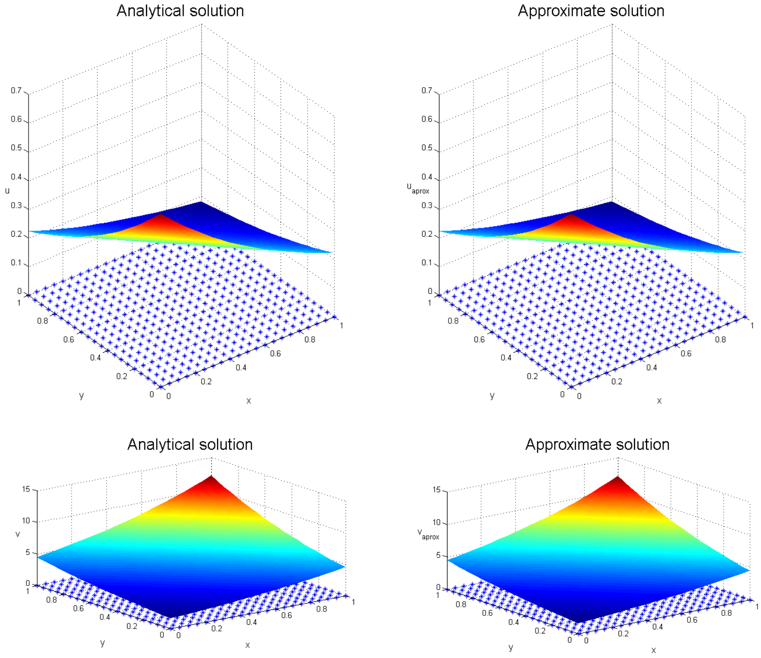

Figure 2. Analytical and approximate solutions of case 1 in the cloud of points 1.

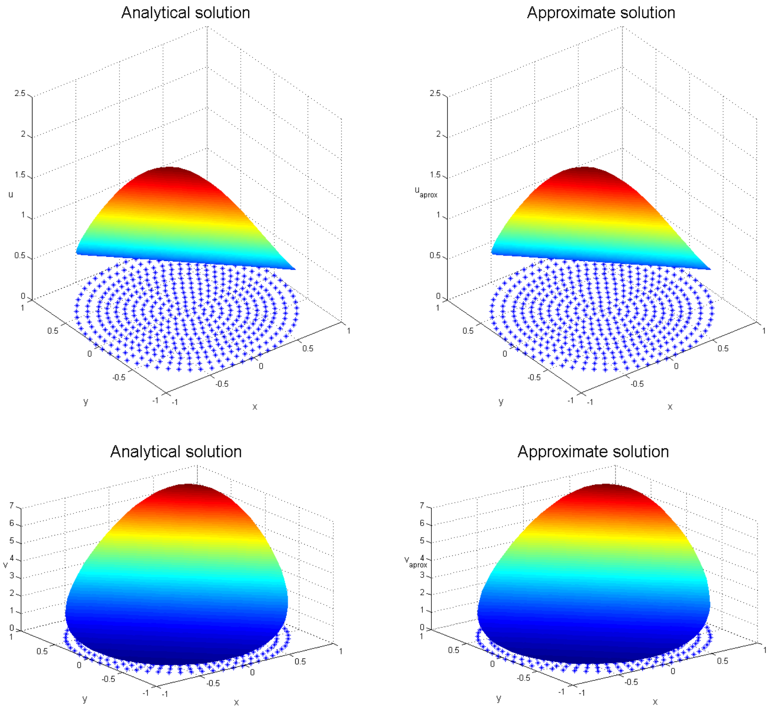

Figure 3. Analytical and approximate solutions of case 1 in the cloud of points 2.

Convergence test

In order to perform a convergence test, we use the clouds of points of Figure 4: 5 (347 nodes), 6 (491 nodes) and 7 (573 nodes). The values of the $l_2(U)$ and $l_2(V)$ errors, and Δt are collected in Table 2.

Table 2. Error norms l_2 and Δt for case 1 in clouds of nodes of Figure 4.

Cloud of nodes	$l_2(U)$	$l_2(V)$	Δt
Cloud 5	2.7430×10^{-4}	3.0255×10^{-4}	$0.001\ s$
Cloud 6	1.6243×10^{-4}	1.9572×10^{-4}	$0.001\ s$
Cloud 7	1.1902×10^{-4}	1.5702×10^{-4}	$0.001\ s$

We plot in Figure 5 the $l_2(U)$ and $l_2(V)$ error norms, respectively, for $T = 1\ s$, $\Delta t = 0.001\ s$ and three clouds of points in Figure 4.

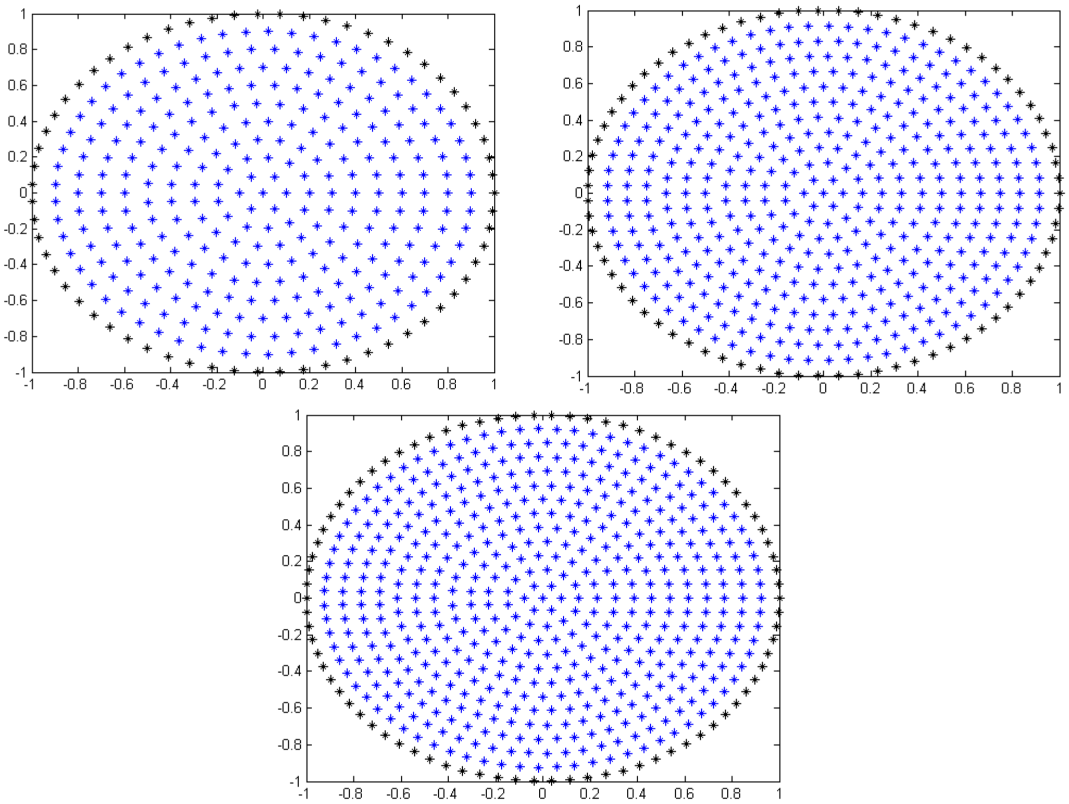

Figure 4. Clouds of points used: 5 to 7.

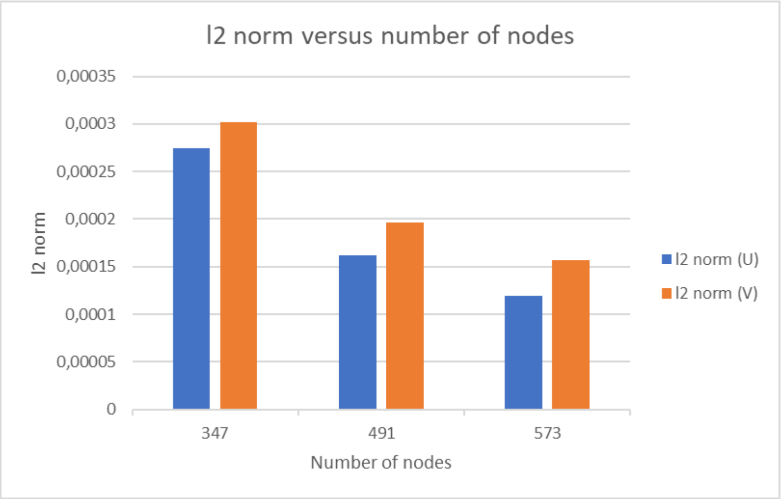

Figure 5. Case 1 on clouds of points 5, 6, and 7 of Figure 5, l_2 error norms U and V versus number of nodes.

Case 2: Neumann boundary conditions

We consider for this first case the clouds of points 3 and 4 of Figure 1.

$$\begin{cases} \dfrac{\partial U}{\partial t} = U^2V - 2U + \dfrac{1}{4}\Delta U, & (x,y) \in \Omega,\ t > 0, \\ \dfrac{\partial V}{\partial t} = U - U^2V + \dfrac{1}{4}\Delta V, & (x,y) \in \Omega,\ t > 0, \\ U(x,y,0) = e^{-(x+y)},\ V(x,y,0) = e^{(x+y)}, & (x,y) \in \Omega. \end{cases} \tag{4.4}$$

By a direct check, the exact solution is (the Neumann boundary conditions are chosen so that the equation is fulfilled):

$$U(x,y,t) = e^{-\left(\frac{t}{2}+x+y\right)};\ V(x,y,t) = e^{\left(\frac{t}{2}+x+y\right)}. \tag{4.5}$$

Table 3 shows the error norms in the clouds of points.

Table 3. Error norms l_2 and l_∞ for case 2 in clouds of nodes of the Figure 1.

Cloud of nodes	$l_2(U)$	$l_\infty(U)$	$l_2(V)$	$l_\infty(V)$
Cloud 3	2.8835×10^{-5}	1.5345×10^{-4}	6.1822×10^{-4}	3.4847×10^{-3}
Cloud 4	2.3402×10^{-4}	1.4582×10^{-3}	5.8804×10^{-4}	4.0817×10^{-3}

Figures 4 and 5 show the plots of the analytical, (U, V), and approximate, (u, v), solutions of case 2 in the clouds of points 3 and 4.

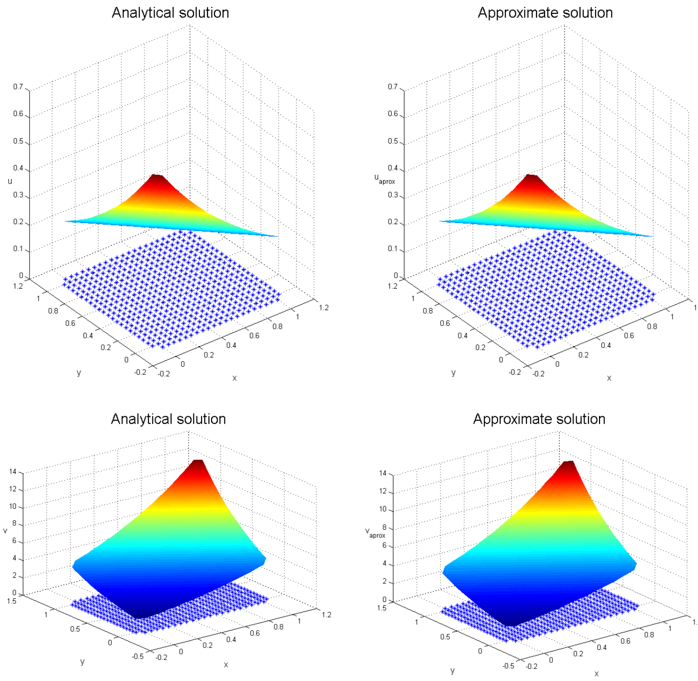

Figure 6. Analytical and approximate solutions of case 2 in the cloud of points 3.

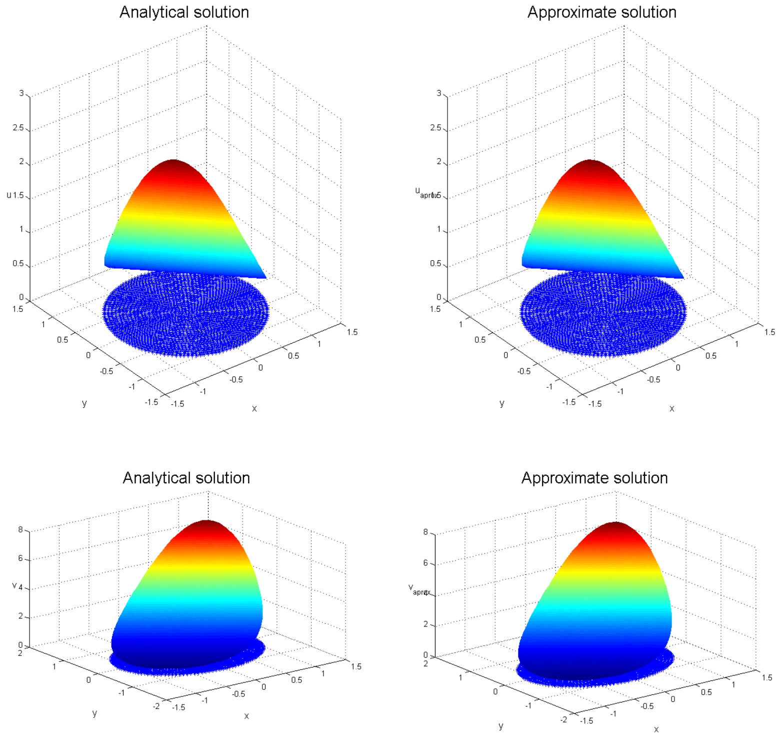

Figure 7. Analytical and approximate solutions of case 2 in the cloud of points 5.

5. Conclusions

We have studied the convergence and obtained the discretization of system (1.1) using a Generalized Finite Difference Method explicit scheme and the order of convergence of the method is shown. The conditional convergence has been obtained for the numerical scheme. We have presented several numerical cases over different domains. The main advantages of the method are its flexibility in discretizing irregular and complex domains and the simplicity in calculations, saving time and computational resources. As a future line of research, there is the possibility of implementing space-time methods, such as STCM, and comparing them with the current method.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence tools in the creation of this article.

Conflict of interest

Prof. Dr. Francisco Ureña is the Guest Editor of special issue "Applications of Partial Differential Equations to Science and Engineering Problems: Numerical Resolution" for AIMS Mathematics. Prof. Dr. Francisco Ureña was not involved in the editorial review and the decision to publish this article.

The authors declare that they have no conflicts of interest.

References

1. A. Whye-Teong, The two-dimensional reaction-diffusion Bruselator system: A dual-reciprocity boundary element solution, *Eng. Anal. Bound. Elem.*, **27** (2003), 897–903. https://doi.org/10.1016/S0955-7997(03)00059-6
2. M. Mohammadi, R. Mokhtart, R. Schaback , A Meshless Method for Solving the 2D Brusselator Reaction-Diffusion System, *CMES*, **101** (2014), 113–138.
3. Z. Zafar, K. Rehan, M. Mushtaq, M. Rafiq, Numerical treatment for nonlinear Brusselator chemical model, *J. Differ. Equ. Appl.*, **23** (2017), 521–538, http://doi.org/10.1080/10236198.2016.1257005
4. R. K. Saeed, S. A. Manaa, F. H. Easif, Numerical Solution of Brusselator Model by Expansion Methods, *Aust. J. Basic Appl. Sci.*, **4** (2010), 3389–3403.
5. R. C. Mittal, R. Jiwari, Numerical solution of two-dimensional reaction-diffusion Brusselator system, *Appl. Math. Comput.*, **217** (2011), 5404–5415. https://doi.org/10.1016/j.amc.2010.12.010
6. W. Sun, W. Qu, Y. Gu, P. W. Li, An arbitrary order numerical framework for transient heat conduction problems, *Int. J. Heat Mass Tran.*, **218** (2024), 124798. https://doi.org/10.1016/j.ijheatmasstransfer.2023.124798
7. W. Sun, H. Ma, W. Qu, A hybrid numerical method for non-linear transient heat conduction problems with temperature-dependent thermal conductivity, *Appl. Math. Lett.*, **146** (2024), 108868. https://doi.org/10.1016/j.aml.2023.108868
8. J. Benito, F. Ureña, L. Gavete, B. Alonso, Application of the Generalized Finite Difference Method to improve the approximated solution of PDEs, *CMES*, **38** (2009), 39–58.
9. J. Benito, F. Ureña, L. Gavete, Influence of several factors in the generalized finite difference method, *Appl. Math. Modell.*, **25** (2001), 1039–1053. https://doi.org/10.1016/S0307-904X(01)00029-4
10. L. Gavete, F. Ureña, J. Benito, A. Garcia, M. Ureña, E. Salete, Solving second order non-linear elliptic partial differential equations using generalized finite difference method, *J. Comput. Appl. Math.*, **318** (2017), 378–387. https://doi.org/10.1016/j.cam.2016.07.025
11. F. Ureña, L. Gavete, A. Garcia, J. Benito, A. M. Vargas, Solving second order non-linear parabolic PDEs using generalized finite difference method (GFDM), *J. Comput. Appl. Math.*, **354** (2019), 221–241. https://doi.org/10.1016/j.cam.2018.02.016

AIMS Press