Hindawi
Complexity
Volume 2020, Article ID 5163704, 15 pages
https://doi.org/10.1155/2020/5163704

*Research Article*

# A Nonoscillatory Second-Order Time-Stepping Procedure for Reaction-Diffusion Equations

**Philku Lee**,[1] **George V. Popescu**,[2,3] **and Seongjai Kim**[1]

[1]*Department of Mathematics and Statistics, Mississippi State University, Starkville, Mississippi State, MS 39762, USA*
[2]*Institute for Genomics, Biocomputing and Biotechnology, Mississippi State University, Starkville, Mississippi State, MS 39762, USA*
[3]*The National Institute for Laser, Plasma & Radiation Physics, 077126 Măgurele, Ilfov, Romania*

Correspondence should be addressed to George V. Popescu; popescu@igbb.msstate.edu

Received 24 October 2019; Accepted 6 March 2020; Published 30 April 2020

Academic Editor: Lingzhong Guo

Copyright © 2020 Philku Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

After a theory of morphogenesis in chemical cells was introduced in the 1950s, much attention had been devoted to the numerical solution of *reaction-diffusion* (RD) *partial differential equations* (PDEs). The *Crank–Nicolson* (CN) method has been a common second-order time-stepping procedure. However, the CN method may introduce spurious oscillations for nonsmooth data unless the time step size is sufficiently small. This article studies a nonoscillatory second-order time-stepping procedure for RD equations, called a *variable-θmethod*, as a perturbation of the CN method. In each time level, the new method detects points of potential oscillations to implicitly resolve the solution there. The proposed time-stepping procedure is nonoscillatory and of a second-order temporal accuracy. Various examples are given to show effectiveness of the method. The article also performs a sensitivity analysis for the numerical solution of biological pattern forming models to conclude that the numerical solution is much more sensitive to the spatial mesh resolution than the temporal one. As the spatial resolution becomes higher for an improved accuracy, the CN method may produce spurious oscillations, while the proposed method results in stable solutions.

## 1. Introduction

As molecular imaging and single cell analysis is advancing our understanding of spatial processes shaping the cellular dynamics, new models of nonlinear dynamics are necessary. Originating in study of organism development, spatial pattern formation has received a large amount of research over the past decade. Among the most studied, the *reaction-diffusion* (RD) systems are generating patterns that have been shown to represent well morphogenesis. A theory of morphogenesis based on a RD model was initially proposed by Turing [1]. Gierer and Meinhardt [2] were the first to explore pattern formation in biological systems using the RD equation. Subsequently, several equations of RD type have been studied to understand patterning in developmental biology. Some were derived from phenomenological models (Gierer–Meinhardt) while other modeled simple reaction schemes (Schnackenberg trimolecular autocatalytic reactions model [3], Gray–Scott model [4], Brusselator model [5], chlorite-iodide-malonic acid, CIMA model [6]). Recent work on RD systems demonstrates that it can be used to understand biological patterns formation [7], while [8] and [9] reviewed RD systems can be used to investigate spatial patterning in developmental systems.

The RD model for biological pattern formation is defined as follows [10]. Let $\Omega$ be a bounded domain in $\mathbb{R}^d$, $d = 1, 2, 3$, $\Gamma = \partial\Omega$ denote the boundary of $\Omega$, and $J = (0, T]$ for some $T > 0$:

$$\begin{aligned} \frac{\partial u}{\partial t} - \mathscr{D}\Delta u &= f(u), \quad && \Omega \times J, \\ \frac{\partial u}{\partial \nu} &= 0, && \Gamma \times J, \\ u &= u^0, && \Omega \times \{t = 0\}, \end{aligned} \tag{1}$$

where $u = [u_1, u_2]^T$, $\mathscr{D} = \mathrm{diag}[D_1, D_2]^T$ is the diffusion tensor, $\Delta$ denotes the Laplace operator, $\partial/\partial\nu$ is the outward normal derivative on the boundary $\Gamma$, and $f(u)$ is the reaction kinetics of the system given as

$$f(u) = \begin{bmatrix} f_1(u_1, u_2) \\ f_2(u_1, u_2) \end{bmatrix}. \tag{2}$$

After Turing proposed a theory of morphogenesis in chemical cells in 1952 [1], much attention has been devoted to the numerical solution of RD problems of form (1); see [11–14] and references therein. Most of the numerical methods studied employed finite difference or finite element approximations for the spatial discretization, while some researchers use finite volume and collocation methods. Once the nonlinear reaction terms are treated (linearized or extrapolated), the *Crank–Nicolson* (CN) method can be applied as a second-order time-stepping procedure. Time-stepping procedures are required at each time step to solve a system of linear algebraic equations, which, although sparse, is compute intensive for multidimensional problems. In order to enhance efficiency of time-stepping procedures, one can adopt the *alternating direction implicit* (ADI) method as in [11–13, 15]. In particular, Fernandes et al. [12] introduce an ADI extrapolated CN orthogonal spline collocation method for RD problems.

ADI was invented as a perturbation of the CN method by Douglas, Peaceman, and Rachford in 1955 [16–18] and has been employed effectively for the calculation of numerical solution of various time-dependent multidimensional problems, either parabolic or hyperbolic [19, 20]. ADI reduces a multidimensional problem to multiple easy-to-solve one-dimensional problems, for an extra cost of a *splitting error* in $\mathrm{O}(\Delta t^2)$, where $\Delta t$ is the time step size. However, the splitting error can be much larger than the sum of spatial and temporal discretization errors, unless the time step size is sufficiently small [21].

On the contrary, the CN method applied for nonsmooth data may introduce spurious oscillations to the numerical solution unless the time step size is sufficiently small to satisfy the maximum principle 12, which has been recognized in the original paper as well [22]. For this reason, whenever a larger time step or a higher spatial resolution is desirable/necessary, the (less accurate, first order) implicit method which is immune to oscillations has been used at least for several initial time steps with nonsmooth initial data [23]. The CN method and its perturbations (such as ADI) must be applied with care when the solution involves fast transitions or sharp edges; in particular, the time step size should be set very small, e.g., $\Delta t = \mathrm{O}(\Delta x^2)$, where $\Delta x$ is the spatial grid size. In order to overcome the oscillation problem of the CN method applied for linear parabolic problems of nonsmooth data, the authors recently suggested a *variable-θmethod* in which the time-stepping parameter of the conventional $\theta$-method, $\theta \in [0, 1]$, was determined based on local oscillatory characteristics of the solution and the data [24].

In this article, we apply the variable-$\theta$ method for the numerical solution of two-component nonlinear RD equations, as given in (1). The variable-$\theta$ method is a perturbation of the CN method which evolves the solution implicitly at points where the solution shows a certain portent of oscillations and maintains as a similar accuracy as the CN method with smooth data. The proposed method would be an adequate choice of time-stepping procedure for the numerical solution of RD *partial differential equations* (PDEs) when a larger time step or a higher spatial mesh resolution is desirable. We have performed a sensitivity analysis for the numerical solution of biological pattern forming models to the spatial and temporal grid sizes. It has been observed from various examples that accuracy of the numerical solution is much more sensitive to the spatial mesh resolution than the temporal one. When the spatial mesh resolution is set high for a higher accuracy, the method allows to keep the temporal resolution moderate or low. The suggested variable-$\theta$ method can result in a smooth/stable numerical solution by suppressing possible oscillations, unlike the CN method.

The article is organized as follows. Section 2 includes a brief review on the CN method and its spurious oscillatory behaviors, as preliminaries. In Section 3, a variable-$\theta$ method is presented for the numerical solution of two-component nonlinear RD equations. We adopt the *successive over-relaxation* (SOR) method to solve the resulting linear systems at each time level. Section 4 considers a heuristic technique for the choice of the optimal relaxation parameter for SOR. Section 5 gives numerical examples that show effectiveness of the variable-$\theta$ method applied to RD problems in 1D and 2D spaces. In Section 6, we summarize our experiments and present conclusions.

## 2. Preliminaries

In this section, we present a brief review of time-stepping procedures, for the numerical solution of *linear* parabolic equations of the form

$$\begin{aligned} \frac{\partial u}{\partial t} - \Delta u &= f, \quad \Omega \times J, \\ \frac{\partial u}{\partial \nu} &= 0, \quad \Gamma \times J, \\ u &= u^0, \quad \Omega \times \{t = 0\}, \end{aligned} \tag{3}$$

where $D > 0$ is a diffusion coefficient and $f$ is a reaction/source term. We also consider difficulties arising when dealing with nonsmooth data (initial values, boundary conditions, and/or the source term).

*2.1. The θ-Method: Difference Equation.* Let $\Omega$ be a rectangular domain in $\mathbb{R}^2$: $\Omega = (a_x, b_x) \times (a_y, b_y)$. By partitioning $\Omega \times J$, we obtain the space-time grid points:

$$\left(x_{ij}, t^n\right) := \left(x_i, y_j, t^n\right), \quad i = 0, 1, \ldots, n_x, j = 0, 1, \ldots, n_y, n = 0, 1, \ldots, n_t, \tag{4}$$

where $n_x$, $n_y$, and $n_t$ are prescribed positive integers and

$$\begin{aligned} x_i &= a_x + i \cdot \Delta x, \\ y_j &= a_y + j \cdot \Delta y, \\ t^n &= n \cdot \Delta t, \\ \Delta x &= \frac{b_x - a_x}{n_x}, \\ \Delta y &= \frac{b_y - a_y}{n_y}, \\ t &= \frac{T}{n_t}. \end{aligned} \tag{5}$$

Define the discrete domain, the set of the spatial grid points, by

$$\Omega_d = \left\{\left(x_i, y_j\right) : 0 \le i \le n_x, 0 \le j \le n_y\right\}, \tag{6}$$

and denote the set of boundary grid points by $\Gamma_d = \Omega_d \cap \Gamma$ and the set of interior grid points by $\Omega_d^0 = \Omega_d \backslash \Gamma_d$.

Let $g_{ij}^n = g(x_{ij}, t^n)$ for all functions $g$ defined in $(x, t)$. Then, the second-order 5-point *finite difference* (FD) approximation $\mathscr{A}$ of $-\Delta$ at $t = t^n$ reads

$$\mathscr{A} u_{ij}^n = A_1 u_{ij}^n + A_2 u_{ij}^n, A_1 u_{ij}^n := -\delta_{xx} u_{ij}^n, A_2 u_{ij}^n := -\delta_{yy} u_{ij}^n, \tag{7}$$

where the FD operators are defined as

$$\begin{aligned} \delta_{xx} u_{ij}^n &= \frac{u_{i-1,j}^n - 2u_{ij}^n + u_{i+1,j}^n}{\Delta x^2}, \\ \delta_{yy} u_{ij}^n &= \frac{u_{i,j-1}^n - 2u_{ij}^n + u_{i,j+1}^n}{\Delta y^2}. \end{aligned} \tag{8}$$

For the temporal derivative $\partial u / \partial t$, a convenient FD approximation can give

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t}. \tag{9}$$

Expressing the spatial derivative by a weighted average $\theta \in [0, 1]$ of previous and current time values, we can formulate the $\theta$-method for (3) as

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} + \mathscr{A}\left[\theta u_{ij}^{n+1} + (1 - \theta) u_{ij}^n\right] = f_{ij}^{n+\theta}, \tag{10}$$

where $f_{ij}^{n+\theta}$ is either $f(x_{ij}, t^{n+\theta})$ or $\theta f_{ij}^{n+1} + (1 - \theta) f_{ij}^n$. A simple algebraic rearrangement of (10) in a vector form becomes

$$(I + \theta \Delta t \mathscr{A}) u^{n+1} = [I - (1 - \theta) \Delta t \mathscr{A}] u^n + \Delta t f^{n+\theta}, \tag{11}$$

where $u^n = [u_{ij}^n]_{0 \le i \le n_x, 0 \le j \le n_y}$ and $f^{n+\theta} = [f_{ij}^{n+\theta}]_{0 \le i \le n_x, 0 \le j \le n_y}$, considered as column vectors. Popular choices of $\theta \in [0, 1]$ are 0, 1, and 1/2, which are, respectively, the explicit method (the forward Euler method), the implicit method (the backward Euler method), and the semi-implicit method (the Crank–Nicolson method).

(i) Forward Euler method: when $\theta = 0$, algorithm (11) is stable when it satisfies

$$\begin{aligned} \mu_x + \mu_y &\le \frac{1}{2}, \\ \mu_x &= \frac{\Delta t}{\Delta x^2}, \\ \mu_y &= \frac{\Delta t}{\Delta y^2}. \end{aligned} \tag{12}$$

Although the explicit method is efficient for each time step, its stability condition in (12) enforces, choosing a small time step size $\Delta t$; it may become less efficient compared with other implicit methods. It is elementary in numerical analysis that when $\theta \ge 1/2$, the $\theta$-method (10) is unconditionally stable.

(ii) Crank–Nicolson method: when $\theta = 1/2$, (11) can be rewritten as

$$\left(I + \frac{\Delta t}{2} \mathscr{A}\right) u^{n+1} = \left(I - \frac{\Delta t}{2} \mathscr{A}\right) u^n + \Delta t f^{n+1/2}. \tag{13}$$

The CN method has been the most popular time-stepping procedure for the numerical solution of parabolic problems because it is stable and of second-order accuracy in both spatial and temporal directions. However, the CN method applied for nonsmooth data may introduce spurious oscillations to the numerical solution unless the algorithm parameters satisfy the maximum principle [22, 23]. As analyzed by the authors [24], the undesired oscillations are due to instability involved in the explicit half step of the CN method, the first term in the right side of (13). The *variable θ-method* proposed in [24] suppresses spurious oscillations, by evolving the solution implicitly ($\theta_{ij} = 1$) at points $x_{ij}$ where the solution shows a certain portent of oscillations or reduced smoothness, and maintains as a similar accuracy as the CN method with smooth data.

(iii) Backward Euler method: when $\theta = 1$, algorithm (11) reads

$$(I + \Delta t \mathscr{A}) u^{n+1} = u^n + \Delta t f^{n+1}. \tag{14}$$

Although the implicit method shows a first-order accuracy in the temporal direction, it never introduces spurious oscillations to its numerical solutions.

*2.2. Numerical Oscillations of the CN Method.* Although the CN method is unconditionally stable and of second-order

accuracy in both spatial and temporal directions, it may introduce spurious oscillations into the numerical solution for nonsmooth data. For simplicity, consider a homogeneous diffusion equation with discontinuous initial values defined in the one-dimensional (1D) space:

$$\begin{aligned} &\frac{\partial u}{\partial t} - u_{xx} = 0, \\ &(x,t) \in (0,1) \times [0,T], \\ &u(0,t) = u(1,t) = 0, \quad t \in [0,T], \\ &u(x,0) = \begin{cases} 0 & \text{if } 0 < x < \frac{1}{4}, \\ 1 & \text{if } \frac{1}{4} \le x < \frac{3}{4}, \\ 0 & \text{if } \frac{3}{4} \le x < 1, \end{cases} \end{aligned} \tag{15}$$

of which the exact solution is given by

$$u(x,t) = \sum_{k=1}^{\infty} \frac{4}{k\pi} \sin\left(\frac{k\pi}{2}\right) \sin\left(\frac{k\pi}{4}\right) \sin(k\pi x) e^{-k^2\pi^2 t}, \tag{16}$$

for $(x,t) \in (0,1) \times [0,T]$.

Figure 1 depicts the exact and numerical solutions evolved by the CN and the variable-$\theta$ method [24], while Figure 2 compares the numerical solutions at $T = 1.0$ obtained by the three numerical methods. The numerical solutions are obtained with $\Delta t = 0.01$ and $\Delta x = \Delta y = 0.025$. As one can see, spurious oscillations occur along with the step discontinuities in the numerical solution of the CN method, as shown in Figure 1(b). On the contrary, as in Figures 1(c) and 2, the variable-$\theta$ method results in an accurate numerical solution without any oscillations. One should notice from Figures 1(a) and 2 that the implicit method is immune to spurious oscillations, but its error is considerably large due to a first-order discretization error in the temporal direction. Although the CN method reveals spurious oscillations, it is quite accurate far from discontinuities. The variable-$\theta$ method results in numerical solutions which are *smooth* as for the implicit method and *accurate* as for the CN method associated with smooth data.

The variable-$\theta$ method is a hybrid time-stepping procedure that is based on the CN method $(\theta = 1/2)$ and alternately using the implicit method $(\theta = 1)$ at points, where the numerical solution shows a certain portent of oscillations or reduced smoothness (the wobble set).

## 3. A Variable-$\theta$ Method for Two-Component Nonlinear Problems

This section introduces an effective time-stepping procedure for the numerical solution of two-component RD problems (1).

*3.1. Linearization through Extrapolation.* Once the spatial derivatives are approximated by second-order finite difference schemes, as in Section 2.1, the semidiscrete problem for (1) is formulated as

$$\begin{aligned} \frac{\partial u}{\partial t} + \mathscr{D}\mathrm{A}u &= f(u), \quad t \in (0,T], \\ u &= u^0, \qquad t = 0. \end{aligned} \tag{17}$$

Let numerical solutions be obtained up to the $n$th time level, $n > 0$. For the numerical solution in the $(n+1)$th level, we first extrapolate numerical solutions in the two previous levels to approximate the solution at the midpoint $t^{n+1/2}$:

$$\tilde{u}^{n+1/2} := \frac{3}{2}u^n - \frac{1}{2}u^{n-1}. \tag{18}$$

See [12], for details of second-order extrapolation schemes for $n \ge 0$. Then, the $\theta$-method for the two-component RD problem reads:

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} + \mathscr{D}\mathscr{A}\left[\theta u_{ij}^{n+1} + (I-\theta)u_{ij}^n\right] = f\left(\tilde{u}^{n+1/2}\right)_{ij}, \tag{19}$$

where $u = [u_1, u_2]^T$, $\mathscr{D} = \mathrm{diag}[D_1, D_2]^T$, and $\theta = \mathrm{diag}[\theta_1, \theta_2]^T$.

The linearized problem (19) can be resolved by solving for two separate components $u^{n+1} = [u_1^{n+1}, u_2^{n+1}]^T$. Each component in (19) can be formulated as follows:

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} + D\mathscr{A}\left[\theta u_{ij}^{n+1} + (I-\theta)u_{ij}^n\right] = f_{ij}^{n+1/2}, \tag{20}$$

where $u$, $D$, and $\theta$ denote, respectively, $u_k$, $D_k$, and $\theta_k$, for $k = 1$ or 2, and $f^{n+1/2}$ is a known source term. The $\theta$-method (20) can be rewritten in a vector form as

$$(I + \theta\Delta t\, D\mathscr{A})u^{n+1} = [I - (1-\theta)\Delta t\, D\mathscr{A}]u^n + \Delta t\, f^{n+1/2}. \tag{21}$$

We present here the main steps of variable-$\theta$ method for a nonoscillatory solution of (20); a complete study of the method for diffusion equation was published in [24].

*3.2. The Variable-$\theta$ Method.* The method begins with defining the *wobble set*, the set of wobble points, as a collection of the grid points where the solution has high fluctuations so that the implicit method $(\theta = 1)$ should be applied for the numerical solution not to develop oscillations.

One can easily verify that numerical oscillations of the CN method occur when its explicit half step produces spurious oscillations. Such nonphysical oscillations may happen particularly when the time step size $\Delta t$ is larger than the stability limit of the explicit scheme. Thus, the wobble set may be formed to include points where the explicit half step of the CN method introduces undesired local extrema. It follows from (21) that the explicit half step of the CN method $(\theta = 1/2)$ reads

$$u^{n,*} := \left(I - \frac{\Delta t}{2} D\mathscr{A}\right)u^n. \tag{22}$$

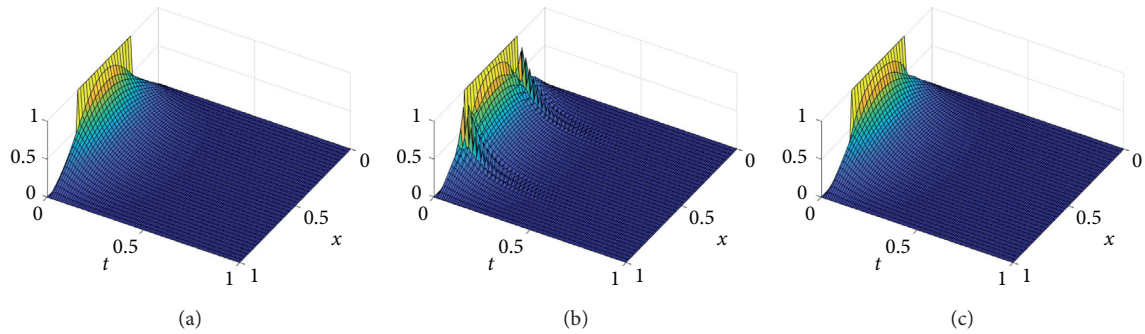



Figure 1: Propagation of the exact and numerical solutions for (15): (a) the exact solution and the numerical solution by (b) the CN method and (c) the variable-$\theta$ method [24], for $0 \le t \le T = 1.0$, when $\Delta t = 0.01$ and $\Delta x = 0.025$.

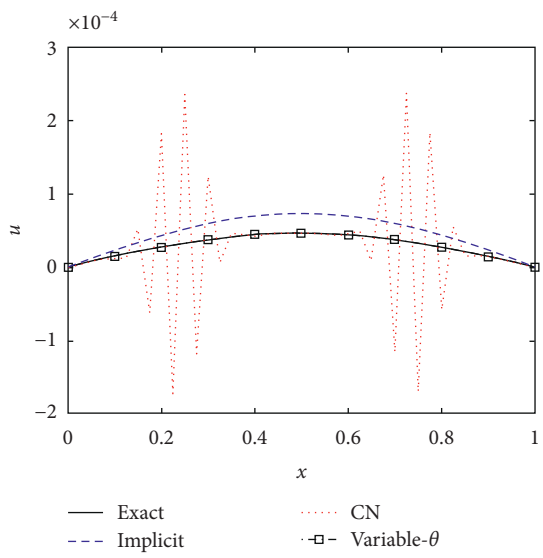



Figure 2: The numerical solutions of (15) at $T = 1.0$, compared with the exact solution.

Let $x_{ij}$ be an interior grid point and consider the four partial directions (made with eight vicinal points of $x_{ij}$): four directions having 0°, 45°, 90°, and 135° from the positive $x$-direction. When spurious oscillations are observed in at least one direction, we select the point $x_{ij}$ as a wobble point.

Define an index function for local extrema (idxt) as

$$\operatorname{idxt}(a,b,c) = \begin{cases} 0, & \text{if } \min(a,c) < b < \max(a,c), \\ 1, & \text{if } b = \max(a,c), \\ -1, & \text{if } b = \min(a,c), \\ 2, & \text{if } \max(a,c) < b, \\ -2, & \text{if } b < \min(a,c). \end{cases} \tag{23}$$

Let $P$, $Q$, and $R$ be point indices and define

$$\operatorname{iswb}(P,Q,R,n) = \begin{cases} 1, & \text{if } \operatorname{idxt}\left(u_P^{n,*}, u_Q^{n,*}, u_R^{n,*}\right) \neq 0 \quad \text{and} \\ & \left|\operatorname{idxt}\left(u_P^{n,*}, u_Q^{n,*}, u_R^{n,*}\right) + \operatorname{idxt}\left(u_P^{n}, u_Q^{n}, u_R^{n}\right)\right| < 4, \\ 0, & \text{otherwise.} \end{cases} \tag{24}$$

Then, the wobble set (for the computation of $u^{n+1} = \left\{u_{ij}^{n+1}\right\}$) is defined as

$$\begin{aligned} W^n = \Big\{ x_{ij} \in \Omega_d^0 \mid \operatorname{iswb}[(i-1,j),(i,j),(i+1,j),n] = 1 \\ \text{or } \operatorname{iswb}[(i-1,j-1),(i,j),(i+1,j+1),n] = 1 \\ \text{or } \operatorname{iswb}[(i,j-1),(i,j),(i,j+1),n] = 1 \\ \text{or } \operatorname{iswb}[(i+1,j-1),(i,j),(i-1,j+1),n] = 1 \Big\}. \end{aligned} \tag{25}$$

*Remark 1.* The function `iswb` selects candidates for the wobble set from local extrema satisfying $\operatorname{idxt}(u_P^{n,*}, u_Q^{n,*}, u_R^{n,*}) \neq 0$; however, the condition

$$\left|\operatorname{idxt}\left(u_P^{n,*}, u_Q^{n,*}, u_R^{n,*}\right) + \operatorname{idxt}\left(u_P^{n}, u_Q^{n}, u_R^{n}\right)\right| < 4, \tag{26}$$

excludes cases where a strict extremum in $u^n$ becomes a strict extremum in the same sense in $u^{n,*}$. Thus, the wobble set (25) is the set of interior grid points $x_{ij}$ where $u_{ij}^{n,*}$ becomes a local extremum while $u_{ij}^n$ is either a nonextreme or an extreme in the opposite sense, for at least one of four partial directions.

Having the wobble set, the parameter $\theta$ for the computation of $u^{n+1}$ can be assigned pointwise:

$$\theta_{ij}^{n+1} := \theta\left(x_{ij}, t^{n+1}\right) = \begin{cases} 1, & \text{if } x_{ij} \in W^n, \\ 1/2, & \text{otherwise.} \end{cases} \tag{27}$$

Thus, the *variable-θmethod* for (20) can be formulated as

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} + D\mathscr{A}\left[\theta_{ij}^{n+1} u_{ij}^{n+1} + \left(1 - \theta_{ij}^{n+1}\right) u_{ij}^n\right] = f_{ij}^{n+1/2}, \tag{28}$$

or, in a vector form after grouping variables:

$$\left(I + \theta^{n+1} \Delta t\, D\mathscr{A}\right) u^{n+1} = \left[I - \left(1 - \theta^{n+1}\right) \Delta t\, D\mathscr{A}\right] u^n + \Delta t\, f^{n+1/2}. \tag{29}$$

The variable-$\theta$ method is analyzed for its numerical stability and accuracy and verified for various examples [24]. It results in nonoscillatory numerical solutions of which the accuracy is *almost* second-order in time.

*Remark 2.* The ADI procedure was also applied to (19) for which the initial values show sharp transitions. It has been

observed that ADI may introduce undesirable/discontinuous features to its solution unless the time step size is sufficiently small, i.e., $\Delta t = \mathrm{O}(\Delta x^2, \Delta y^2)$. The main problem with ADI is that the diffusion becomes anisotropic, i.e., faster in the coordinate directions. When ADI is applied to the variable-$\theta$ formulation of (19), the anisotropic features are reduced significantly. However, it requires to set $\Delta t$ small enough for a reliable numerical solution.

The algebraic system in (29) will be solved by applying the SOR method, with its initial value at the time level $t^{n+1}$ being set as

$$u^{n+1,0} = 2u^n - u^{n-1}. \tag{30}$$

In particular, SOR converges quite fast for an appropriate choice of the relaxation parameter $\omega$.

In the following, we will consider how to tune the optimal relaxation parameter $\widehat{\omega}$ for SOR.

## 4. The Optimal SOR Parameter $\widehat{\omega}$

In this section, we will try to find a relaxation parameter which is *heuristically optimal*. Let us begin with the 2D algebraic system of (21) with $\theta = 1/2$:

$$\mathscr{L}u = r \in \mathbb{R}^{m\times m}, \tag{31}$$

where $\mathscr{L} = I + (\Delta t/2)D\mathscr{A}$, $r = (I - (\Delta t/2)D\mathscr{A})u + \Delta t f^{n+1/2}$, and $m > 0$ is the dimension of the algebraic system. It is known that the optimal relaxation parameter for the SOR method can be determined as ([25], Section 4.3)

$$\widehat{\omega} = \frac{2}{1 + \sqrt{1 - \rho\left(\mathrm{T}_J\right)^2}}, \tag{32}$$

where $\rho(\mathrm{T}_J)$ is the spectral radius of the Jacobi iteration matrix $\mathrm{T}_J$.

For simplicity, assume that the problem is defined on the unit square with a Dirichlet boundary condition. We further assume that the domain is partitioned into $N \times N$ grids so that $h = \Delta x = \Delta y = 1/N$. Then, the eigenvalues of the second-order 5-point FD coefficient matrix A read ([25], Section 6.5)

$$\lambda_{k,\ell}(\mathscr{A}) = \frac{1}{h^2}\left[4 - 2\cos\left(\frac{k\pi}{N}\right) - 2\cos\left(\frac{\ell\pi}{N}\right)\right], \quad 1 \le k, \ell \le N-1, \tag{33}$$

and therefore the eigenvalues of $\mathscr{L}$ can be formulated as

$$\lambda_{k,\ell}(\mathscr{L}) = 1 + \frac{\Delta t}{2}D\lambda_{k,\ell}(\mathscr{A}) = 1 + \frac{D\Delta t}{h^2}\left[2 - \cos\left(\frac{k\pi}{N}\right) - \cos\left(\frac{\ell\pi}{N}\right)\right], \tag{34}$$

for $1 \le k, \ell \le N-1$. Note that the diagonal element of $\mathscr{L}$ is

$$p := 1 + \frac{\Delta t}{2}D\frac{4}{h^2} = 1 + 2\frac{D\Delta t}{h^2}. \tag{35}$$

So, the eigenvalues of the Jacobi iteration matrix $\mathrm{T}_J$ are given as

$$\lambda_{k,\ell}\left(\mathrm{T}_J\right) = \frac{p - \lambda_{k,\ell}(\mathscr{L})}{p} = \frac{D\Delta t}{h^2}\frac{[\cos(k\pi/N) + \cos(\ell\pi/N)]}{(1 + 2(D\Delta t/h^2))}. \tag{36}$$

In order to find the maximum of $|\lambda_{k,\ell}(\mathrm{T}_J)|$, we first obtain

$$\max_k\left|\cos\left(\frac{k\pi}{N}\right)\right| = \cos\left(\frac{\pi}{N}\right) = 1 - c_1h^2, \tag{37}$$

for some $c_1 > 0$. Here, we have used $h = 1/N$ and the approximation $\cos(x) \approx 1 - x^2/2$. Now the spectral radius of $\mathrm{T}_J$ reads

$$\rho\left(\mathrm{T}_J\right) = \max_{k,\ell}\left|\lambda_{k,\ell}\left(\mathrm{T}_J\right)\right| = \left(1 - c_1h^2\right)\left(1 + \frac{1}{2}\frac{h^2}{D\Delta t}\right)^{-1}. \tag{38}$$

Assuming that $c_1h^2 < h^2/(2D\Delta t) < 1$, we finally obtain

$$\rho\left(\mathrm{T}_J\right) = 1 - c_2\frac{h^2}{D\Delta t}, \tag{39}$$

for some $c_2 > 0$.

It follows from (32) and (39) that the optimal SOR parameter $\widehat{\omega}_{\Delta t,h}$, corresponding to the spatial grid size $h$ and the time step size $\Delta t$, can be written as

$$\widehat{\omega}_{\Delta t,h} = \frac{2}{1 + c_0\left(h/\sqrt{D\Delta t}\right)}, \tag{40}$$

for some $c_0 > 0$. The constant $c_0$ can be found *experimentally* from a selected set of $(\Delta t, h)$, as summarized in the following:

$$\begin{cases} \text{(a)Determine } \widehat{\omega}_{\Delta t_0,h_0} \text{ for prescribed grid sizes } (\Delta t_0, h_0)\text{, heuristically.} \\ \text{(b)Solve (40) for } c_0\text{:} \\ c_0 = \dfrac{\sqrt{D\Delta t_0}}{h_0}\left(\dfrac{2}{\widehat{\omega}_{\Delta t_0,h_0}} - 1\right). \end{cases} \tag{41}$$

Once $c_0$ is estimated as in (41), the parameter $\widehat{\omega}_{\Delta t,h}$ in (40) is *near-optimal* for various choices of $(\Delta t, h)$.

## 5. Numerical Experiments

In this section, we present numerical experiments which show effectiveness of the variable-$\theta$ method. The algorithms are implemented in MATLAB and carried out on a desktop computer of Intel Xeon CPU E5-1620 3.60 GHz processor.

To solve the algebraic system at each time level, the SOR method is employed with the near-optimal parameter $\widehat{\omega}$ calculated as in (40), with $c_0$ being estimated with a small grid problem. The SOR iteration is stopped when the maximum difference of consecutive iterates becomes smaller than a tolerance $\varepsilon = 10^{-6}$:

$$\left\|u^{n,k} - u^{n,k-1}\right\|_\infty < \varepsilon. \tag{42}$$

The $L^\infty$-error $E_\infty[t^n]$, measured at $t = t^n$, is defined as follows:

$$E_\infty[t^n] := \left\| u^n - \hat{u}(t^n) \right\|_\infty, \tag{43}$$

where $\hat{u}$ is the exact solution.

*5.1. One-Component RD System.* To investigate accuracy of the variable-$\theta$ method, we consider the diffusion problem (15) studied earlier in Section 2.2. For a comparison purpose, we have implemented not only the $\theta$-methods and the variable-$\theta$ method but also the *implicit predictor-corrector* $(0,2)$-*Padé* (IPC-[0,2]) method [26] and *Lawson and Morris* (LM) local extrapolation method [23].

Table 1 presents the $L^\infty$-error $E_\infty[T]$ at $T = 1.0$ when the five methods are applied for the numerical solution of (15), with various $\Delta t$ and $\Delta x$. As one can see from the table, the CN method resolves its numerical solution poorly (due to oscillations), except for the case the method satisfies the maximum principle. On the contrary, the variable-$\theta$ method results in a second-order accuracy, with its errors being smallest among all the methods for most cases. The new method is a hybrid time-stepping procedure which assembles merits from the CN method (high-accuracy) and the implicit method (smoothness).

Now, consider a nonlinear RD problem of the form

$$\begin{gathered} \frac{\partial u}{\partial t} - u_{xx} = u(1-u), \\ (x,t) \in (0,1) \times [0,T], \end{gathered} \tag{44}$$

with the boundary and initial values, as given in (15).

Figure 3 presents the numerical solutions evolved by the implicit method, the CN method, and the variable-$\theta$ method, when $\Delta t = 0.01$ and $\Delta x = 0.025$ (the mesh is the same as the one selected in Figure 1). Similar to the linear problem in Figure 1, spurious oscillations are introduced into the numerical solution of the nonlinear problem by the CN method only. It should be noticed that spurious oscillations of the CN method are damped out much faster for the nonlinear problem than the linear problem, which is due to the reaction kinetic term $f(u) = u(1-u)$. For the nonlinear problem, It seems that the oscillations at early time steps do not affect the solution at later steps much. This observation explains a partial reason that the second-order CN method has been popular for the numerical solution of PDEs in mathematical biology. However, for other applications, such spurious oscillations at early moments may alter the numerical solution significantly so as for the CN method to be unstable; see Figure 4 below. It is important to develop an effective algorithm which can suppress spurious oscillations for convenient choices of algorithm parameters; the variable-$\theta$ method is effective and stable.

*5.2. Two-Component Nonlinear RD Systems.* Two-component RD systems enable to explain a much wider range of phenomena than their one-component counterparts. Many two-component models have been developed and numerically verified for dynamical patterning behaviors in biology and chemistry. In this section, we consider two-component models interested in the literature of biology and chemistry, to verify effectiveness of the variable-$\theta$ method.

*5.2.1. Gray–Scott Model in 1D.* We apply the numerical methods for the numerical solution of the Gray–Scott model [4, 27] defined as (1) associated with the following reaction kinetics:

$$f(u) = \left[ F(1-u_1) - u_1 u_2^2, u_1 u_2^2 - (F+k) u_2 \right]^T, \tag{45}$$

for any constants $F$ and $k$. Let $\Omega = (0,1)$. We assign two sets of model constants and initial and boundary conditions as follows [28]:

$$\mathscr{D} = \left[ 10^{-4}, 10^{-6} \right]^T, F = 0.035, k = 0.049, \tag{46a}$$

$$u_1(x,0) = 1 - \frac{1}{2} \sin^{100}(\pi x), u_2(x,0) = \frac{1}{4} \sin^{100}(\pi x), x \in (0,1), \tag{46b}$$

$$u_1(0,t) = u_1(1,t) = 1, u_2(0,t) = u_2(1,t) = 0, \quad t \in [0,T], \tag{46c}$$

where (46b) is a *mid-pulse* initial condition, and

$$\mathscr{D} = \left[ 10^{-4}, 5 \cdot 10^{-5} \right]^T, F = 0.025, k = 0.0544, \tag{47a}$$

$$u_1(x,0) = 1 - \frac{1}{2} \cos^{100}\left(\frac{\pi x}{2}\right), u_2(x,0) = \frac{1}{4} \cos^{100}\left(\frac{\pi x}{2}\right), \quad x \in (0,1), \tag{47b}$$

$$\frac{\partial u_1}{\partial x}(0,t) = \frac{\partial u_1}{\partial x}(1,t) = 0, \frac{\partial u_2}{\partial x}(0,t) = \frac{\partial u_2}{\partial x}(1,t) = 0, \quad t \in [0,T], \tag{47c}$$

where (47b) is a *left-pulse* initial condition.

In Figure 5, we present the propagation of the numerical solution of $u_2$ by the variable-$\theta$ method associated with (46a)–(46c) at the grid sizes $\Delta t = 0.01$ and $\Delta x = 0.004$ over $0 \le t \le T = 2000$. The initial midpulse splits in early moments to travel in both directions (self-replication of the pulse). As each of the pulses travels, it becomes thicker (bigger) up to a certain width and begins to replicate itself recursively.

Table 1: $L^\infty$-error $E_\infty[T]$ for (15) at $T = 1.0$.

| $(\Delta t, \Delta x)$ | Implicit | CN | IPC-[0, 2] | LM | Variable-$\theta$ |
|---|---|---|---|---|---|
| $(0.1, 0.05)$ | $9.0 \cdot 10^{-4}$ | $2.4 \cdot 10^{-1}$ | $6.0 \cdot 10^{-5}$ | $8.4 \cdot 10^{-5}$ | $8.4 \cdot 10^{-5}$ |
| $(0.01, 0.05)$ | $2.8 \cdot 10^{-5}$ | $4.8 \cdot 10^{-7}$ | $1.6 \cdot 10^{-6}$ | $3.2 \cdot 10^{-6}$ | $8.5 \cdot 10^{-7}$ |
| $(0.1, 0.025)$ | $8.9 \cdot 10^{-4}$ | $3.7 \cdot 10^{-1}$ | $5.9 \cdot 10^{-5}$ | $8.3 \cdot 10^{-5}$ | $5.1 \cdot 10^{-5}$ |
| $(0.01, 0.025)$ | $2.7 \cdot 10^{-5}$ | $2.0 \cdot 10^{-4}$ | $9.1 \cdot 10^{-7}$ | $2.5 \cdot 10^{-6}$ | $1.5 \cdot 10^{-7}$ |
| $(0.1, 0.0125)$ | $8.9 \cdot 10^{-4}$ | $4.3 \cdot 10^{-1}$ | $5.8 \cdot 10^{-5}$ | $8.3 \cdot 10^{-5}$ | $2.7 \cdot 10^{-4}$ |
| $(0.01, 0.0125)$ | $2.7 \cdot 10^{-5}$ | $3.9 \cdot 10^{-2}$ | $7.5 \cdot 10^{-7}$ | $2.4 \cdot 10^{-6}$ | $1.2 \cdot 10^{-7}$ |

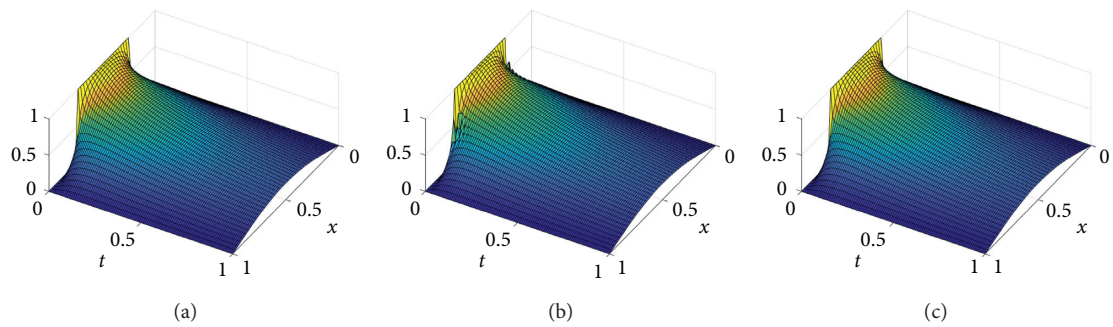



Figure 3: Propagation of the numerical solutions for (44): (a) the implicit method, (b) the CN method, and (c) the variable-$\theta$ method, for $0 \le t \le T = 1.0$, when $\Delta t = 0.01$ and $\Delta x = 0.025$.

Figure 6 depicts the propagation of the numerical solution of $u_2$ by the variable-$\theta$ method associated with (47a)–(47c) over $0 \le t \le T = 5000$, at the same resolution, as in Figure 5. One can clearly observe a traveling pulse which begins from the left edge point and reflects whenever it hits the boundary, due to the no-flux boundary condition (47c).

To investigate bifurcation in the RD system and our method numerical accuracy, we present numerical solutions of the wave-splitting problem (46a)–(46c) obtained with various spatial and temporal grid sizes, as shown in Figure 7. The image $I_{k\ell}$ represents the numerical solution obtained with the mesh resolution $(\Delta t, \Delta x) = (10^{-k}, 0.01/2^{\ell-1})$. For example, the image $I_{23}$ is associated with the mesh resolution $(\Delta t, \Delta x) = (1/100, 1/400)$. One can easily point out from the images that the spatial resolution alters the numerical solution *dramatically* even with halved spatial grid sizes (compare the images horizontally), while the temporal resolution affects *little* the numerical solution even with one-order smaller temporal step sizes (compare them vertically).

The main reason for such a sensitivity to the spatial resolution is that the RD does not have as much time before growing to reach the margins of the mesh in low spatial resolutions (of large $\Delta x$'s). When this happens, the RD pattern typically deteriorates and it does not travel in an appropriate speed nor reaches a condition to replicate itself on time, see ([29], Section 4.2) for similar observations. We summarize the experiments with the Gray–Scott model in 1D as follows:

(i) Accuracy of the numerical solution is much more sensitive to the spatial mesh resolution than the temporal one.

(ii) Thus, it is crucial to set a high spatial resolution (small $\Delta x$'s) for a desirable accuracy.

(iii) As the spatial resolution becomes higher, the CN method may more likely produce spurious oscillations, while the variable-$\theta$ method results in stable solutions.

*5.2.2. Gray–Scott Model in 2D.* Note that the two-component Gray–Scott model is formulated as in (1) with the reaction kinetics $f(u)$ given in (45). We choose problem coefficients as follows [15]:

$$\begin{aligned} \Omega &= (-1, 1) \times (-1, 1), \\ \mathscr{D} &= [0.001, 0.001]^T, \\ F &= 1, \\ k &= 0. \end{aligned} \tag{48}$$

For the purpose of error analysis, we select a smooth solution $\widehat{u} = [\widehat{u}_1, \widehat{u}_2]$ defined as

$$\begin{aligned} \widehat{u}_1(x, y, t) &= \cos(2t)\cos(2\pi x)\cos(\pi y), \\ \widehat{u}_2(x, y, t) &= \cos(2t)\cos(\pi x)\cos(2\pi y), \end{aligned} \tag{49}$$

and replace the reaction kinetics $f(u)$ with $f_{\widehat{u}}(u)$:

$$f_{\widehat{u}}(u) := f(u) + \frac{\partial \widehat{u}}{\partial t} - \mathscr{D}\Delta\widehat{u} - f(\widehat{u}). \tag{50}$$

Then, $\widehat{u} = [\widehat{u}_1, \widehat{u}_2]$ in (49) would be the exact solution of $\partial u/\partial t - \mathscr{D}\Delta u - f(u) = f_{\widehat{u}}(u)$ with the initial condition $u^0 = \widehat{u}(x, y, 0)$.

Table 2 summarizes the $L^\infty$-error $E_\infty[T]$ with $T = 1.0$ and the elapsed time (CPU) for the implicit, CN, variable-$\theta$ methods for three different meshes refined by a factor of 2 in both spatial and temporal directions with $\Delta x = \Delta y$. Since the solution including the initial condition is smooth over the

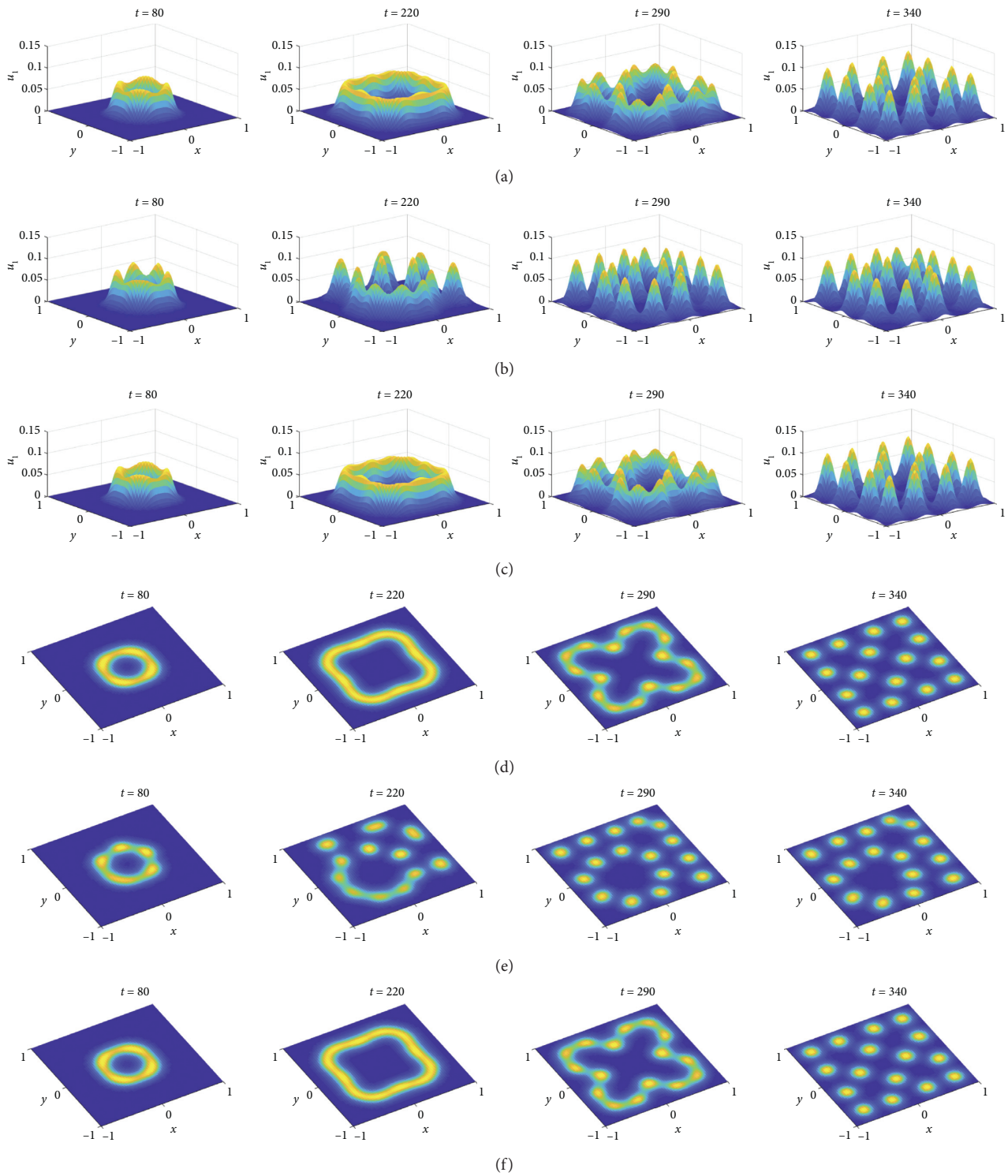
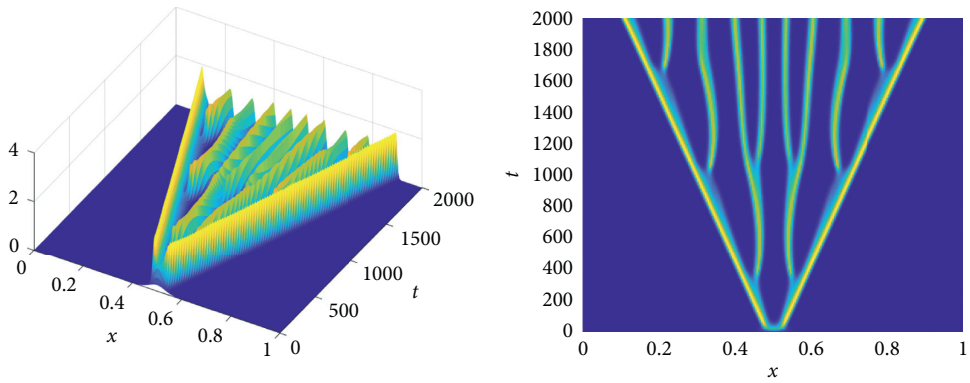



FIGURE 4: Numerical solutions for $u_1$ of the Gierer–Meinhardt model at the spatial resolution $\Delta x = \Delta y = 1/32$: (a) the variable-$\theta$ method with $\Delta t = 0.05$, (b) the CN method with $\Delta t = 0.05$, and (c) the CN method with $\Delta t = 0.005$; (d–f) are the aerial views of (a–c), respectively.

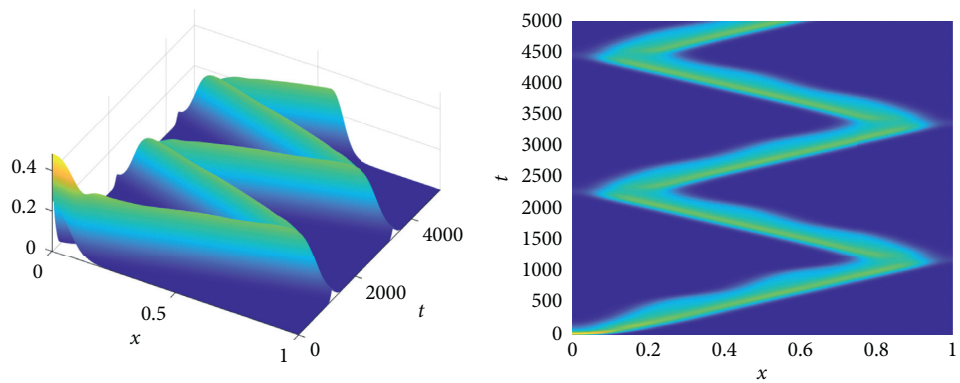
entire time interval $[0, T = 1]$, the CN method introduces no spurious oscillations into its numerical solution and proves a second-order accuracy for the two-component Gray–Scott model in the 2D space. One should notice that the variable-$\theta$ has also proved its second-order accuracy, the same as the CN method. On the contrary, the implicit method involves considerable errors due to its first-order convergence in temporal direction. Figure 8 shows the numerical solution by the variable-$\theta$ method, and its error at $T = 1.0$ for the Gray–Scott model, when $\Delta t = 0.025$ and $\Delta x = \Delta y = 0.05$.

For all the three methods, the algebraic system is solved by the SOR method with its optimal relaxation parameter




(a) (b)

Figure 5: Propagation of the numerical solution of $u_2$ for the 1D Gray–Scott model by the variable-$\theta$ method: (a) the wave-splitting (self-replication of the pulse) and (b) its aerial view over $0 \le t \le T = 2000$, when $\Delta t = 0.01$ and $\Delta x = 0.004$.




(a) (b)

Figure 6: Propagation of the numerical solution of $u_2$ for the 1D Gray–Scott model by the variable-$\theta$ method: (a) the pulse traveling and reflecting and (b) its aerial view over $0 \le t \le T = 5000$, when $\Delta t = 0.01$ and $\Delta x = 0.004$.

being calibrated from the lowest resolution, $(\Delta t, \Delta x = \Delta y) = (0.05, 0.1)$. That is, the constant $c_0$ in (41) is evaluated using the experimentally optimal $\widehat{\omega}_{\Delta t_0, h_0}$ with $(\Delta t_0, h_0) = (0.05, 0.1)$ and then (40) is utilized to estimate $\widehat{\omega}_{\Delta t, h}$ for other grid sizes $(\Delta t, h = \Delta x = \Delta y)$. With the *near-optimal* parameter and an effective initialization scheme in (30), for both the CN and variable-$\theta$ methods, the SOR method has converged in (59) iterations in average for solving the two algebraic systems (for $u_1$ and $u_2$) in a time level. SOR is comparable with ADI in efficiency when the parameter is set optimal and the initialization is carried out accurately; SOR has proven its efficiency for the numerical solution of elliptic obstacle problems [30]. For the Gray–Scott model in 2D, the variable-$\theta$ method becomes about a third most expensive computationally than the CN method, due to the wobble set processing.

*5.2.3. Gierer–Meinhardt Model.* The Gierer–Meinhardt model [2] is (1) defined in $\Omega = (-1, 1) \times (-1, 1) \subset \mathbb{R}^2$ with the following reaction kinetics and parameters:

$$\mathscr{D} = \left[\varepsilon^2, \frac{\kappa}{\mu}\right]^T,$$
$$f(u) = \left[\frac{u_1^2}{u_2} - u_1, \frac{1}{\mu}\left(\frac{u_1^2}{\varepsilon} - u_2\right)\right]^T, \tag{51}$$

for which various numerical methods have been developed [12, 31, 32]. We cast the experiment employing coefficients and the initial condition used in [32]:

The initial values are depicted in Figure 9. In this section, we restrict our attention to the dynamics of $u_1$ of the model.

$$\varepsilon = 0.04, \mu = 0.1, \kappa = 0.0128,$$
$$u_1(x, y, 0) = \frac{1}{2}\left[1 + 0.001 \sum_{k=1}^{20} \cos\left(\frac{k\pi y}{2}\right)\right] \operatorname{sech}^2\left(\frac{\sqrt{x^2 + y^2}}{2\varepsilon}\right),$$
$$u_2(x, y, 0) = \frac{\cosh\left(1 - \sqrt{x^2 + y^2}\right)}{3 \cos h(1)}. \tag{52}$$

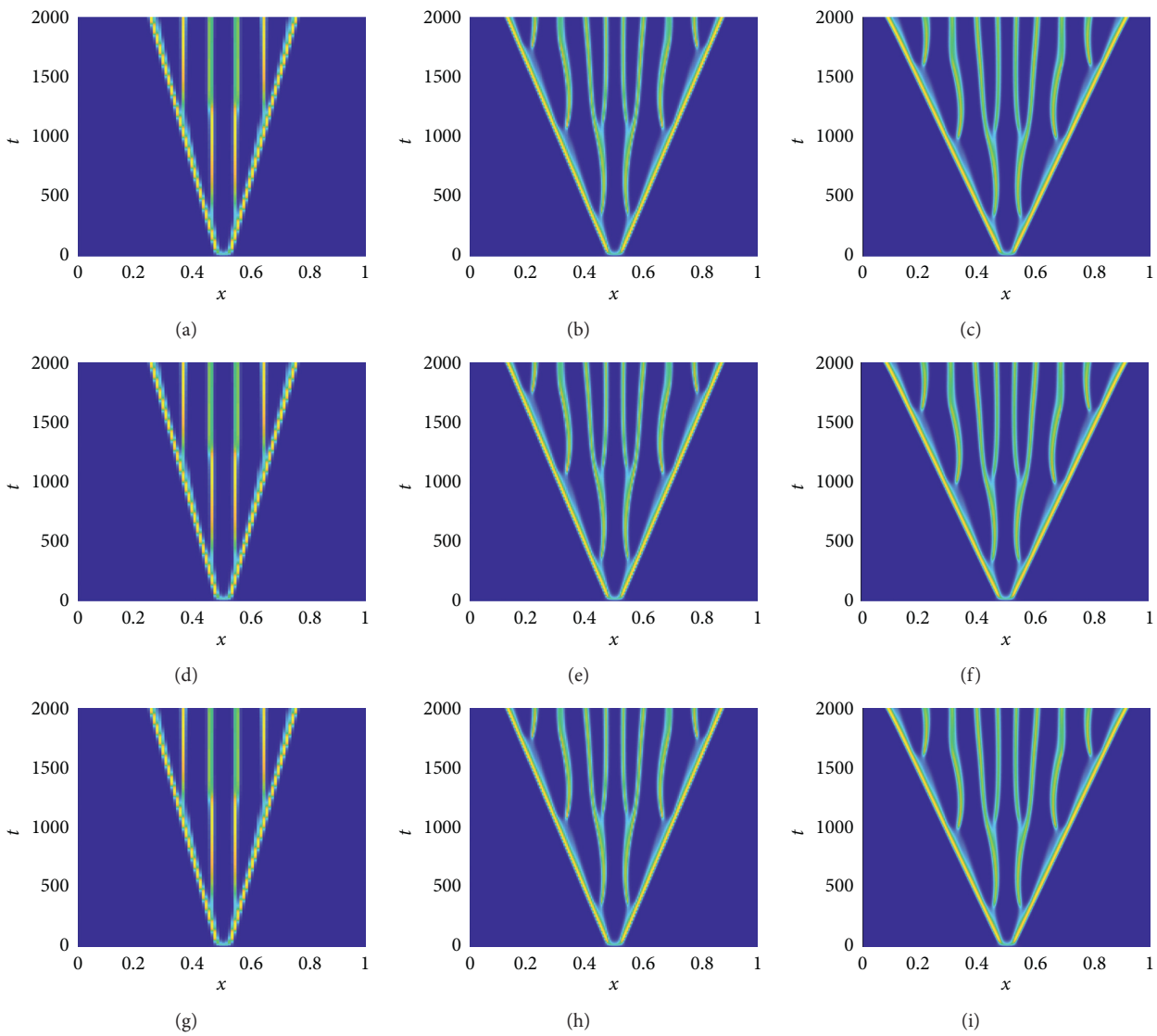



Figure 7: The wave-splitting (self-replication of the pulse) by the variable-$\theta$ method over $0 \le t \le 2000$ with various $(\Delta t, \Delta x)$. The image $I_{k\ell}$ represents the numerical solution obtained with the mesh resolution $(\Delta t, \Delta x) = (10^{-k}, 0.01/2^{\ell-1})$. (a) $(I_{11})$. (b) $(I_{12})$. (c) $(I_{13})$. (d) $(I_{21})$. (e) $(I_{22})$. (f) $(I_{23})$. (g) $(I_{31})$. (h) $(I_{32})$. (i) $(I_{33})$.

In order to investigate effectiveness of the variable-$\theta$ method and oscillatory behaviors of the CN method as well, we have carried out numerical experiments for the Gierer–Meinhardt model with a relatively low spatial resolution. Figure 4 presents numerical solutions at two different times for $u_1$ of the Gierer–Meinhardt model with the spatial resolution $\Delta x = \Delta y = 1/32$. When the time step size is set $\Delta t = 0.05$, the variable-$\theta$ method evolves the numerical solution as shown in Figures 4(a) and 4(d), for which the final steady-state pattern is the same as that in [32]. On the contrary, with the same $\Delta t = 0.05$, the CN method has produced a quite different pattern as in Figures 4(b) and 4(e), due to the nonsmooth initial values (Figure 9). However, the CN method can recover the correct steady-state pattern when it runs with $\Delta t = 0.005$, as depicted in Figures 4(c) and 4(f). For a similar accuracy, the variable-$\theta$ method (taking 170 s) is about 7 times more efficient than the CN method (taking 1242 s).

We summarize our experiments with the Gray–Scott and Gierer–Meinhardt models in 2D as follows:

(i) The variable-$\theta$ method shows the same accuracy as the CN method for problems of smooth data

(ii) For nonsmooth data, the variable-$\theta$ method evolves a smooth solution for all choices of $\Delta t$, while the CN method introduces spurious oscillations to alter the solution unless the time step size is sufficiently small

(iii) When a large time step size is desirable, the suggested method is a few times more efficient than the CN method for a similar accuracy

Table 2: $L^\infty$-error $E_\infty[T]$ and the elapsed time (CPU) for the numerical solution of the Gray–Scott model in the 2D space: $T = 1.0$.

| $\varepsilon = 10^{-6}$ | Implicit | | | CN | | | Variable-$\theta$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $(\Delta t, \Delta x = \Delta y)$ | $E_\infty[T]$ | | CPU | $E_\infty[T]$ | | CPU | $E_\infty[T]$ | | CPU |
| | $u_1$ | $u_2$ | | $u_1$ | $u_2$ | | $u_1$ | $u_2$ | |
| $(0.05, 0.1)$ | $4.3 \cdot 10^{-2}$ | $4.0 \cdot 10^{-2}$ | 0.036 s | $9.6 \cdot 10^{-4}$ | $9.4 \cdot 10^{-4}$ | 0.047 s | $9.6 \cdot 10^{-4}$ | $9.4 \cdot 10^{-4}$ | 0.062 s |
| $(0.025, 0.05)$ | $2.0 \cdot 10^{-2}$ | $1.7 \cdot 10^{-2}$ | 0.225 s | $2.5 \cdot 10^{-4}$ | $2.3 \cdot 10^{-4}$ | 0.280 s | $2.5 \cdot 10^{-4}$ | $2.3 \cdot 10^{-4}$ | 0.394 s |
| $(0.0125, 0.025)$ | $8.0 \cdot 10^{-3}$ | $8.0 \cdot 10^{-3}$ | 1.629 s | $6.4 \cdot 10^{-5}$ | $5.5 \cdot 10^{-5}$ | 2.102 s | $6.4 \cdot 10^{-5}$ | $5.5 \cdot 10^{-5}$ | 2.798 s |
| Conv. order | 1.2 | 1.2 | | 1.9 | 2.0 | | 1.9 | 2.0 | |

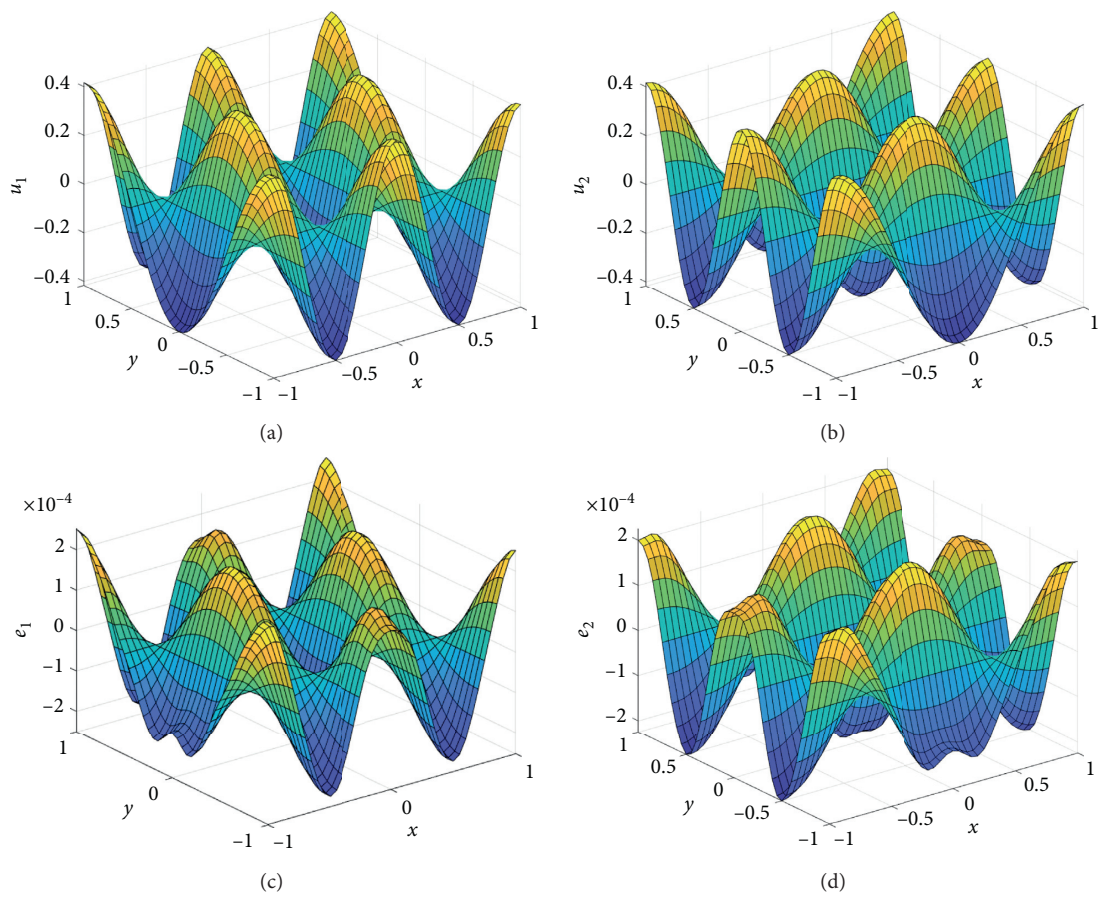



Figure 8: The numerical solution of the variable-$\theta$ method and its error at $T = 1.0$ for the Gray–Scott model, when $\Delta t = 0.025$ and $\Delta x = \Delta y = 0.05$: (a) $u_1$, (b) $u_2$, (c) $e_1 = u_1 - \widehat{u_1}$, and (d) $e_2 = u_2 - \widehat{u_2}$.

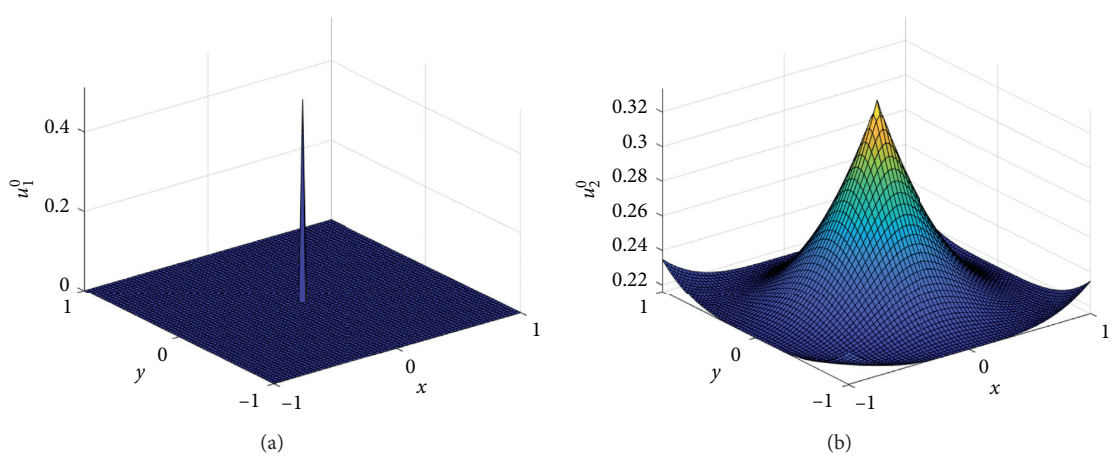



Figure 9: The initial values in (52) for the Gierer–Meinhardt model with $\epsilon = 0.04$ at the mesh resolution $\Delta x = \Delta y = 1/32$: (a) $u_1(x, y, 0)$ and (b) $u_2(x, y, 0)$.

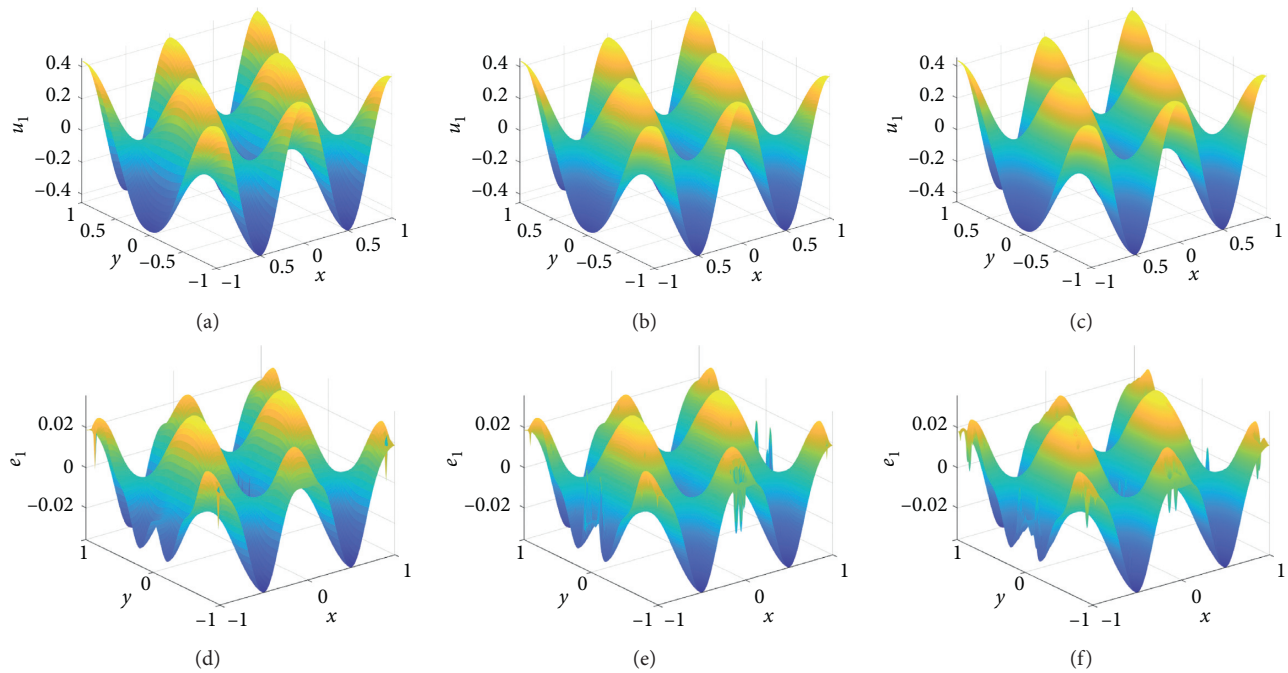



Figure 10: Numerical solutions for $u_1$ of the Gray–Scott model at $T = 1.0$ by the variable-$\theta$ method varying $\Delta x = \Delta y$ with fixed $\Delta t = 0.25$. The image $I_\ell$ represents the numerical solution obtained with the mesh resolution $(\Delta t, \Delta x = \Delta y) = (0.25, 0.02/2^{\ell-1})$ and $I'_\ell$ represents the error of $I_\ell$. (a) $(I_1)$. (b) $(I_2)$. (c) $(I_3)$. (d) $(I'_1)$. (e) $(I'_2)$. (f) $(I'_3)$.

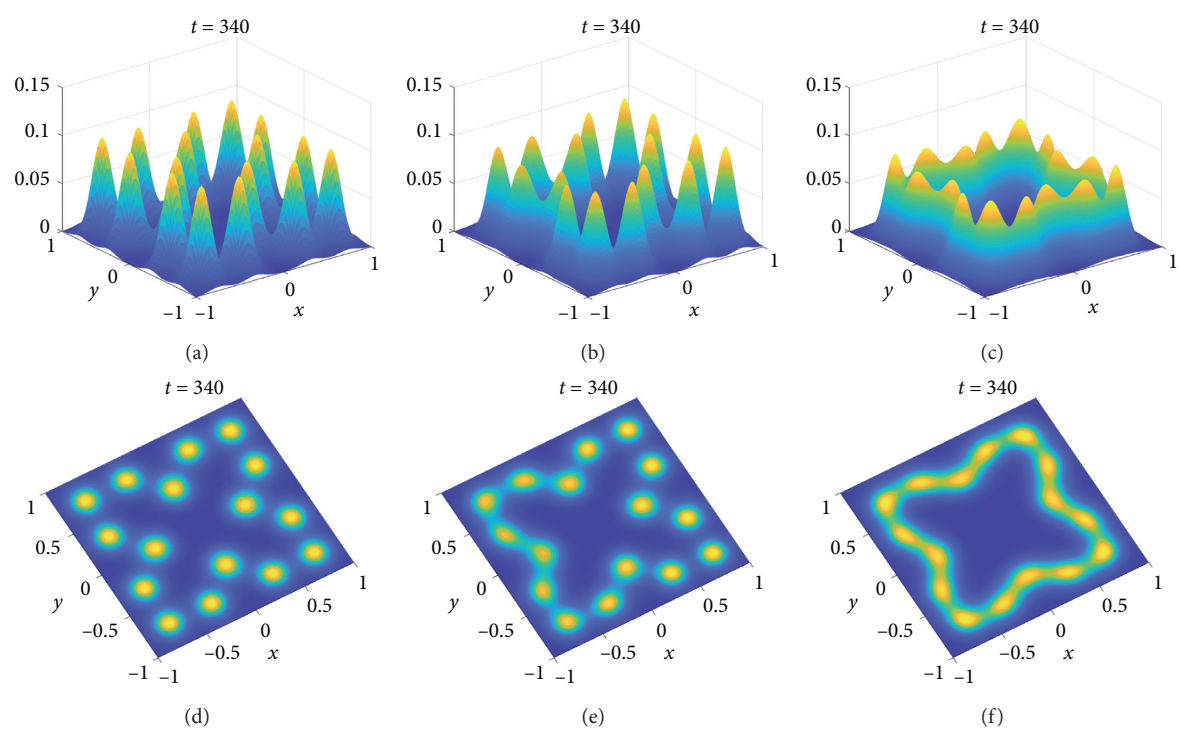



Figure 11: Numerical solutions for $u_1$ of the Gierer–Meinhardt model at $T = 340$ by the variable-$\theta$ method, varying $\Delta x = \Delta y$ with fixed $\Delta t = 0.1$. The image $I_\ell$ represents the numerical solution obtained with the mesh resolution $(\Delta t, \Delta x = \Delta y) = (0.1, 0.02/2^{\ell-1})$ and $I'_\ell$ represents the aerial view of $I_\ell$. (a) $(I_1)$. (b) $(I_2)$. (c) $(I_3)$. (d) $(I'_1)$. (e) $(I'_2)$. (f) $(I'_3)$.

*Remark 3.* Although the variable-$\theta$ method can employ larger time step sizes than the CN method to get stable numerical solutions for problems of nonsmooth data, one may not set the time step size too large, due to an accuracy issue rather than the stability issue. Furthermore, for nonlinear problems, the overall stability of the numerical algorithm can be determined by not

only grid sizes but also numerical schemes including methods of dealing with the nonlinear terms.

Figure 10 presents numerical solutions and their errors of $u_1$ for the Gray–Scott model (48)–(50) at $T = 1.0$ by the variable-$\theta$ method, varying $\Delta x = \Delta y$ with fixed $\Delta t = 0.25$. Compared with Figure 8, the solutions show stability and a good accuracy, although the time step size is as large as $\Delta t = 0.25$. As shown in the bottom line in Figure 10, all three spatially different cases show the same level of errors since the entire errors are dominated by temporal direction errors. We conclude from this example that grid sizes in both temporal and spatial directions would not significantly affect the stability of the proposed method when the initial condition is smooth and the nonlinearity is not severe.

As an example of nonsmooth data and severe nonlinearity, we select the Gierer–Meinhardt model (51) and (52) to simulate with large temporal step sizes. When $\Delta t \geq 0.2$, the proposed algorithm introduced a rapid decay of solution values independently of the spatial grid size, so that the pattern is not formed appropriately. We believe that it is due to the error incorporated with the reaction term (19) when $u^{n+1/2}$ is approximated by the extrapolation scheme (18). However, when $\Delta t \leq 0.1$, our method produces stable solutions for all choices of spatial grid sizes. Figure 11 presents numerical solutions of $u_1$ for the Gierer–Meinhardt model at $T = 340$ by the variable-$\theta$ method with fixed $\Delta t = 0.1$ and various $\Delta x = \Delta y$. Note that for Gierer–Meinhardt model, the pattern forming is slow down as the spatial grid size becomes smaller, as shown in $(I_2)$ and $(I_3)$ of Figure 11; this tendency has been observed for all other choices of $\Delta t \leq 0.1$. This is another example that accuracy of the numerical solution is much more sensitive to the spatial mesh resolution than the temporal one.

## 6. Conclusions

The *Crank–Nicolson* (CN) method has been a popular second-order time-stepping procedure for the numerical solution of systems of nonlinear RD equations. However, the CN method may introduce spurious oscillations for nonsmooth data unless the time step size is sufficiently small. We have studied a nonoscillatory time-stepping procedure for RD equations, called a *variable-θ method*, as a perturbation of the CN method. In each time level, the new method detects points of potential oscillations and resolves the solution applying the implicit method locally at those points. The proposed time-stepping procedure has proven nonoscillatory and having a second-order temporal accuracy, although the initial conditions are nonsmooth. Various examples have been considered to show effectiveness of the method. We also have performed a sensitivity analysis for the numerical solution of biological pattern forming models to conclude that the numerical solution is much more sensitive to the spatial mesh resolution than the temporal one.

## Data Availability

The experiment data and figures used to support the findings of this study are included within the article, and the MATLAB codes for the experiments are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

This research was supported by NSF-MCB (1714157) awarded to George V. Popescu.

## References

[1] A. M. Turing, "The chemical basis of morphogenesis," *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, vol. 237, no. 641, pp. 37–72, 1952.

[2] A. Gierer and H. Meinhardt, "A theory of biological pattern formation," *Kybernetik*, vol. 12, no. 1, pp. 30–39, 1972.

[3] J. Schnakenberg, "Simple chemical reaction systems with limit cycle behaviour," *Journal of Theoretical Biology*, vol. 81, no. 3, pp. 389–400, 1979.

[4] P. Gray and S. K. Scott, "Autocatalytic reactions in the isothermal, continuous stirred tank reactor: isolas and other forms of multistability," *Chemical Engineering Science*, vol. 38, no. 1, pp. 29–43, 1983.

[5] I. Prigogine and G. Nicolis, "Self-organisation in nonequilibrium systems: towards a dynamics of complexity," in *Bifurcation Analysis*, pp. 3–12, Springer, Berlin, Germany, 1985.

[6] I. Lengyel and I. R. Epstein, "Modeling of turing structures in the chlorite–iodide–malonic acid–starch reaction system," *Science*, vol. 251, no. 4994, pp. 650–652, 1991.

[7] S. Kondo and T. Miura, "Reaction-diffusion model as a framework for understanding biological pattern formation," *Science*, vol. 329, no. 5999, pp. 1616–1620, 2010.

[8] P. K. Maini and R. E. Baker, *Developmental Biology: Mathematical Modelling of Development*, Wiley, Hoboken, NJ, USA, 2001.

[9] K. U. Torii, "Two-dimensional spatial patterning in developmental systems," *Trends in Cell Biology*, vol. 22, no. 8, pp. 438–446, 2012.

[10] E. Crampin and P. Maini, "Reaction-diffusion models for biological pattern formation," *Methods and Applications of Analysis*, vol. 8, no. 3, pp. 415–428, 2001.

[11] C. Chiu and N. Walkington, "An ADI method for hysteretic reaction-diffusion systems," *SIAM Journal on Numerical Analysis*, vol. 34, no. 3, pp. 1185–1206, 1997.

[12] R. I. Fernandes, B. Bialecki, and G. Fairweather, "An ADI extrapolated Crank-Nicolson orthogonal spline collocation method for nonlinear reaction-diffusion systems on evolving domains," *Journal of Computational Physics*, vol. 299, pp. 561–580, 2015.

[13] I. Sgura, B. Bozzini, and D. Lacitignola, "Numerical approximation of Turing patterns in electrodeposition by ADI methods," *Journal of Computational and Applied Mathematics*, vol. 236, no. 16, pp. 4132–4147, 2012.

[14] F. Shakeri and M. Dehghan, "The finite volume spectral element method to solve Turing models in the biological pattern formation," *Computers & Mathematics with Applications*, vol. 62, no. 12, pp. 4322–4336, 2011.

[15] R. I. Fernandes and G. Fairweather, "An ADI extrapolated Crank-Nicolson orthogonal spline collocation method for nonlinear reaction-diffusion systems," *Journal of Computational Physics*, vol. 231, no. 19, pp. 6248–6267, 2012.

[16] J. Douglas Jr., "On the numerical integration of $^{\partial/2}u\partial x^2 + \partial^2 u/\partial y^2 = \partial u/\partial t$ by implicit methods," *Journal of the Society for Industrial and Applied Mathematics*, vol. 3, no. 1, pp. 42–65, 1955.

[17] J. Douglas Jr. and D. W. Peaceman, "Numerical solution of two-dimensional heat-flow problems," *AIChE Journal*, vol. 1, no. 4, pp. 505–512, 1955.

[18] D. W. Peaceman and H. H. Rachford Jr., "The numerical solution of parabolic and elliptic differential equations," *Journal of the Society for Industrial and Applied Mathematics*, vol. 3, no. 1, pp. 28–41, 1955.

[19] J. Douglas Jr., S. Kim, and H. Lim, "An improved alternating-direction method for a viscous wave equation," *Contemporary Mathematics*, vol. 329, pp. 99–104, 2003.

[20] H. Lim, S. Kim, and J. Douglas Jr., "Numerical methods for viscous and nonviscous wave equations," *Applied Numerical Mathematics*, vol. 57, no. 2, pp. 194–212, 2007.

[21] J. Douglas Jr. and S. Kim, "Improved accuracy for locally one-dimensional methods for parabolic equations," *Mathematical Models and Methods in Applied Sciences*, vol. 11, no. 9, pp. 1563–1579, 2001.

[22] J. Crank and P. Nicolson, "A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type," in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 43, pp. 50–67, Cambridge University Press, Cambridge, UK, 1947.

[23] J. D. Lawson and J. L. Morris, "The extrapolation of first order methods for parabolic partial differential equations. I," *SIAM Journal on Numerical Analysis*, vol. 15, no. 6, pp. 1212–1224, 1978.

[24] P. Lee and S. Kim, "A variable-$\theta$ method for parabolic problems of nonsmooth data," *Computers & Mathematics with Applications*, vol. 79, no. 4, pp. 962–981, 2020.

[25] R. Varga, *Matrix Iterative Analysis*, Springer, Berlin, Germany, 2nd edition, 2000.

[26] A. Q. M. Khaliq, T. A. Biala, S. S. Alzahrani, and K. M. Furati, "Linearly implicit predictor-corrector methods for space-fractional reaction-diffusion equations with non-smooth initial data," *Computers & Mathematics with Applications*, vol. 75, no. 8, pp. 2629–2657, 2018.

[27] P. Gray, "Autocatalytic reactions in the isothermal, continuous stirred tank reactor: oscillations and instabilities in the system $a + 2b \longrightarrow 3b; b \longrightarrow c$," *Chemical Engineering Science*, vol. 39, no. 6, pp. 1087–1097, 1984.

[28] P. A. Zegeling and H. Kok, "Adaptive moving mesh computations for reaction–diffusion systems," *Journal of Computational and Applied Mathematics*, vol. 168, no. 1-2, pp. 519–528, 2004.

[29] C. Gingras and P. G. Kry, "Procedural modelling with reaction diffusion and growth of thin shells," in *Proceedings of the 45th Graphics Interface Conference on Proceedings of Graphics Interface 2019*, Canadian Human-Computer Communications Society, New York, NY, USA, pp. 1–7, 2019.

[30] P. Lee, T. W. Kim, and S. Kim, "Accurate and efficient numerical solutions for elliptic obstacle problems," *Journal of Inequalities and Applications*, vol. 34, no. 7, pp. 1–25, 2017.

[31] M. McCourt, N. Dovidio, and M. Gilbert, "Spectral methods for resolving spike dynamics in the gierer-meinhardt model," *Communications in Computational Physics*, vol. 3, pp. 659–678, 2008.

[32] Z. Qiao, "Numerical investigations of the dynamical behaviors and instabilities for the gierer-meinhardt system," *Communications in Computational Physics*, vol. 3, no. 2, pp. 406–426, 2008.