MATHEMATICAL BIOSCIENCES
AND ENGINEERING
Volume 4, Number 2, April 2007

http://www.mbejournal.org/
pp. 187–203

# AN OPTIMAL ADAPTIVE TIME-STEPPING SCHEME FOR SOLVING REACTION-DIFFUSION-CHEMOTAXIS SYSTEMS

Chichia Chiu

Department of Mathematics, Michigan State University
East Lansing, MI 48864

Jui-Ling Yu

Department of Applied Mathematics, Providence University
Taichung, Taiwan

(Communicated by Qing Nie)

Abstract. Reaction-diffusion-chemotaxis systems have proven to be fairly accurate mathematical models for many pattern formation problems in chemistry and biology. These systems are important for computer simulations of patterns, parameter estimations as well as analysis of the biological systems. To solve reaction-diffusion-chemotaxis systems, efficient and reliable numerical algorithms are essential for pattern generations. In this paper, a general reaction-diffusion-chemotaxis system is considered for specific numerical issues of pattern simulations. We propose a fully explicit discretization combined with a variable optimal time step strategy for solving the reaction-diffusion-chemotaxis system. Theorems about stability and convergence of the algorithm are given to show that the algorithm is highly stable and efficient. Numerical experiment results on a model problem are given for comparison with other numerical methods. Simulations on two real biological experiments will also be shown.

1. **Introduction.** Reaction-diffusion-chemotaxis systems have been used successfully to model chemical and biological processes which involve pattern formations. Since virtually all models for spatial pattern generation are necessarily nonlinear, analytical solutions are generally not available. Numerical solutions of such equations are thus necessary for computer simulation of patterns and data analysis. A key question for such numerically generated patterns is how we know that the computer-generated patterns come from the original models (systems) and not from noise or error introduced by numerical approximation or computation. This is specially important for noise-sensitive patterns. To answer this question, we need to look at the characteristics of reaction-diffusion-chemotaxis systems, which separate them from other parabolic type of systems.

In 1952, A. M. Turing [26] first suggested that some patterns that occur in chemistry were resulted from interaction between chemical reaction and diffusion. Since then a substantial amount of research has been done on this subject. See

2000 *Mathematics Subject Classification.* 65N06, 65N12, 65N55, 92-08.

*Key words and phrases.* reaction, diffusion, chemotaxis, systems, equations, pattern formation problems, explicit methods.

[18] for a survey of related developments. The general form of 2-function reaction-diffusion systems is

$$\begin{cases} u_t &= D_u \Delta u + f(u,v) \\ v_t &= D_v \Delta v + g(u,v), \end{cases} \tag{1}$$

together with appropriate boundary and initial conditions. Here $u$ and $v$ are considered as two chemical concentrations; $f(u,v)$ and $g(u,v)$ are reaction kinetics which describe the interrelation between the chemicals. The diffusion terms, $D_u \Delta u$ and $D_v \Delta v$, reflect each particle's random movement with diffusivities $D_u > 0$ and $D_v > 0$. Under certain conditions, the reaction-diffusion system described above has a unique solution. The proof of the existence and uniqueness of the solution can be found in [6]. The system is useful for modeling patterns in chemistry and biology. Some examples can be found in [18].

The basic idea for showing patterns using (1) is to do a local linearization of $f$ and $g$ at an equilibrium point and demonstrate that eigenvalues responsible for growth (decay) can have positive (negative) real parts that can lead to growth at some places and decay at others. So, the whole picture is a spatially inhomogeneous pattern [12] [23].

A motile organism can move toward higher nutrient levels or better environment, for example, higher level of oxygen. It can also move away from a chemical. This type of motion is called *chemotaxis.* The chemotaxis phenomenon has been observed in many microorganisms, plants and animals. For bacteria, chemotaxis has been known since the end of of the nineteenth century [30]. Keller and Segel [13] were among the first to construct differential equation models for chemotaxis. Their model has been applied to many real cases. The Keller-Segel model extends the reaction-diffusion model with a convection term to simulate chemotaxis. For many motile organisms, the reaction-diffusion-chemotaxis systems are more realistic models than reaction-diffusion systems. Chemotaxis can play crucial roles in pattern formations [13] [15]. For different biological problems, the corresponding equations of reaction-diffusion-chemotaxis systems may be different depending on the species involved and their biological properties. In this paper, we will consider a general reaction-diffusion-chemotaxis system which will be used for our numerical experiments on generating patterns given in [2]. The system is given by

$$\begin{cases} u_t &= D_u \Delta u + \nabla \cdot (\chi_1(p) u \nabla p) + \nabla \cdot (\chi_2(q) u \nabla q) + f(p,u) \\ p_t &= D_p \Delta p + g(p,q,u), \\ q_t &= D_q \Delta q + h(p,q,u), \end{cases} \tag{2}$$

together with appropriate boundary and initial conditions. Here $u$ is considered as a microorganism (bacteria) density function, $p$ and $q$ are two chemoattractant concentration functions (one operates as a nutrient and the other does not), and $f$, $g$ and $h$ are reaction kinetics which describe the interrelation between the chemicals. The diffusion terms, $D_u \Delta u$, $D_p \Delta p$, $D_q \Delta q$, reflect each particle's random movement with diffusivities $D_u > 0$, $D_p >$ and $D_q > 0$. $\nabla p$ and $\nabla q$ are gradients of $p$ and $q$. $\chi = \chi_1$ or $\chi_2$ is defined by the Keller-Segal model

$$\chi(x) \;=\; \frac{C_0}{(C_1 + x)^p},$$

where $C_0 > 0$ and $C_1 > 0$ are constants and $p \geq 1$ is an integer power. Observe that $\lim_{x \to \infty} \chi(x) = 0$, which indicates that particles tend to stay where they are when the local concentration of chemoattractant is large.

The theoretical analysis of reaction-diffusion-chemotaxis systems for pattern generation is similar to the analysis of reaction-diffusion systems. That is, to show some eigenvalues of local linearization can have positive real parts.

Numerical methods that have been used to solve reaction-diffusion systems include finite element techniques, monotone iterative methods and explicit finite difference schemes [16] [17] [20] [28] [29]. In this paper, we consider a general approach to the problem. First discretize in space, then solve the resulting ordinary differential equation (ODE) system. This is called the method of lines, which is the first step for construction of many numerical methods. There are numerous ways to discretize in space. The most popular ones are finite difference, finite element and various kinds of spectral methods. For construction and demonstration of our numerical method, we use a straightforward finite difference discretization. Other discretization methods would give different forms of the ODE system, for which our numerical scheme, given in the sequel, can also be applied.

To solve the ODE system, the simplest and the most straightforward numerical scheme is the forward Euler scheme, which is fully explicit and easy to implement. However, the conditional stability forces a small step size in time for numerical computations. This is also true for other explicit methods. Consequently, the explicit schemes require long runtimes. This problem is specially serious for stiff systems, which are typical for systems with highly nonlinear reaction terms. The common remedies for this problem are to use implicit or semi-implicit schemes [8] [9] [11] [14]. Although such implicit methods are efficient for problems with negative real parts for eigenvalues of the discrezation matrices, these schemes create the same problem; namely, requiring small step sizes for very positive real parts of the eigenvalues (stiffness on the positive side of the number line). This is likely to happen for pattern formation problems for which the spatial discretization matrices have both positive and negative real parts for their eigenvalues. In addition, for pattern formation problems, the positive real parts of the eigenvalues are the unpredictable ones which evolve along the time. With these factors in mind, the explicit methods are actually better for the reaction-diffusion-chemotaxis systems.

In this paper, we will re-examine the explicit methods and construct a new adaptive time-stepping scheme which optimizes the time step sizes for explicit methods to increase their efficiency. This method is an adaptive strategy which only requires a simple estimation of one eigenvalue at each time step. The numerical experiments given in the following sections showed that our method is more efficient than other similar methods in terms of computer CPU time.

## 2. The method of lines and explicit finite difference discretization.

Most of the numerical methods for solving time-dependent problems or parabolic PDEs, specifically for our case, bears the idea of the method of lines; that is, discretizing in space to change the PDE to an ODE system, then solve the corresponding ODE system. In this paper, we use the standard finite differences of center difference for the Laplacian operator and upwind differences for the chemotaxis term. The reason for using upwind differences is the stability concern. When the chemotaxis dominants, the system behaves more like a hyperbolic system. Then the upwind differences are more stable methods [24]. It should be mentioned again that the following time-stepping scheme can also be applied to other discretization methods when the domain containing the eigenvalues is known or can be estimated.

Let us consider the two-dimensional reaction-diffusion-chemotaxis system (2) with zero flux boundary conditions. For simplicity, we only consider the first equation for the bacteria density, assuming that $p$ and $q$ are known. Consider a uniform rectangular mesh on a square domain $\Omega = [0, L] \times [0, L]$ with mesh size $h = L/N$. Let $(x_i, y_j) \in \Omega$ be a grid point. Then $x_{i+1} = x_i + h$ and $y_{j+1} = y_j + h$. We adopt the standard notation $u_{ij}(t) \approx u(x_i, y_j, t)$. The corresponding ODE system is given by

$$\begin{array}{rcl}
u_{ij}^{'}(t) & = & D_u(\frac{\Delta_{xh}u_{ij}(t)}{h^2} + \frac{\Delta_{yh}u_{ij}(t)}{h^2}) \\
& - & \frac{\chi_{1ij}p_{xij}(t)+|\chi_{1ij}p_{xij}(t)|}{2h}\nabla_{x-}u_{ij}(t) - \frac{\chi_{1ij}p_{xij}(t)-|\chi_{1ij}p_{xij}(t)|}{2h}\nabla_{x+}u_{ij}(t) \\
& & -u_{ij}(t)\nabla_{x+}\left(\chi_{1ij}p_{xij}(t)\right) \\
& - & \frac{\chi_{1ij}p_{yij}(t)+|\chi_{1ij}p_{yij}(t)|}{2h}\nabla_{y-}u_{ij}(t) - \frac{\chi_{1ij}p_{yij}(t)-|\chi_{1ij}p_{yij}(t)|}{2h}\nabla_{y+}u_{ij}(t) \\
& & -u_{ij}(t)\nabla_{y+}\left(\chi_{1ij}p_{yij}(t)\right) \\
& - & \frac{\chi_{2ij}q_{xij}(t)+|\chi_{2ij}q_{xij}(t)|}{2h}\nabla_{x-}u_{ij}(t) - \frac{\chi_{2ij}q_{xij}(t)-|\chi_{2ij}q_{xij}(t)|}{2h}\nabla_{x+}u_{ij}(t) \\
& & -u_{ij}(t)\nabla_{x+}\left(\chi_{2ij}q_{xij}(t)\right) \\
& - & \frac{\chi_{2ij}q_{yij}(t)+|\chi_{2ij}q_{yij}(t)|}{2h}\nabla_{y-}u_{ij}(t) - \frac{\chi_{2ij}q_{yij}(t)-|\chi_{2ij}q_{yij}(t)|}{2h}\nabla_{y+}u_{ij}(t) \\
& & -u_{ij}(t)\nabla_{y+}\left(\chi_{2ij}q_{yij}(t)\right) \\
& & +f(u_{ij}(t), p_{ij}(t))
\end{array} \tag{3}$$

where $\nabla p = (p_x, p_y)^T$ and $\nabla q = (q_x, q_y)^T$ are gradients of $p$ and $q$ while $\Delta_{xh}$ and $\Delta_{yh}$ are the centered second order difference operators such that

$$(\Delta_{xh} + \Delta_{yh})u_{ij}(t) = (u_{i+1j}(t) - 2u_{ij}(t) + u_{i-1j}(t)) + (u_{ij+1}(t) - 2u_{ij}(t) + u_{ij-1}(t)).$$

$\nabla_{x+}$ and $\nabla_{x-}$ are the standard forward and backward first order differences in $x$-direction. Similarly, $\nabla_{y+}$ and $\nabla_{y-}$ are the standard forward and backward first-order differences in $y$-direction. For example,

$$\nabla_{x+}u_{ij}(t) = u_{i+1j}(t) - u_{ij}(t) \text{ and } \nabla_{x-}u_{ij}(t) = u_{ij}(t) - u_{i-1j}(t).$$

The zero flux boundary condition is approximated by

$$\left\{\begin{array}{lllllll}
u_{iN_2+1}(t) & = & u_{iN_2-1}(t), & u_{i(-1)}(t) & = & u_{i1}(t), & \text{for all } i, \\
u_{N_1+1j}(t) & = & u_{N_1-1j}(t), & u_{(-1)j}(t) & = & u_{1j}(t), & \text{for all } j.
\end{array}\right.$$

Let $\mathbf{u}(t) = [u_{ij}(t)]$ and $\mathbf{f}(t, \mathbf{u}(t)) = [f(u_{ij}(t), p_{ij}(t))]$ be the vector obtained by the usual ordering. Then (3) can be written in the matrix form

$$\mathbf{u}^{'}(t) \quad = \quad A(t)\mathbf{u}(t) \ + \ \mathbf{f}(t, \mathbf{u}(t)), \tag{4}$$

where $A(t)$ is a square matrix depending on $t$.

To analyze system (4) and relate it to the pattern formation model (2), one can do the local linearization of the right-hand side of (4). If the real part of any of the eigenvalues of the linearization matrix is positive, that corresponds to a growth mode. If it is negative, then it corresponds to a decay mode. Because of this reason, we will look at the corresponding semilinear system case first, then generalize the idea to the nonlinear case.

## 3. **Construction of an explicit optimal method.**

We will introduce the idea by starting with the homogeneous constant coefficient system. Then we will generalize it to the general linear case and the nonlinear case.

3.1. **Homogeneous linear ODE system with constant coefficients.** Let us consider

$$\mathbf{u}^{'}(t) \quad = \quad M\mathbf{u}(t), \tag{5}$$

where $M$ is a constant square matrix.

The numerical methods for solving systems (4) and (5) can be classified as two groups: single-step methods and multistep methods. In this paper, we will only consider single-step methods, because they are easy to implement, especially for ODE systems coming from nonlinear PDE systems. Multistep methods would be a different approach. Among the single-step methods, there are two classes: explicit methods and implicit methods. Here we consider time discretization $t_0, t_1, t_2, \cdots$, where $t_0$ is the starting time. For simplicity, we first look at uniform time step size $\Delta t = t_n - t_{n-1}$ for all $n$. Then $\mathbf{u}^n \approx \mathbf{u}(t_n)$ is the $n^{th}$-step numerical solution. When a single-step method is applied to a scalar test problem, $u' = \lambda u$, where $\lambda$ is an eigenvalue of $M$, generally, it can be written as

$$u_{n+1} = p(\lambda)u_n,$$

where $p(\lambda)$ is a function of $\lambda$, which is called the amplification factor. Assume that $\lambda = a + bi$, where $a$ and $b$ are the real part and imaginary part of $\lambda$ respectively. For the pattern formation problems, we are interested in the long time behavior of the numerical solution. This leads to the long time stability definition as follows.

Definition 3.1. *A single-step method is said to be stable if* $|P(\lambda)| < 1$ *when* $a < 0$ *and* $|P(\lambda)| \leq 1 + a\Delta t + o(\Delta t)$ *when* $a \geq 0$.

Here $o(\Delta t)$ means higher order ($> 1$) of $\Delta t$. This definition is different from the usual A-stability, where only the case, $a < 0$, is considered. Our definition is designed to guarantee appropriate numerical solution for both decay and growth cases. With this definition, we get constraints on the step size, $\Delta t$. Typically, $\Delta t$ for explicit methods is limited when $a < 0$, and it is limited for implicit method when $a > 0$. The simple examples are Euler's method (explicit) and the backward Euler's method. For Euler's method, we need to have $|P(\lambda)| = |1 + \lambda\Delta t| < 1,\ a < 0$. This forces $\Delta t$ to be small when $a$ is negative and of large magnitude. But it does not put any condition on the case when $a \geq 0$. On the other hand, for the backward Euler's method for which $|P(\lambda)| = |\frac{1}{1-\lambda\Delta t}| > 1,\ a > 0$. This condition also forces $\Delta t$ to be small when $a$ is large and positive. We will show that for reaction-diffusion-chemotaxis systems, the eigenvalues with negative real parts are relatively easy to estimate while the eigenvalues with positive real part are difficult to predict. For this reason, we choose single-step explicit methods, which are basically Runge-Kutta methods. In this paper, we propose a new optimal two-stage Runge-Kutta method for solving (4) from the reaction-diffusion-chemotaxis system (2). To explain the idea, we will consider (5) first. Any two-stage Runge-Kutta method applied to (5) results in the formula

$$\mathbf{u}_{n+1} \ = \ \left(I + \Delta t M + \frac{1}{2}\Delta t^2 M^2\right)\mathbf{u}_n.$$

This leads to the amplification factor

$$p(\lambda) = 1 + \Delta t\lambda + \frac{1}{2}\Delta t^2\lambda^2.$$

It is a classical result that the truncation error of two-stage Runge-Kutta methods is $O(\Delta t^3)$. The following theorem shows that we can approximate the amplification factor by a simpler function which depends only on the real part of $\lambda$ with an error of $O(\Delta t^3)$. So we can work with this simpler function without losing the order of accuracy.

THEOREM 3.1.

$$|p(\lambda)|^2 \;=\; |p(a+ib)|^2 \;=\; (1+\Delta ta+\frac{1}{2}\Delta t^2a^2)^2+O(\Delta t^3).$$

*Proof:*

$$\begin{aligned}
|p(\lambda)|^2 &= (1+\Delta ta+\frac{\Delta t^2}{2}(a^2-b^2))^2+(\Delta tb+ab\Delta t^2)^2\\
&= (1+\Delta ta)^2+\Delta t^2(1+\Delta ta)(a^2-b^2)+O(\Delta t^4)+\Delta t^2b^2+O(\Delta t^3)\\
&= (1+\Delta ta)^2+\Delta t^2(1+\Delta ta)a^2+O(\Delta t^3)\\
&= (1+\Delta ta)^2+\Delta t^2(1+\Delta ta)a^2+\frac{1}{4}\Delta t^4a^4+O(\Delta t^3)\\
&= (1+\Delta ta+\frac{1}{2}\Delta t^2a^2)^2+O(\Delta t^3).
\end{aligned}$$

□

This theorem indicates that the two-stage Runge-Kutta methods satisfy the stability condition automatically for $a > 0$ but need constraints on $\Delta t$ when $a < 0$. So we want to find a step size that not only guarantees the stability but also damps the negative modes ($a < 0$) efficiently. In this way, we are sure that the negative modes ($a < 0$) will decay efficiently and only the positive modes ($a > 0$) will grow. In this sense, we can say that our numerically generated patterns truly follow the original models.

For a real matrix, $M$, it is easy and straightforward to estimate the real parts of the eigenvalues by the discs centered at the diagonal elements using the Gerschgorin Theorem [1]. So we can assume that we already know the range of the negative real parts of the eigenvalues of $M$. For explicit methods, the negative real parts of the eigenvalues ($a < 0$) are responsible for the stability and the restriction of the step size. The following theorem states how we can damp the negative modes efficiently in order to obtain optimal step sizes.

THEOREM 3.2. *Let $f(a,\Delta t) = 1+a\Delta t+\frac{1}{2}a^2\Delta t^2$. Assume that $-a_{max} \leq a \leq -a_{min}$, where $a_{max} > a_{min} > 0$ are two positive numbers. Then the solution of the following mini-max problem occurs at $\Delta t = \Delta t_{opt} = \frac{2}{a_{max}+a_{min}}$. The mini-max problem is stated as*

$$\min_{\Delta t>0}\;\max_{-a_{max}\leq a\leq -a_{min}} 1+a\Delta t+\frac{1}{2}a^2\Delta t^2.$$

**Proof:** For each fixed $a$, $f(a,\Delta t)$ is a quadratic polynomial in $\Delta t$. Then for any fixed $\Delta t$, the maximum of $f(a,\Delta t)$ is obtained at $f(-a_{min},\Delta t)$, $\Delta t < \Delta t_{opt}$, and at $f(a_{max},\Delta t)$, $\Delta t > \Delta t_{opt}$. So the mini-max occurs when $f(a_{max},\Delta t) = f(a_{min},\Delta t)$, *i.e,*

$$1-a_{max}\Delta t+\frac{1}{2}a_{max}^2\Delta t^2 \;=\; 1-a_{min}\Delta t+\frac{1}{2}a_{min}^2\Delta t^2.$$

This leads to the solution. □

3.2. **Linear case: time dependent system.** Now we want to generalize the method introduced in Section 3.1 for constant coefficient case to the time dependent coefficient case:

$$\mathbf{u}^{'}(t) \quad = \quad M(t)\mathbf{u}(t) \; + \; \mathbf{b}(t), \tag{6}$$

where $M(t)$ is a time-dependent matrix and $\mathbf{b}(t)$ is a time-dependent vector. Since this system originally comes from a nonlinear PDE system, it is difficult to deal with fractional step sizes which are usually required for high-stage Runge-Kutta methods. That leads us to consider two-stage Runge-Kutta methods only in this paper. The general two-stage Reunge-Kutta method for our problem is

$$\begin{aligned}
\mathbf{u}_{n+1} &= \mathbf{u}_n \; + \; \frac{\Delta t}{2}\left(M_n\mathbf{u}_n + \mathbf{b}_n\right) \; + \; \frac{\Delta t}{2}\left(M_{n+1}\mathbf{u}^{(1)} + \mathbf{b}_{n+1}\right), \\
\mathbf{u}^{(1)} &= \mathbf{u}_n \; + \; \Delta t(M_n u_n + \mathbf{b}_n),
\end{aligned}$$

where $M_n = M(t_n)$ and $\mathbf{b}_n = \mathbf{b}(t_n)$. Combining two equations, we get

$$\begin{aligned}
\mathbf{u}_{n+1} &= \mathbf{u}_n \; + \; \frac{\Delta t}{2}\left(M_n\mathbf{u}_n + \mathbf{b}_n\right) \\
&\quad + \frac{\Delta t}{2}\left(M_{n+1}(\mathbf{u}_n \; + \; \Delta t(M_n u_n + \mathbf{b}_n)) + \mathbf{b}_{n+1}\right), \quad \text{where} \\
&= \mathbf{u}_n \; + \; \Delta t\frac{M_n + M_{n+1}}{2}\mathbf{u}_n \; + \; \frac{\Delta t^2}{2}M_nM_{n+1}\mathbf{u}_n \\
&\quad + \Delta t\frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2} \; + \; \frac{\Delta t^2}{2}M_{n+1}\mathbf{b}_n.
\end{aligned}$$

If we assume that all related functions are smooth, we know that $M_{n+1} = M_n + O(\Delta t)$. Let $J_n = \frac{M_n + M_{n+1}}{2}$. The above equation implies that

$$\begin{aligned}
\mathbf{u}_{n+1} &= \mathbf{u}_n \; + \; \Delta t J_n\mathbf{u}_n \; + \; \frac{\Delta t^2}{2}J_n^2\mathbf{u}_n \\
&\quad + \Delta t\frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2} \; + \; \frac{\Delta t^2}{2}J_n\mathbf{b}_n \; + \; O(\Delta t^3).
\end{aligned}$$

Based on the same argument, that the truncation error of the two-stage Runge-Kutta methods is $O(\Delta t^3)$, if we drop the term $O(\Delta t^3)$ in the above formula, we would not change the order of the truncation error. So we have,

$$\begin{aligned}
\mathbf{u}_{n+1} &= \mathbf{u}_n \; + \; \Delta t J_n\mathbf{u}_n \; + \; \frac{\Delta t^2}{2}J_n^2\mathbf{u}_n \\
&\quad + \Delta t\frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2} \; + \; \frac{\Delta t^2}{2}J_n\mathbf{b}_n.
\end{aligned}$$

This formula is equivalent to the following two-stage formula:

$$\begin{aligned}
\mathbf{u}^{(1)} &= \mathbf{u}_n \; + \; \frac{\Delta t}{2}(J_n\mathbf{u}_n \; + \; \mathbf{b}_n), \\
\mathbf{u}_{n+1} &= \mathbf{u}_n \; + \; \Delta t(J_n\mathbf{u}^{(1)} \; + \; \frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2}).
\end{aligned}$$

The obvious reason for doing this is to note that at $n^{th}$ local step, the position of $J_n$ is the same as that of $M$ for the linear constant coefficient case when $\mathbf{b}(t) = 0$ (homogeneous case). Therefore, we can turn our strategy in Section 3.1 into a variable time step scheme for the linear case.

**An explicit optimal time step scheme for solving system (6)**

- Set up initial condition $\mathbf{u}_0$ and initial step size $\Delta t_0$
- $$\begin{aligned} J_n &= \frac{M(t_n) + M(t_n + \Delta t_n)}{2} \\ \mathbf{u}^{(1)} &= \mathbf{u}_n \;+\; \frac{\Delta t_n}{2}(J_n \mathbf{u}_n \;+\; \mathbf{b}_n), \\ \mathbf{u}_{n+1} &= \mathbf{u}_n \;+\; \Delta t_n (J_n \mathbf{u}^{(1)} \;+\; \frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2}). \end{aligned}$$
- Use the Gerschgorin theorem to estimate the interval, $[-a^n_{max}, -a^n_{min}]$ for negative real parts of the eigenvalues of $J_n$. Let $\Delta t_{n+1} = 2/(a^n_{max} + a^n_{min})$ and $t_{n+1} = t_n + \Delta t_{n+1}$. When $a^n_{max}$ is equal to or close to zero, simply choose a reasonable step size for $\Delta t_{n+1}$.

3.3. **An explicit optimal time step scheme for the nonlinear system.** If the vector function, $\mathbf{f}(t, \mathbf{u}(t))$, in (4) is already of semilinear form; that is,

$$\mathbf{f}(t, \mathbf{u}(t)) \;=\; B(t)\mathbf{u}(t) \;+\; \mathbf{b}(t),$$

where $B(t)$ is a time-dependent matrix and $\mathbf{b}(t)$ is a time-dependent vector, then we simply let $M(t)$ in the above formula be $M(t) = A(t) + B(t)$. When $\mathbf{f}(t, \mathbf{u}(t))$ is a general nonlinear function, then under certain smooth conditions, there exists a linear approximation. Assume that $\mathbf{u}_n$ and $\Delta t_n$ are given, when $t \approx t_n$,

$$\mathbf{f}(t, \mathbf{u}(t)) \;\approx\; B_n(t)\mathbf{u}(t) \;+\; \mathbf{b}_n(t),$$

where $B_n(t)$ is a time-dependent matrix and $\mathbf{b}_n(t)$ is a time-dependent vector when $t \approx t_n$. The most straightforward approximation like this would be the Taylor expansion. Under the continuity condition, one can have the error for this approximation being $O(\Delta t_n^2)$. Then the approximation

$$\Delta t_n \mathbf{f}(t, \mathbf{u}(t)) \;\approx\; \Delta t\,(B_n(t)\mathbf{u}(t) \;+\; \mathbf{b}_n(t))$$

has an error of $O(\Delta t_n^3)$. Again based on the truncation error of the two-stage Runge-Kutta method, this error can be treated as within the tolerance. And we can use the local linearization to approximate the nonlinear function at every step. These give the method for the non-linear system (4).

**An explicit optimal time step scheme for solving system (4):**

- Set up initial condition $\mathbf{u}_0$ and initial step size $\Delta t_0$
- Find the local linearization: $\mathbf{f}(t, \mathbf{u}(t)) \;\approx\; B_n(t)\mathbf{u}(t) \;+\; \mathbf{b}_n(t)$.
- Let $M(t) = A(t) + B_n(t)$.
- $$\begin{aligned} \mathbf{b}_n &= \mathbf{b}_n(t_n), \\ \mathbf{b}_{n+1} &= \mathbf{b}_n(t_n + \Delta t_n), \\ J_n &= \frac{M(t_n) + M(t_n + \Delta t_n)}{2}, \\ \mathbf{u}^{(1)} &= \mathbf{u}_n \;+\; \frac{\Delta t_n}{2}(J_n \mathbf{u}_n \;+\; \mathbf{b}_n), \\ \mathbf{u}_{n+1} &= \mathbf{u}_n \;+\; \Delta t_n (J_n \mathbf{u}^{(1)} \;+\; \frac{\mathbf{b}_n + \mathbf{b}_{n+1}}{2}). \end{aligned}$$

- Use the Gerschgorin theorem to estimate the interval, $[-a^n_{max}, -a^n_{min}]$ for negative real parts of the eigenvalues of $J_n$. Let $\Delta t_{n+1} = 2/(a^n_{max} + a^n_{min})$ and $t_{n+1} = t_n + \Delta t_{n+1}$. When $a^n_{max}$ is equal to or close to zero, simply choose a reasonable step size for $\Delta t_{n+1}$.

## 4. **Numerical experiments.**

To demonstrate the implementation and the efficiency of our two-stage adaptive time-stepping method given in the previous sections, we constructed a one-dimensional model equation which has an analytic solution. Using this model equation, we can show more details about how to implement our algorithm and compare the numerical solutions with the analytic (true) solution. For showing the efficiency, we will compare the numerical results and computational cost of our scheme with that of three other methods: a standard two-stage Runge-Kutta method with fixed step size (Standard RK), an adaptive two-stage Runge-Kutta method (adaptive RK) and Runge-Kutta-Fehlberg method of order 2 and 3 (RKF 23). For adaptive RK, we solve the problem twice using step sizes $h$ and $h/2$, then compare answers at the mesh points corresponding to the larger step size. If the two answers agree closely (pre-assigned tolerance), the approximation is accepted and the program keeps the current step size $h$. If the two answers do not agree to a specified accuracy, the step size is reduced to $h/2$. If the answers agree to more significant digits than required, the step size is increased.

To apply our method to a real biological experiment, we also developed a two-dimensional reaction-diffusion-chemotaxis model for a sequence of biological experiments on bacterial chemotaxis reported in [2]. We used our optimal time-stepping scheme for the computer simulations of these biological experiments. The method was also applied to an existing model proposed in [28]. Some of these results will be shown here.

### 4.1. **A one-dimensional model equation.**

Our one-dimensional model equation is given by

$$u_t = u_{xx} - (txu)_x + f, \quad x \in (0, 2\pi),\ t > 0, \tag{7}$$

where $u = u(x,t)$ and $f = f(x,t) = (tx-1)e^{-t}\cos(x-t) + te^{-t}\sin(x-t)$. Here we can view the three terms on the right-hand side as simulations of diffusion, convection (chemotaxis) and reaction. Assume the periodic boundary condition: $u(0,t) = u(2\pi,t),\ t \geq 0$ and the initial condition: $u(x,0) = \sin x$, the analytic solution of (7) is given by

$$u(x,t) = e^{-t}\sin(x-t).$$

We now discretize (7) in space by the center difference for the diffusion term and upwind difference for the convection term. Since $tx$ is always positive for this model, the backward difference is used for the convection term. This leads to the following formula. Let $h = 2\pi/N,\ x_i = ih,\ i = 0,1,\ldots,N,\ u_i(t) \approx u(x_i,t), i = 0,1,2,\ldots,N,$ be the numerical solution. Then

$$u_i' = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} - tx_i\frac{u_i - u_{i-1}}{h} - tu_i + f(x_i,t).$$

Rearrange terms on the right-hand side, we have,

$$u_i' = \frac{1}{h^2}u_{i+1} + (\frac{-2}{h^2} - \frac{tx_i}{h} - t)u_i + (\frac{1}{h^2} + \frac{tx_i}{h})u_{i-1} + f(x_i,t).$$

Using the periodic boundary condition, $u_{-1} = u_{N-1}$ and $u_{N+1} = u_1$, this is equivalent to a system of ordinary differential equations

$$\mathbf{u}'(t) = M(t)\mathbf{u}(t) + \mathbf{f}(t),$$

where

$$\mathbf{u}(t) = (u_0(t)\, u_1(t) \ldots u_N(t))^T,$$
$$\mathbf{f}(t) = (f(x_0,t)\, f(x_1,t) \ldots f(x_N,t))^T,$$

and $M(t)$ is

$$\begin{pmatrix}
\frac{-2}{h^2} - \frac{x_0 t}{h} - t & \frac{1}{h^2} & & & & \frac{1}{h^2} + \frac{x_0 t}{h} & 0 \\
\frac{1}{h^2} + \frac{x_1 t}{h} & \frac{-2}{h^2} - \frac{x_1 t}{h} - t & \frac{1}{h^2} & & & 0 & 0 \\
0 & \frac{1}{h^2} + \frac{x_2 t}{h} & \frac{-2}{h^2} - \frac{x_2 t}{h} - t & \frac{1}{h^2} & & 0 & 0 \\
 & \ddots & \ddots & \ddots & & & \\
 & & \ddots & \ddots & \ddots & & \\
 & & & \ddots & \ddots & \ddots & \\
0 & \frac{1}{h^2} & & & & \frac{1}{h^2} + \frac{x_N t}{h} & \frac{-2}{h^2} - \frac{x_N t}{h} - t
\end{pmatrix}$$

Although this matrix is relatively simple, we should keep in mind that $M(t)$ generally is very complicated for real application problems. Therefore, we assume that eigenvalues of $M(t)$ are generally not available. However, it is easy and inexpensive to find a range for the real parts of the eigenvalues using the Gerschgorin theorem. And this is sufficient for implementing our algorithm. For this purpose, we quote a simple version of the Gerschgorin theorem as follows [1]:

THEOREM 4.1. *(Gerschgorin) For an $n \times n$ matrix $A = (a_{ij})$, the union of all discs:*

$$K_i \;=\; \{\mu \mid |\mu - a_{ii}| \leq \sum_{j \neq i} |a_{ij}|, \quad i = 1, 2, ..., n\},$$

*contains all the eigenvalues of $A$.*

If $A$ is a real matrix, which is true for our cases, then we can find a range for the real parts of the eigenvalues by simply looking at intersection intervals of these discs with the real axis. This is a simple procedure specially for sparse matrices. For many problems, a formula can be found before-hand so that we do not need to find the range at every time step.

Let $A = \frac{M(t_n)+M(t_{n+1})}{2}$, where $t_n$ refers to the same notation as given in the previous sections. For mesh point, $i = 0, 1, 2, \ldots n$, the diagonal elements of $A$ are

$$a_{ii} = \frac{-2}{h^2} - \frac{x_i(t_n + t_{n+1})}{2h} - \frac{t_n + t_{n+1}}{2}.$$

Then,

$$\sum_{j \neq i} |a_{ij}| = \frac{2}{h^2} + \frac{x_i(t_n + t_{n+1})}{2h}.$$

Let $\lambda$ be an eigenvalue of $A$. Then the Gerschgorin theorem implies

$$\left|\lambda - (\frac{-2}{h^2} - \frac{x_i(t_n + t_{n+1})}{2h} - \frac{t_n + t_{n+1}}{2})\right| \leq \frac{2}{h^2} + \frac{x_i(t_n + t_{n+1})}{2h}, \; i = 0, 1, 2, \ldots, N.$$

This implies

$$\frac{-4}{h^2} - \frac{x_i(t_n + t_{n+1})}{h} - \frac{t_n + t_{n+1}}{2} \leq Re(\lambda) \leq -\frac{t_n + t_{n+1}}{2},\ i = 0, 1, 2, \ldots, N.$$

Since

$$\begin{aligned} \frac{-4}{h^2} - \frac{x_i(t_n + t_{n+1})}{h} - \frac{t_n + t_{n+1}}{2} &\geq \frac{-4}{h^2} - \frac{x_N(t_n + t_{n+1})}{h} - \frac{t_n + t_{n+1}}{2} \\ &= \frac{-4}{h^2} - \frac{2\pi(t_n + t_{n+1})}{h} - \frac{t_n + t_{n+1}}{2}, \end{aligned}$$

$$-a_{max} \leq Re(\lambda) \leq -a_{min},$$

where

$$a_{min} = \frac{t_n + t_{n+1}}{2},\ a_{max} = \frac{4}{h^2} + \frac{2\pi(t_n + t_{n+1})}{h} + \frac{t_n + t_{n+1}}{2}.$$

Applying the algorithm given in the previous section, the optimal step size at current step is given by,

$$\Delta t_n = \frac{2}{a_{min} + a_{max}} = \frac{1}{\frac{2}{h^2} + \frac{t_n + t_{n+1}}{2} + \frac{\pi(t_n + t_{n+1})}{h}}.$$

Table 1 shows the number of iterations with different methods at different time. For calculating the number of iterations, every execution of one stage of Runge-Kutta method is counted as one iteration. So one time step of a two-stage Runge-Kutta method is equivalent to two iterations. This table indicates that the optimal time-stepping scheme converges while the standard two-stage Runge-Kutta methods converges slower (larger number of iterations) or diverges. It also shows that the other two adaptive methods require larger number of iterations. Table 2 gives the comparison of the CPU time for the four methods to demonstrate the efficiency of our method.

Table 1. Number of iterations for four methods. Here “Di” stands for “Divergent”.

| Method / Time | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Optimal time stepping | 284 | 639 | 1064 | 1561 | 2129 |
| Adaptive RK | 2994 | 6000 | 8964 | 11178 | 13416 |
| RKF23 | 1175 | 2765 | 4310 | 6075 | 8250 |
| Standard RK $\Delta t = 0.004$ | 250 | Div | Div | Div | Div |

Figure 1 shows the relative error between numerical solutions and the analytic solution. While the standard two-stage Runge-Kutta method blows up in a short time, the optimal time-stepping scheme and the other two adaptive methods converge with almost the same error rate.

4.2. **Computer simulations of chemotaxis in bacteria.** Chemotaxis is not new to biologists. Early experiments for showing bacterial chemotaxis can be traced back to 1800s [3]. This subject has been studied by scientists for various purposes. Our focus here is to demonstrate the efficiency and accuracy of our numerical method. For this purpose, we will show two numerical results. In the first result,

Table 2. Comparison of CPU time between the optimal time-stepping scheme, Adaptive RK, RKF 23 and Standard RK. The computation was done on Dell Latitude D620, 2600.

| Time / CPU (Secs) | Our scheme | Adaptive RK | RKF 23 | Standard RK $\Delta t = 0.004$ |
|---|---|---|---|---|
| 10 | 0.6406 | 0.9688 | 0.9688 | Div |
| 20 | 2.0156 | 2.7013 | 2.7188 | Div |
| 30 | 4.1563 | 4.9375 | 5.2656 | Div |
| 40 | 7.0313 | 8.1406 | 8.5469 | Div |
| 50 | 10.6719 | 12.0625 | 12.6094 | Div |

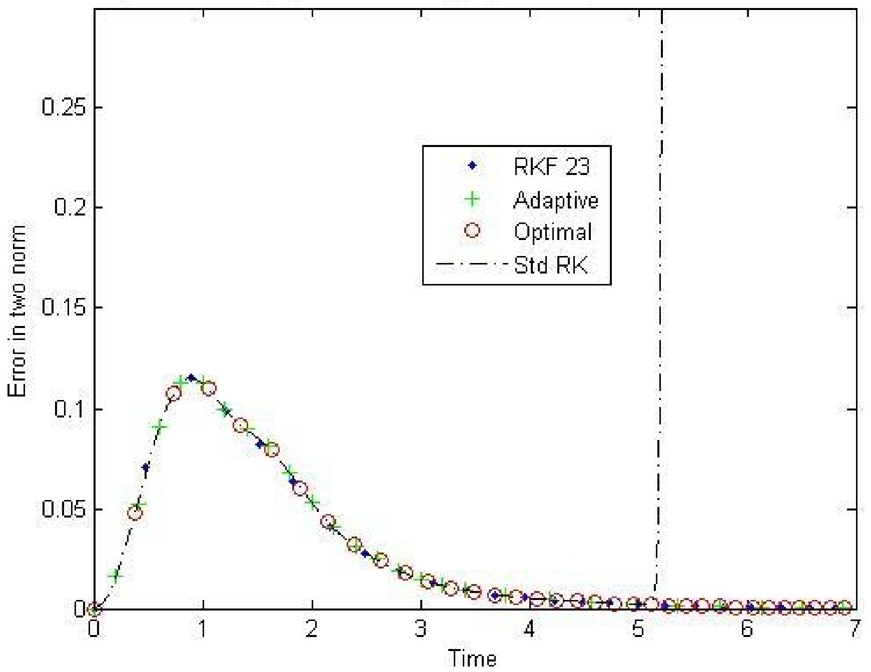

Figure 1. Error between numerical simulation and analytic solution.

we use the model proposed in [10] to the experimental results on bacterial chemotaxis done by J. Adler in 1966 [2]. To test our numerical scheme, we provide the second numerical experiment on the reaction-diffusion-chemotaxis model proposed by Tyson, Lubkin and Murry [27] with the biological experiments done by Budrene and Berg [7]. In both cases, we generated the patterns which simulated the biological experiments.

**Numerical Experiment 1:** The experimental results on bacterial chemotaxis done by J. Adler in 1966 [2] are relatively old results in this field. We chose Adler's experiments mainly for two reasons: (1) The results are generally well known and well accepted. Scientists still refer to these results for their own research works. (2) Compared with other reported results on chemotaxis, Adler's paper was well written with many details which are important for mathematical modeling.

One of the main results given in [2] is the two-band pattern formed by bacteria when two chemoattractants are in presence and the concentration of one attractant is relatively higher than the concentration of the other chemoattractant. These two chemoattractants were identified as galactose and oxygen. The experiments can be done with a capillary tube or a closed petri dish. We simulated the petri dish case to demonstrate the two-dimensional computation of our algorithm. Here, a particular form of the system (2) is used The bacteria equation $u$ is

$$\begin{aligned} u_t &= D_u\Delta u - \nabla(u\chi_1(p)\nabla p) - \nabla(u\chi_2(q)\nabla q) + f_1(p)u, && (8)\\ p_t &= D_p\Delta p - f_1(p)u/Y_p, && (9)\\ q_t &= D_q\Delta q - f_2(q)u/Y_q, && (10) \end{aligned}$$

where $u$ is the bacteria density, $p$ is the galactose (nutrient) density and $q$ is the limited oxygen density. Here $f_1$ and $f_2$ are constructed by Michaelis-Menten kinetics

$$f_i(x) = \frac{V_i x}{K_i + x},\ i = 1, 2,$$

and the Keller-Segel model is used for constructing $\chi_1$ and $\chi_2$.

$$\chi_i(x) = \frac{CK_i}{(DK_i + x)^2},\ i = 1, 2.$$

The boundary conditions and the initial conditions are given as

$$\frac{\partial u}{\partial n}|_{\partial\Omega} = \frac{\partial p}{\partial n}|_{\partial\Omega} = \frac{\partial q}{\partial n}|_{\partial\Omega} = 0.$$

$$u(x,0) = \begin{cases} u_0(x) & \text{if } \|x\| \le \ell \\ 0 & \text{Otherwise.} \end{cases}$$

$$p(x,0) = p_0(x), \quad q(x,0) = q_0(x).$$

Here the zero flux boundary conditions are imposed for obvious reasons. The initial condition for $u$ corresponds to the case that the bacteria was put in at the center and the initial functions for $g$ and $s$ are assumed to be uniform constant distributions. Since we do not have enough data from [2], we borrowed the parameters from [31]. The following is the parameter list.

- $u$ : bacteria density : [u]=0.001$cells/cm^3$.
- $g$ : concentration of galactose : [g]=0.005$mmole/cm^3$.
- $s$ : concentration of oxygen : [s]=0.033$mmole/cm^3$.
- $u_0(x) = 5 \times 10^{-6}$ if $\|x\| \le \ell$, where $\ell = 7.5 \times 10^{-2}$
- $s_0(x) = 4 \times 10^{-5}$, $g_0(x) = 2.64 \times 10^{-5}$
- $t$ : time : [t]=hours.
- $x$ : distance along the tube (peri dish) : [x]=cm.
- $\chi_1(g)$ = chemotactic sensitivity for galactose : $[\chi_1(s)]$=$cm^2/hour$.

- $\chi_2(s)$ = chemotactic sensitivity for oxygen : $[\chi_2(g)]$=$cm^2/hour$.
- $CK_1 = 0.02$, $DK_1 = 2 \times 10^{-6}$
- $CK_2 = 0.04$, $DK_2 = 3.3 \times 10^{-5}$
- $V_1 = 0.35$, $K_1 = 4 \times 10^{-6}$
- $V_2 = 0.6$, $K_2 = 5.5 \times 10^{-5}$
- $D_u = 0.001$ : bacteria motility : $[D_u]$=$cm^2/hr$
- $D_g = 0.005$, and $D_s = 0.033$ : constant diffusivities : $[D_.]$= $cm^2/hr$.
- $Y_g = 0.001$ : bacteria yield constant, which gives the yield of bacteria per galactose taken up : $[Y_g]$= mass u/mass g.
- $Y_s = 0.0001$ : bacteria yield constant, which gives the yield of bacteria per oxygen taken up : $[Y_s]$= mass u/mass s.
- $\Omega = [-R, R] \times [-R, R]$, where $R = 1$

Figure 2 shows the simulation results for the case that the concentration of oxygen is kept as constant and is the lower concentration one among the two chemoattractants. When the concentration of galactose gets higher, the appearance of the second band (inner band) is delayed and the two bands are farther apart from each other. This is the case shown by one of Adler's experiments. We have computer simulations of other Alder's experiments. The numerical results confirm that our model and our numerical method are valid and accurate [32].

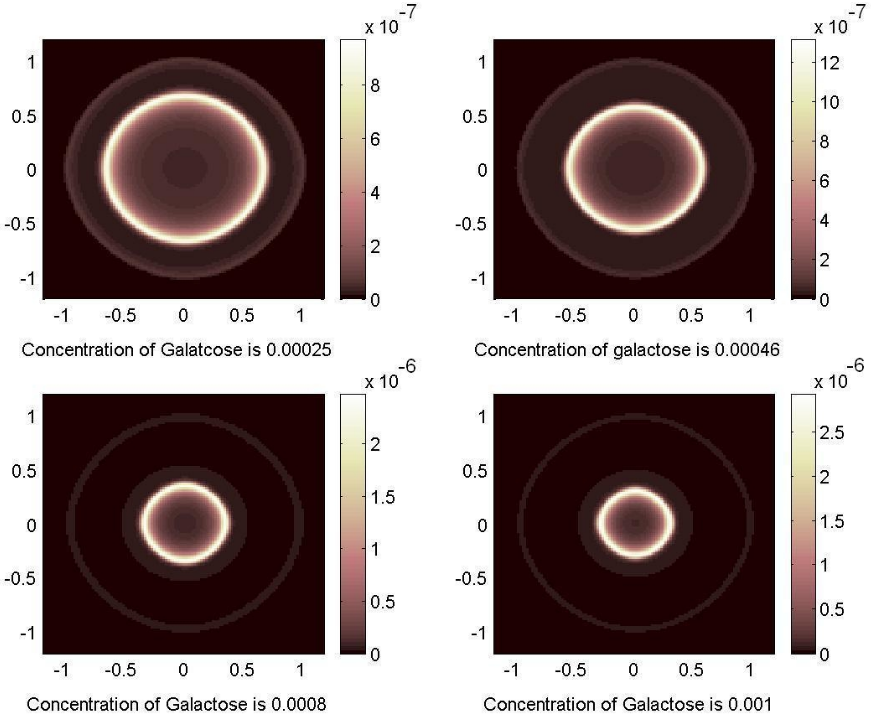

FIGURE 2. Simulations of bacterial bands when the concentration of galactose is higher than that of oxygen and the concentration of oxygen is kept as constant.

**Numerical Experiment 2:** Our second numerical experiment is based on the biological experiments reported in [7] and the corresponding mathematical model given in [27]. There are two sets of experimental findings reported in [7]. First, for bacteria cells inoculated in semisolid agar on intermediates of the tricarboxylic acid (TCA) cycle, they form symmetrical rings of spots or stripes. Second, for bacterial growth in liquid medium, when placed in a liquid medium and exposed to intermediates of TCA cycle, *E. coli* and *S. typhimurium* cells arrange themselves into high density aggregates. Unlike the first case, these aggregates rearrange randomly as time evolves and fade on a relatively short time scale compared with the bacteria generation time. These experimental results are different from that shown in [3]. In [3], chemoattractant gradients were created artificially. For the patterns generated by the experiments in [7], attractant chemicals were excreted by the bacterial cells as they grew and moved around.

A non-dimensionalized mathematical model was proposed in [27] for the second kind of experiments in [7]. The model is a system of reaction-diffusion-chemotaxis equations, which is given by

$$\frac{\partial u}{\partial t} = d_u \nabla^2 u - \alpha \nabla \cdot \left[ \frac{u}{(1+v)^2} \nabla v \right] \tag{11}$$

$$\frac{\partial v}{\partial t} = \nabla^2 v + w \frac{u^2}{\mu + u^2}. \tag{12}$$

Here $u$, $v$ and $w$ represent cell density, chemoattractant concentration and succinate concentration respectively. For the simulation, we used the same boundary conditions and initial conditions for $u$ and $v$ as stated in [27]. These are the zero flux boundary condition for both $u$ and $v$, zero initial distribution for $v$ and the random perturbation at $u_0 = 1$ for the initial condition of $u$. The random perturbation was obtained via the $C^{++}$ random number generator routine. The noise amplitude is of order $10^{-1}$.

Figure 3 is our reproduction of Figure 7 in [27]. We used the same parameters as given in [27]. However, we used our algorithm and our own C++ code. It can be observed that our graphs carry the same characteristics as Fig.7 in [27] but generate different patterns, which are typical for randomly generated data.

For our numerical study, we also simulated the first case reported in [7] using the above model. We kept all parameters the same as before except the initial condition for $u$. Here, the initial condition was changed to $u_0(x) = b \exp^{-a|x|}$, where $a > 0$ and $b > 0$ are constant parameters. This is a peak function which simulates the center inoculation of cells at the initial time. Our numerical experiments showed symmetrical rings similar to those shown in Figure 2, but with more layers. However, we failed to produce spots and stripes. This suggests that further study is needed for the modeling of the first case reported in [7].

## 5. **Conclusion.**

In this paper, a two-stage adaptive time-stepping scheme is proposed for solving general reaction-diffusion-chemotaxis systems. This method is fully explicit and is based on estimating the optimal step size at each iteration. Because of its explicitness and simplicity, our scheme combined with any spatial discretization can be applied to highly nonlinear systems. The scheme is easy to

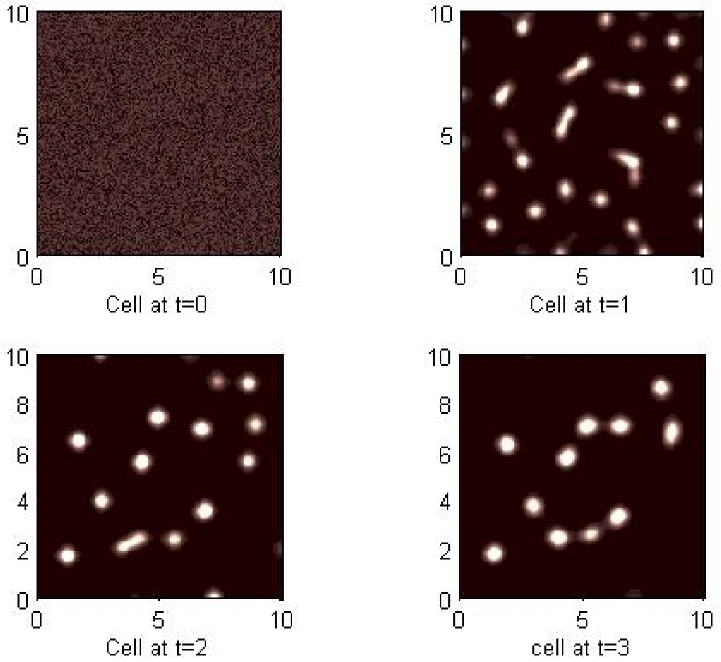

FIGURE 3. Simulations of bacterial bands when the concentration of galactose is higher than that of oxygen and the concentration of oxygen is kept as constant.

implement and is designed specifically for handling spatially nonhomogeneous pattern formation problems. The efficiency and accuracy of the method were shown by mathematical analysis and demonstrated by numerical experiments on model problems and real pattern simulations on bacterial chemotaxis.

6. **Acknowledgement.** The authors would like to thank the reviewer of the first and the second versions of this paper for the very meaningful and critical comments which lead to many significant improvements of this paper. We want to thank the editors of MBE, Dr. Yang Kuang and Dr. Qing Nie for their help and encouragement. We finally want to thank our colleague, Dr. Cliff Weil for his professional help on writing and editing of this paper.

## REFERENCES

[1] K. E. Atkinson, *An Introduction to Numerical Analysis*, 2nd Ed., John Wiley & Sons. Inc., 1989.

[2] J. Adler, *Chemotaxis in Bacteria*, Science, Vol.153(1966), pp708–716.

[3] J. Adler, *Chemotaxis in Bacteria*, Annu. Rev. Biochem. 44(1975), 341–356.

[4] K. E. Atkinson, *An Introduction to Numerical Analysis*, John Wiley, New York, 1978.

[5] J. Bard, *A model for generating aspects of zebra and other mammalian coat patterns*, J. Theoret. Biol., 93(1981), pp. 363–385.

[6] N. F. Britton, *Reaction-Diffusion Equations and Their Application to Biology*, Academic Press, New York, 1986.

[7] E. O. Budrene and H. C. Berg, *Complex patterns formed by motile cells of Escherichia Coli*, Nature, (1991) 349(6310), pp. 630–633.

[8] C. Chiu, F. C. Hoppensteadt, and W. Jäger, *Analysis and computer simulation of accretion patterns in bacterial cultures*, J. Math. Biol., (1994) 32, pp. 841–855.

[9] C. Chiu and N. Walkington, *An ADI method for Hysteretic Reaction-diffusion systems*, SIAM J. Numer. Anal., (1997) 34, No.3, pp. 1185–1206.

[10] C. Chiu and F. C. Hoppensteadt, *Mathematical models and simulations of bacterial growth and chemotaxis in a diffusion gradient chamber*, J. Math. Biol., (2001) 42, pp. 120–144.

[11] C. Chiu and H. Wei, *A Multigrid Method for Pattern Formation Problems in Biology*, Differential and Integral Equations, (2003)Vol.16, No.2., pp201–220.

[12] P. C. Fife, *Mathematical Aspects of Reacting and Diffusing Systems,* Lect. Notes in Biomathematics 28, Berlin Heidelberg New York: Springer 1979.

[13] E. F. Keller and L. A. Segel, *Model for Chemotaxis*, J. Theor. Biol.., 30(1971), pp225–234.

[14] D. Lebietz and H. Maurer, *External optimal control of self-organisation dynamics in a chemotaxis reaction diffusion system*, Syst. Biol., IEE, (2004), pp. 1–8.

[15] S. A. Levin and L. A. Segel, *Pattern Generation in Space and Aspect*, SIAM Rev., 27(1985), pp. 45–67.

[16] N. Li, J. Steiner, and S. Tang, *Convergence and stability analysis of an explicit finite difference method for 2-dimensional reaction-diffusion equations*, J. Austral. Math. Soc. Ser. B, 36 (1994), pp. 234-241.

[17] R. E. Mickens, *Nonstandard finite difference schemes for reaction-diffusion equations*, Numer. Methods Partial Differential Equations, 15 (1999), pp. 201–214.

[18] J. D. Murray, *Mathematical Biology, Biomathematics Texts*, Springer-Verlag, 1989.

[19] J. D. Murray, *A Per-pattern Formation Mechanism for Animal Coat Markings*, J. Theor. Biol., 88 (1981), pp. 161–199.

[20] C. V. Pao, *Numerical analysis of coupled systems of nonlinear parabolic equations*, SIAM J. Numer. Anal., 36 (1999), pp. 393–416.

[21] R. D. Richtmyer and K. W. Morton, *Difference methods for initial value problems*, Interscience, New York, 1967.

[22] J. Schnackenberg, *Simple Chemical reaction systems with limit cycle behaviour*, J. Theor. Biol., 81, (1979), pp. 389–400.

[23] J. Smoller, *Shock Waves and Reaction-Diffusion Equations,* Berlin Heidelberg New York Tokyo: Springer 1983.

[24] J. C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations*, Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, Calif, 1989.

[25] D. Thomas, *Artificial Enzyme Membranes, Transport, Memory, and Oscillatory Phenomena. In: D. Thomas and J.-P. Kernevez (Eds.) Analysis and Control of Immobilized Enzyme Systems.* Berlin Heidelberg New York, Springer, 1975, pp. 115-150.

[26] A. M. Turing, *The Chemical Basis of Morphogenesis.* Phil. Trans. Roy. Soc. Lond. B237, (1952), pp. 37–72.

[27] Rebecca Tyson, R. R. Lubkin and J. D. Murray, *Model and analysis of chemotactic bacterial patterns in a liquid medium*, J. Math. Biol., (1999) 38, pp. 359–375.

[28] Rebecca Tyson, L. G. Stern and Randall J. LeVeque, *Fractional step methods applied to a chemotaxis model*, J. Math. Biol., (2000) 41, pp. 455–475.

[29] Y. M. Wang, *Petrov-Galerkin methods for systems of nonlinear reaction-diffusion equations*, Appl. Math. Comput., 96 (1998), pp. 209–236.

[30] C. Weibull, *The Bacteria*, I. C. Gunsalus and R. Y. Stanier, Eds., Academic Press, New York, 1960, Vol. 1, p. 153.

[31] Mark T. Widman, David Emerson, C. Chiu, R. Mark Worden *Modeling Microbial Chemotaxis in a Diffusion Gradient Chamber*, Biotechnology and Bioengineering, Vol.55, No.1, 1997.

[32] J. Yu, *A fully explicit optimal two-stage numerical scheme for solving reaction-diffusion-chemotaxis systems*, Ph.D. thesis, Michigan State Univ., 2005.

Received on July 22, 2005. Accepted on December 1, 2006.

*E-mail address*: chiu@math.msu.edu
*E-mail address*: yujuiling@hotmail.com