

A Positive Splitting Method for Mixed Hyperbolic–Parabolic Systems

Alf Gerisch,¹ David F. Griffiths,² Rüdiger Weiner,¹ Mark A. J. Chaplain,²

¹*Institut für Numerische Mathematik, Fachbereich Mathematik und Informatik, Martin-Luther Universität Halle-Wittenberg, Postfach, 06099 Halle (Saale), Germany*

²*Department of Mathematics, University of Dundee, Dundee, DD1 4HN, UK*

Received 7 January 1999; accepted 13 March 2000

In this article we present a method of lines approach to the numerical solution of a system of coupled hyperbolic–parabolic partial differential equations (PDEs). Special attention is paid to preserving the positivity of the solution of the PDEs when this solution is approximated numerically. This is achieved by using a flux-limited spatial discretization for the hyperbolic equation. We use splitting techniques for the solution of the resulting large system of stiff ordinary differential equations. The performance of the approach applied to a biomathematical model is compared with the performance of standard methods. © 2001 John Wiley & Sons, Inc. *Numer Methods Partial Differential Eq* 17: 152–168, 2001

Keywords: mixed hyperbolic–parabolic PDE system; finite difference approximation; method of lines; splitting methods; positive methods

I. INTRODUCTION

We consider the hyperbolic–parabolic system of partial differential equations (PDEs)

$$\frac{\partial n}{\partial t} = -\nabla \cdot (nG(c)\nabla c) + g_1(n, c) \quad (1.1a)$$

$$\frac{\partial c}{\partial t} = \Delta c + g_2(n, c) \quad (1.1b)$$

for the unknown functions $n(x, y, t)$ and $c(x, y, t)$ on the interior of the unit square $(0, 1) \times (0, 1)$ in space and for $t \in (0, t_{final}]$ in time, where ∇ denotes the gradient operator $(\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$ and Δ is the Laplace operator. The functions G , g_1 and g_2 are prescribed functions of n and c . The quantity n is convected by a velocity field, which depends nonlinearly on the quantity c and its gradient. This kind of term arises frequently in biomathematical models, if processes like

Correspondence to: David F. Griffiths, Department of Mathematics, University of Dundee, Dundee, DD1 4HN, UK (e-mail: dfg@maths.dundee.ac.uk)

© 2001 John Wiley & Sons, Inc.

chemotaxis or haptotaxis are present [1-3]. For $G(c) \geq 0$, n is convected towards areas of larger c values. The quantity c is the solution of the diffusion equation with nonlinear source term (1.1b) but might, in other applications, be described by a differential equation of different type [for example, an ordinary differential equation (ODE)]. Beside this coupling of the equations, there is also a coupling of the system of PDEs via the reaction terms g_1 and g_2 . The equations may be termed a “reaction-diffusion-convection” system.

The system of PDEs is supplied with initial conditions

$$n(x, y, 0) = n_0(x, y), \quad c(x, y, 0) = c_0(x, y) \tag{1.2}$$

and appropriate boundary conditions consisting of one condition for c at every point of the boundary and Dirichlet boundary conditions for n on inflow boundaries. There are no boundary conditions for n on outflow boundaries. In this article, we concentrate on the following boundary conditions:

$$n(1, y, t) = n_r(y) \quad (\text{assuming inflow on the right boundary}) \text{ and} \tag{1.3}$$

$$c(0, y, t) = c_l(y), \quad c(1, y, t) = c_r(y), \quad c_y(x, 0, t) = c_y(x, 1, t) = 0. \tag{1.4}$$

The functions n_0, c_0, n_r, c_l and c_r are given such that initial and boundary conditions are consistent. Slight adjustments in the algorithm are necessary to accomodate other boundary conditions.

Our motivation for considering problems of this kind comes from biomathematical models describing tumor-induced angiogenesis [9, 1]. Our test problem is such a model and is derived and described in greater detail in [4, 5].

Example 1. Equations (1.1a, 1.1b) with

$$G(c) = \kappa, \tag{1.5}$$

$$g_1(n, c) = \mu n(1 - n) \max \{0, c - c^*\} - \beta n, \tag{1.6}$$

$$g_2(n, c) = -\frac{\alpha n c}{\gamma + c} - \lambda c, \tag{1.7}$$

initial conditions and boundary conditions

$$n_0(x, y) = \begin{cases} 1 & : x \geq 0.95 \text{ and } y \in \bigcup_{i=1}^4 [y_i \pm \frac{35}{1000}], y_i = 0.2, 0.36, 0.5, 0.64, 0.8 \\ 0 & : \text{otherwise} \end{cases} \tag{1.8}$$

$$c_0(x, y) = \cos\left(\frac{\pi}{2}x\right), \tag{1.9}$$

$$n_r(y) = n_0(1, y), \quad c_l(y) = 1, \quad c_r(y) = 0, \tag{1.10}$$

and parameter values (typical for the application [4])

$$\alpha = 10, \beta = 4, \gamma = 1, \kappa = 0.7, \lambda = 1, \mu = 100, c^* = 0.2. \tag{1.11}$$

A variety of approximate solution methods is available for pure hyperbolic or pure parabolic PDEs and we make use of these by decoupling the system (1.1a, 1.1b) through splitting techniques. In particular, we treat the hyperbolic equation (or parts arising from it) with explicit methods and the parabolic equation implicitly.

A fully discrete splitting method was proposed [5, 6] in which the easier subproblems (1.1a) and (1.1b) were solved by using Strang splitting in conjunction with MacCormack’s method for (1.1a) and ADI methods for (1.1b). However, this approach was found to be unsatisfactory,

despite its computational economy, because it was prone to generating spurious oscillations and consequent instability through negative solution values. An alternative approach is described in Section II based on the method of lines (MOL). Here we consider the case that n and c describe concentrations or densities, and, as such, are always nonnegative. We, therefore, expect the functions G , g_1 , and g_2 and the boundary conditions to be of such a structure that the exact solution of the system (1.1a,1.1b) is nonnegative for all times $t > 0$ provided that the initial conditions for n and c are nonnegative.

The first- and second-order spatial derivatives of c in the system are approximated by standard second-order central differences, but the gradient operator in (1.1a) is approximated by a flux-limited second-order discretization similar to the one proposed in [7]. The aim of the limiting procedure is that the nonnegativity property of the PDEs solution is carried over to the solution of the ODE system obtained by the spatial approximation. The result is a large stiff system of ODEs, which is solved by splitting the right-hand side into the terms representing the convection and the diffusion/reaction of the underlying PDE system. Within this splitting, explicit Runge–Kutta methods (RKMs) are used to solve the resulting systems of ODEs when the right-hand side is derived from the convection discretization, while implicit schemes are employed for ODE systems when the right-hand side represents diffusion and reaction (these are themselves splitting schemes). It is shown that the overall method is second-order in time.

Section III presents numerical experiments demonstrating the performance of the proposed MOL scheme on Example I. and a comparison is made with other methods. Finally, Section IV contains a discussion and gives our main conclusions.

An equidistant linear grid covers the spatial domain having M intervals in each spatial direction with grid spacing $h = 1/M$. The index i refers always to grid values along the y -axis and the index j always to grid values along the x -axis. We define $x_j = jh$, $y_i = ih$ and denote the continuous time approximations to $n(x_j, y_i, t)$ and $c(x_j, y_i, t)$ by $N_{ij}(t)$ and $C_{ij}(t)$, respectively. The dependence on t is omitted, if it is clear from the context. We use an adaptive grid in time with grid points $t_{k+1} = t_k + \tau_k$ for $k = 0, 1, \dots$, and time-steps τ_k generated by an appropriate step-size control ($t_0 = 0$). In the following, the variable time-step size τ_k is denoted by τ (without subscript). N_{ij}^k and C_{ij}^k denote the point approximations to $n(x_j, y_i, t_k)$ and $c(x_j, y_i, t_k)$, respectively. The subscript or superscript k always refers to points on the time scale.

II. METHOD OF LINES APPROACH

The MOL is a standard approach used for the solution of time-dependent PDEs. The PDE is replaced by a large, in general, stiff system of ODEs by approximating the spatial derivatives on a finite grid in space. The solution of the ODE system is a time-continuous function for each spatial grid point and, hence, defines an approximation to the solution of the PDE restricted to the grid.

Section II.A describes the spatial approximation of (1.1a–1.1b). We assume in the following that n and c describe nonnegative quantities, a property that may be destroyed unless space discretizations are carried out carefully. Our approach leads to positivity as well as second-order accuracy away from extremal points of the solution.

In Section II.B, a second-order splitting scheme for the numerical solution of the ODE system is developed. It is specially designed with the ODE system obtained in Section II.A in mind, and we address issues related to computational efficiency and positive solutions.

A. Approximation of the Spatial Derivatives

As stated in the Introduction, we consider the case that the solutions n and c of our PDE system (1.1a–1.1b) are nonnegative for all times $t \geq 0$. This property should carry over to the ODE system obtained by spatially discretizing the PDE.

Consider the initial value problem (IVP) for ODE systems:

$$u'(t) = f(t, u(t)), \quad t \geq t_0, \quad u(t_0) = u_0, \quad t_0 \in \mathbb{R}, \quad u_0 \in \mathbb{R}^m, \quad (2.1)$$

where $f : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ has the property

$$f \text{ is continuous and (2.1) has a unique solution for all } t_0 \text{ and for all } u_0. \quad (2.2)$$

This property holds, for instance, if f is globally Lipschitz continuous in the second argument.

Definition. (Positive ODE system, positive semi-discretization) *The ODE in (2.1) and the IVP (2.1) are said to be positive if f has the property (2.2) and $u(t) \geq 0$ holds for all $t \geq t_0$ whenever $u_0 \geq 0$. A semi-discretization of a given PDE (with nonnegative solution) is called positive if it leads to a positive ODE system.*

Lemma 2.1. (See [8].) *Let f satisfy condition (2.2). The IVP (2.1) associated with this function is positive if and only if*

$$v_i = 0, \quad v_j \geq 0 \text{ for all } j \neq i \quad \Rightarrow \quad f_i(t, v) \geq 0$$

holds for all t and any vector $v \in \mathbb{R}^m$ and all $i = 1, 2, \dots, m$.

We now look at the spatial discretization of Eqs. (1.1a) and (1.1b) in turn.

Parabolic Equation (1.1b) The semi-discretization of a linear diffusion equation (with periodic boundary conditions and without source term) is considered in [9, p. 28] with respect to positivity. The positivity requirement leads to an order barrier of two, and it is shown that the standard second-order central difference operator leads to a minimal error coefficient. Guided by these results, Eq. (1.1b) is discretized in space as

$$\frac{d C_{ij}}{dt} = \frac{1}{h^2} (C_{i,j+1} + C_{i,j-1} + C_{i+1,j} + C_{i-1,j} - 4C_{ij}) + g_2(N_{ij}, C_{ij}) \quad (2.3)$$

for $i = 0, 1, \dots, M$ and $j = 1, 2, \dots, M - 1$. The given boundary values of c are used on the left and right boundary, while rows of fictitious grid points are introduced at $y = -h$ and $y = 1 + h$ in order to approximate the homogeneous Neumann boundary conditions at $y = 0$ and $y = 1$ with second-order accuracy by central differences. This process leads to $C_{-1,j} = C_{1j}$ and $C_{M+1,j} = C_{M-1,j}$ for all $j = 1, 2, \dots, M - 1$. The second-order accurate spatial discretization of (1.1b) by (2.3) is now fully defined.

Lemma 2.2. *The ODE system defined by (2.3) is positive, if the prescribed Dirichlet boundary values are nonnegative, the function g_2 is Lipschitz continuous, and if $g_2(n, 0) \geq 0$ for all possible values of n .*

Proof. This result follows from Lemma 2.1. ■

Remark. *The condition required by Lemma 2.2 is satisfied for the function g_2 of Example 1. and for the given boundary conditions.*

Hyperbolic Equation (1.1a) For the semi-discretization of Eq. (1.1a), we again focus on positivity of the resulting ODE system. A first-order upwind space discretization would give us such a system, but would also introduce an excessive amount of numerical diffusion, which would have a first-order effect on the speed of travelling waves and might even preclude the existence of monotonic wave profiles unless the grid sizes are sufficiently small [10]. On the other hand, higher-order (linear) space discretizations are prone to introducing “wiggles” and possibly negative values in the solution, which we would like to avoid, because they can lead to instabilities through the nonlinear source term. Flux limiters provide a popular means of combining high-order methods and oscillation-free solutions. We follow the construction of a positive semi-discretization through flux limiters given in [7].

The approximation of two-dimensional convection is constructed from the one-dimensional setting

$$\frac{\partial w(x, y, t)}{\partial t} = -(v(x, y, t)w(x, y, t))_x, \quad t \geq 0, \quad (x, y) \in (0, 1)^2, \quad (2.4)$$

$$w(x, y, 0) = w_0(x, y) \geq 0, \quad w(1, y, t) = w_1(y, t) \geq 0, \quad (2.5)$$

where the quantity w is convected only in the x -direction by the velocity field v , which is assumed to be known for the following discussion. We suppose an inflow boundary at $x = 1$ and an outflow boundary at $x = 0$ and, therefore, require $v(1, y, t) \leq 0$ and $v(0, y, t) \leq 0$. The solution of this PDE then satisfies $w(x, y, t) \geq 0$ for $t \geq 0$ and $(x, y) \in (0, 1)^2$.

The function $\tilde{f}(x, y, t) = v(x, y, t)w(x, y, t)$ is the flux function of (2.4) and $W_{ij}(t)$ denotes a time-continuous approximation of $w(x_j, y_i, t)$. We define the semi-discrete flux function $f_{ij}(t) = v(x_j, y_i, t)W_{ij}(t)$. Let $F_{i,j+1/2}$ be a consistent approximation of the flux $\tilde{f}(x_{j+1/2}, y_i, t)$ depending on the neighboring (semi-discrete) fluxes f_{il} , $l = 0, 1, \dots, M$. Here consistency means that $F_{i,j+1/2} = \tilde{f}(x_{j+1/2}, y_i, t)$ whenever f_{il} , $l = 0, 1, \dots, M$, is set to $\tilde{f}(x_{j+1/2}, y_i, t)$ [11]. Equation (2.4) is now approximated in space by the semi-discrete, conservative expression

$$\frac{dW_{ij}(t)}{dt} = -\frac{1}{h} (F_{i,j+1/2} - F_{i,j-1/2}), \quad (2.6)$$

where we have yet to define the terms $F_{i,j+1/2}$ for $j = -1, 0, 1, \dots, M - 1$. This is our next concern.

The general flux expression

$$F_{i,j+1/2} = f_{ij} + \frac{1}{2}\Phi(r_{i,j+1/2})(f_{ij} - f_{i,j-1}) \quad (2.7)$$

is considered for positive velocities $v(x_j, y_i, t)$ in [7]. The function Φ is called the limiter function, and the ratio r monitors the smoothness of the flux and is defined as

$$r_{i,j+1/2} = \frac{f_{i,j+1} - f_{i,j}}{f_{i,j} - f_{i,j-1}}. \quad (2.8)$$

Remark. In actual computations, we add $\varepsilon \ll 1$ (we use 10^{-30}) to both numerator and denominator to avoid division by zero in uniform flow regions [12].

The expression corresponding to (2.7) for negative velocity $v(x_j, y_i, t)$ is obtained by reflecting all indices that appear in $f_{i\cdot}$ about $j + 1/2$ yielding

$$F_{i,j+1/2} = f_{i,j+1} + \frac{1}{2}\Phi(r_{i,j+3/2}^{-1})(f_{i,j+1} - f_{i,j+2}). \quad (2.9)$$

We favor the flux interpolation approach in contrast to the state interpolation approach used in [7]. The reason is that, in our objective Eq. (1.1a), the velocity depends on c , whose values are known only at grid points.

Inserting (2.7) and (2.9) in Eq. (2.6) gives

$$\frac{dW_{ij}(t)}{dt} = -\frac{1}{h} \left[\left(1 + \frac{1}{2}\Phi(r_{i,j+1/2}) \right) - \frac{\Phi(r_{i,j-1/2})}{2r_{i,j-1/2}} \right] (f_{ij} - f_{i,j-1}) \quad (2.10)$$

and

$$\frac{dW_{ij}(t)}{dt} = -\frac{1}{h} \left[\left(1 + \frac{1}{2}\Phi(r_{i,j+1/2}^{-1}) \right) - \frac{\Phi(r_{i,j+3/2}^{-1})}{2r_{i,j+3/2}^{-1}} \right] (f_{i,j+1} - f_{ij}), \quad (2.11)$$

respectively. Following [7], we require the bracketed terms to be nonnegative, that is, $1 + \frac{1}{2}\Phi(a) - \frac{\Phi(b)}{2b} \geq 0$ for all $a, b \in \mathbb{R}$ in order to obtain a positive semi-discretization (see below). Sufficient conditions on Φ are

$$\Phi(r) = 0 \text{ if } r \leq 0, \text{ and } 0 \leq \Phi(r) \leq \delta, \quad \Phi(r) \leq 2r \text{ if } r > 0, \quad (2.12)$$

where $\delta > 0$ serves as a free parameter, which can be used to obtain higher accuracy near peaks [7].

The limiter Φ should, in view of the conditions in Lemma 2.1, be chosen as a Lipschitz continuous function. The maximum limiter proposed in [7] is based on the so called κ -methods (which include the second-order central, second-order upwind, and third-order upwind biased approximation). Although this limiter is Lipschitz continuous, there are points $r \geq 0$ at which it is not differentiable. We, therefore, propose using the limiter function due to van Leer, see, e.g., [11], given by

$$\Phi_{VL}(r) = \frac{|r| + r}{1 + |r|}, \quad (2.13)$$

which satisfies the conditions (2.12) for $\delta = 2$, is Lipschitz continuous and continuously differentiable for all $r \neq 0$.

Remark. *Numerical experiments indicate the superiority of van Leer's limiter over limited κ -methods with respect to positivity. However, if positivity of the numerical solution is less important than accuracy, then the limiter proposed in [7] should be used.*

Lemma 2.3. *Suppose that $F_{i,j\pm 1/2}$ is defined by (2.7) or (2.9) depending on the sign of $v(x_j, y_i, t)$ and uses van Leer's limiter (2.13). If, depending on the sign of $v(x_j, y_i, t)$, $r_{i,j\pm 1/2} > 0$ or $r_{i,j+1\pm 1/2} > 0$ (that is, v has the same sign locally around the point (x_j, y_i) away from local extrema of w), then the approximation (2.6) is second-order accurate in space for sufficiently smooth functions w and v , i.e.,*

$$\frac{1}{h} (F_{i,j+1/2} - F_{i,j-1/2}) - (v(x_j, y_i, t)w(x_j, y_i, t))_x = \mathcal{O}(h^2). \quad (2.14)$$

Proof. Taylor expansion of the expressions. ■

The lemma ensures that the approximation is second-order in smooth, monotone regions of the solution w .

Next we describe exactly which of the flux approximations (2.7, 2.9) is to apply for the semi-discretization at a specific point (x_j, y_i) in order to obtain a positive semi-discretization.

1. $W_{ij}(t) > 0$. In this case, Lemma 2.1 does not impose any conditions and we compute $F_{i,j\pm 1/2}$ by applying (2.7), if $v(x_j, y_i, t) \geq 0$, and (2.9), if $v(x_j, y_i, t) < 0$.
2. $W_{ij}(t) = 0$. Here Lemma 2.1 requires $\frac{dW_{ij}(t)}{dt} \geq 0$ for the right-hand side of Eq. (2.6). From $W_{ij}(t) = 0$, it follows that $f_{ij} = 0$, and we decide upon the approximation depending on the sign of v at neighboring points.
 - a. $v(x_{j-1}, y_i, t) \geq 0$: Use (2.7) to compute $F_{i,j\pm 1/2}$ giving Eq. (2.10). The bracketed expression is nonnegative and, hence, so is the right-hand side.
 - b. Else if $v(x_{j+1}, y_i, t) \leq 0$: Use (2.9) to compute $F_{i,j\pm 1/2}$ giving Eq. (2.11). The bracketed expression is nonnegative and, hence, so is the right-hand side.
 - c. $v(x_{j-1}, y_i, t) < 0$ and $v(x_{j+1}, y_i, t) > 0$. Set $\frac{dW_{ij}(t)}{dt} = 0$ consistent with the PDE (see the remark below for a discussion).

Remark. *The solution of (2.4–2.5) is nonnegative and case 2c is treated such that the resulting ODE system also has a nonnegative solution (applying (2.10) or (2.11) in this case would lead to a violation of Lemma 2.1).*

Suppose, at time t and point (x_j, y_i) , that case 2c occurs and that $w(x_j, y_i, t) = 0$ and that the functions w and v are sufficiently smooth. Then also $w_x(x_j, y_i, t) = 0$ and, hence, $w_t(x_j, y_i, t) = 0$ by (2.4) justifying the choice in case 2c. Case 2c would be treated as case 2a or 2b on a sufficiently fine spatial grid, except in the case where $v(x_j, y_i, t) = 0$ and $v_x(x_j, y_i, t) > 0$ holds.

We now consider the semi-discretization near the boundary. Figure 1 shows the stencils of the semi-discretization for both positive and negative velocities for grid points away from the boundary.

We first consider the inflow boundary at $x = 1$ and the discretization at the point (x_{M-1}, y_i) (the values of $w(x_M, y_i, t)$ are given by the Dirichlet boundary condition). Points further to the left constitute no difficulties. If (2.7) is applied for the discretization of the convection term, then all required fluxes f_{il} , $l = M - 3, \dots, M$ are known. The approximation (2.9) is used if $W_{i,M-1} > 0$ and $v(x_{M-1}, y_i, t) < 0$, or if $W_{i,M-1} = 0$ and $v(x_M, y_i, t) \leq 0$ (the velocity is always nonpositive on an inflow boundary at $x = 1$ and the case 2c cannot occur). The application of (2.9) requires the flux $f_{i,M+1}$, which we define by second-order extrapolation, namely $f_{i,M+1} := 3f_{i,M} - 3f_{i,M-1} + f_{i,M-2}$. This leads to a semi-discretization satisfying the conditions of Lemma 2.1.

We now consider the outflow boundary at $x = 0$ for which we have $v(0, y_i, t) \leq 0$. We define $f_{i,-1}$ by second-order extrapolation: $f_{i,-1} = 3f_{i,0} - 3f_{i,1} + f_{i,2}$. This allows us to apply approximation (2.9) at the grid point $(0, y_i)$ if $W_{i,0} > 0$ or if $W_{i,0} = 0$, and $v(x_1, y_i, t) \leq 0$ and again leads to a positive semi-discretization. In the exceptional case $W_{i,0} = 0$ and $v(x_1, y_i, t) > 0$, we set $\frac{dW_{ij}(t)}{dt} = 0$ (case 2c). At the grid point (x_1, y_i) , we apply (2.9) if $W_{i,1} > 0$ and $v(x_1, y_i, t) < 0$, or if $W_{i,1} = 0$ and $v(x_2, y_i, t) \leq 0$. If $W_{i,1} > 0$ and $v(x_1, y_i, t) \geq 0$, or if $W_{i,1} = 0$ and $v(x_0, y_i, t) \geq 0$ (that is = 0 because of the outflow boundary), then we apply (2.7) using the defined value of $f_{i,-1}$. All these possibilities lead to a positive semi-discretization. In the remaining exceptional case, we set the right-hand side of the ODE to zero.



FIG. 1. Stencils of the semidiscretization at (x_j, y_i) using (2.7), left, and (2.9), right.

Finally, the scheme we have developed is applied to the hyperbolic equation (1.1a) of the PDE system. Again we consider discretization of convection in the x -direction only. The corresponding y -part is computed in a similar manner and added, with the term $g_1(N_{ij}, C_{ij})$, to the right-hand side of the semi-discretization.

The velocity function v is now defined as $v(x, y, t) = G(c(x, y, t))c_x(x, y, t)$, and the function n is the counterpart of the convected quantity w above. We apply the scheme just described to this new situation, and all that remains is to approximate the velocity v at spatial grid points. We define second-order approximations of $c_x(x_j, y_i, t)$, $i = 1, \dots, M$, by

$$\begin{aligned} (C_{i,0})_x &= \frac{1}{h}(C_{i,0} - 3C_{i,1} + 2C_{i,2}), \\ (C_{ij})_x &= \frac{1}{2h}(C_{i,j+1} - C_{i,j-1}), \quad j = 1, 2, \dots, M-1, \\ (C_{i,M})_x &= -\frac{1}{h}(C_{i,M} - 3C_{i,M-1} + 2C_{i,M-2}), \end{aligned}$$

where the boundary values of c are used, if necessary. The approximation for $v(x_j, y_i, t)$ is then given by $v(x_j, y_i, t) \approx G(C_{ij})(C_{ij})_x$.

Lemma 2.4. *The described semi-discretization of Eq. (1.1a) leads to a positive ODE system, if the prescribed Dirichlet boundary values are nonnegative, the function g_1 is Lipschitz continuous, and if $g_1(c, 0) \geq 0$ for all possible values of c .*

Proof. This result follows from Lemma 2.1. ■

Remark. *The condition required by Lemma 2.4 is satisfied for the function g_1 of Example 1 and for the given boundary conditions.*

The ODEs for $N_{ij}(t)$ and $C_{ij}(t)$ constitute a large autonomous system of ODEs with initial conditions $N_{ij}(0) = n_0(x_j, y_i)$ and $C_{ij}(0) = c_0(x_j, y_i)$. An appropriate numerical scheme for the solution of this system is developed in the next section.

B. Time-Split Scheme for the ODE System

The discretization of the spatial derivatives in the previous section results in a large m -dimensional system of ODEs in autonomous form:

$$u'(t) = f_1(u(t)) + f_2(u(t)), \quad t \in (0, t_{final}], \quad u(0) = u_0. \quad (2.15)$$

The vector $u(t)$ contains all unknown time-continuous functions $N_{ij}(t)$ and $C_{ij}(t)$ in an appropriate order. The function $f_2(u(t))$ represents the approximation of the diffusion and reaction terms in (1.1a, 1.1b), and $f_1(u(t))$ the discretized convection. We regard f_2 as the stiff part and f_1 as the nonstiff part of the ODE system.

In this section, we develop a method for the solution of (2.15), which is both computationally efficient and positive (in the sense that it generates nonnegative approximations to the solution of a positive ODE system). An implicit method should be used for the solution of $u' = f_2(u)$ because of the stiffness of this equation, otherwise stability of the scheme would require very small time-steps and, hence, would lead to excessive computational cost. However, we do not employ an implicit solver for the ODE $u' = f_1(u)$, since a restriction on the time-step would be needed to maintain positivity (except under certain special circumstances) and a major advantage of implicitness would be lost. Furthermore, experiments indicate that methods requiring the Jacobian of the right-hand side are less appropriate for the solution of ODE systems resulting

from a (nonsmooth) flux-limited spatial discretization of the advection equation. Typically, there are many rejected steps, which is not the case if unlimited discretizations are used (but then we have negative values in the solution). The cause of this behavior is the nonsmoothness of the right-hand side of the ODEs, and we, therefore, apply an explicit method with good positivity properties.

With these two competing concepts (implicitness and stability vs. explicitness and positivity) in mind, it seems natural to use a splitting approach for the solution of (2.15). The solution of (2.15) at time $t_k + \tau$ with initial data u_k at time t_k can be written as

$$\begin{aligned} u(t_k + \tau) &= u_k + \int_{t_k}^{t_k + \tau} [f_1(u(\xi)) + f_2(u(\xi))] d\xi \\ &= u_k + \underbrace{\int_{t_k}^{t_k + \frac{\tau}{2}} f_1(u(\xi)) d\xi}_{=z_1(t_k + \frac{\tau}{2})} + \int_{t_k}^{t_k + \tau} f_2(u(\xi)) d\xi + \int_{t_k + \frac{\tau}{2}}^{t_k + \tau} f_1(u(\xi)) d\xi, \end{aligned}$$

where $z_1(t_k + \frac{\tau}{2})$ is the solution of $z_1'(t) = f_1(z_1(t))$, $z_1(t_k) = u_k$ at $t = t_k + \frac{\tau}{2}$. Therefore, we propose the following (Strang) splitting step for the solution of (2.15):

$$\begin{aligned} z_1'(t) &= f_1(z_1(t)), & z_1(t_k) &= u_k & \Rightarrow z_1(t_k + \frac{\tau}{2}), \\ z_2'(t) &= f_2(z_2(t)), & z_2(t_k) &= z_1(t_k + \frac{\tau}{2}) & \Rightarrow z_2(t_k + \tau), \\ z_3'(t) &= f_1(z_3(t)), & z_3(t_k + \frac{\tau}{2}) &= z_2(t_k + \tau) & \Rightarrow z_3(t_k + \tau). \end{aligned} \quad (2.16)$$

Theorem 2.5. *If each ODE system in the splitting scheme (2.16) is solved with a method of consistency order two or higher and the approximation to $z_3(t_k + \tau)$ is used as an approximation for $u(t_k + \tau)$, then the splitting scheme (2.16) for the ODE (2.15) is consistent with order two.*

Proof. Straightforward by Taylor expansion. ■

Remark. *If the right-hand side of an ODE system is split into more than two parts, then the theorem, applied recursively to an appropriately extended splitting scheme, also gives second-order consistency. This can be useful for multidimensional diffusion discretization or for separate treatment of source terms.*

As stated at the beginning of this section, we treat ODEs with right-hand side f_1 explicitly and those with f_2 implicitly. We can, therefore, regard this splitting scheme as an implicit–explicit (IMEX) scheme. Such schemes have appeared frequently in recent literature, e.g., [13–15]. An important question in these investigations is the linear stability of IMEX schemes. The scalar test equation

$$u'(t) = \mu u(t) + \lambda u(t), \quad \mu, \lambda \in \mathbb{C} \quad (2.17)$$

is considered in [15] and the following question is investigated:

Is the IMEX method stable for given eigenvalues μ, λ if $\tau\mu$ is inside the stability domain of the underlying explicit scheme and $\tau\lambda$ is inside the stability domain of the underlying implicit scheme?

The answer for the methods considered in [15] is negative, except for the IMEX Euler method of order one. For the splitting method (2.16) applied to the test Eq. (2.17), we obtain the stability function given by

$$R(\tau\mu, \tau\lambda) = R^{(1)}\left(\frac{\tau\mu}{2}\right) R^{(2)}(\tau\lambda) R^{(1)}\left(\frac{\tau\mu}{2}\right),$$

where $R^{(i)}$ is the stability function of the method used to solve ODEs with the right-hand side f_i , ($i = 1, 2$). The following result is then immediate.

Theorem 2.6. *The splitting algorithm (2.16) is stable with respect to the test equation (2.17), if $\frac{\tau\lambda}{2}$ is inside the stability domain of the underlying explicit scheme and $\tau\lambda$ is inside the stability domain of the underlying implicit scheme.*

Note that the stability domain of the explicit method with stability function $R^{(1)}$ is effectively enlarged by a factor two, because it is applied with time-steps $\tau/2$. The specific choice of methods for the solution of the ODEs in (2.16) is considered next.

Explicit Method A positive, nonstiff problem of the form

$$u'(t) = f_1(u(t)), \quad u(t_k) = u_k \tag{2.18}$$

has to be solved in steps one and three of the splitting scheme (2.16). We use an explicit RKM to advance the initial condition over a time-step τ . Theorem 2.5 states that we require only a second-order accurate method in order to obtain overall second-order consistency of the splitting scheme, so we restrict ourselves to such methods. A more important question is how to restrict the time-step τ so that the solution of the RKM is nonnegative.

We look at positivity of the method applied to the constant coefficient problem with $f_1(u) = Pu$. Constant coefficient problems are positive if and only if the off-diagonal elements of the matrix P are nonnegative [16]. The question of positivity of a method reduces to the question of absolute monotonicity of its stability function $R(x)$. (If implicit methods were used, then it would be necessary that the matrix P have nonpositive diagonal entries and have no eigenvalues on the positive real axis in order for the following theory to apply.)

Definition. ([16], absolutely monotonic, threshold factor, positivity threshold) *A rational function $R(x)$ is called absolutely monotonic at $x \in \mathbb{R}$ if and only if R and all its derivatives are nonnegative in x . The threshold factor of R , denoted by $T(R)$, is defined as*

$$T(R) = \sup\{r \mid r = 0 \text{ or } (r > 0, R \text{ is absolutely monotonic } \forall x \in [-r, 0])\}.$$

Let $\alpha > 0$. The (linear) positivity threshold, $\tau_{max}(\alpha)$, of a method with stability function R is defined as the largest $\tau_0 \in [0, \infty]$ such that the method is positive for all step sizes $\tau \in (0, \tau_0)$ on all positive systems (2.18) with $f_1(u) = Pu$ and diagonal entries $p_{ii} \geq -\alpha$.

With these definitions, it follows from results of Bolley and Crouzeix [17] that

$$\tau_{max}(\alpha) = T(R) \frac{1}{\alpha}.$$

Absolute monotonicity of polynomials is investigated in [16] and, for s -stage explicit RKMs of order s , it is shown that $T(R) = 1$. The optimal stability polynomial with regard to absolute monotonicity for three stage, second-order methods is $R(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{12}$ for which $T(R) = 2$. In other words, the largest allowable step-size for positivity of the method is doubled at the cost of only one additional evaluation of the right-hand side of the ODE. An examination of the domain of linear stability of this function shows that it is also larger than the corresponding domain of a 2-stage, second-order method. The solution of the order conditions for order two and the optimality conditions on the stability function for a 3-stage explicit RKM are given in the form of a Butcher array in Fig. 2 (left). We use the free parameters (a_{32}, b_2, b_3) in the method

$$\left| \begin{array}{ccc|ccc}
 0 & & & & & \\
 \frac{1}{12b_3a_{32}} & & 0 & & & \\
 -\frac{b_2+12b_3^2a_{32}^2-6b_3a_{32}}{12b_3^2a_{32}} & a_{32} & 0 & b_2, b_3, a_{32} \in \mathbb{R}, & & \\
 \hline
 1-b_2-b_3 & b_2 & b_3 & b_3, a_{32} \neq 0 & & \\
 \end{array} \right| \quad \left| \begin{array}{ccc}
 0 & & \\
 \frac{2}{9} & 0 & \\
 \frac{1}{6} & \frac{1}{2} & 0 \\
 \hline
 \frac{1}{4} & 0 & \frac{3}{4}
 \end{array} \right|$$

FIG. 2. Butcher array of a general 3-stage, second order explicit RKM with optimal linear positivity (left) and a specific choice which satisfies one of the two third-order conditions (right).

to satisfy one of the two third-order conditions. The Butcher array on the right of Fig. 2 gives one possible choice of these parameters, and it is this method that is used for the solution of the ODEs in steps one and three of the splitting scheme (2.16). We note that we cannot satisfy both third-order conditions without reducing the threshold factor of the method to $T(R) = 1$.

Implicit Method The ODE system

$$z_2'(t) = f_2(z_2(t)), \quad z_2(t_k) = \tilde{z}_1 \tag{2.19}$$

has to be solved from t_k to $t_k + \tau$ in step two of the splitting algorithm (2.16). It is the semi-discretization of a heat conduction equation and, because of its stiffness, we solve it with a linearly implicit scheme. The only implicit relations in these methods are linear equation systems involving the Jacobian of the right-hand side of the ODE system. The Jacobian J_2 of f_2 is broadly banded in our case (i.e., a bandwidth $\mathcal{O}(M)$ on an $M \times M$ grid, whatever arrangement of the components of z_2 is used) and, therefore, the application of direct solvers for the linear systems would ruin the efficiency of the splitting scheme (2.16). Iterative methods offer an alternative, but we favor another approach. The right-hand side of the ODE can be written as $f_2 = f_{2a} + f_{2b}$, where f_{2a} contains the terms arising from the discretization of c_{yy} and f_{2b} contains all the remaining terms. The Jacobian J_{2a} of f_{2a} is a tridiagonal matrix, if the components of z_2 are appropriately arranged and the Jacobian J_{2b} of f_{2b} is a pentadiagonal matrix for a different arrangement of z_2 . Therefore, from the point of view of computational efficiency, it seems worthwhile to consider linearly implicit splitting methods. We employ a linearly implicit variant of the trapezoidal splitting method [18, 19]. This method involves only the solution of two linear systems (one with J_{2a} , the other with J_{2b}) per time-step. A time-step τ of this method generates approximation \tilde{z}_2 of $z_2(t_k + \tau)$ by

$$\begin{aligned}
 \tilde{z}_2^{(1)} &= \tilde{z}_1 + \frac{\tau}{2} f_{2a}(\tilde{z}_1) \\
 \tilde{z}_2^{(2)} &= \tilde{z}_2^{(1)} + \frac{\tau}{2} f_{2b}(\tilde{z}_2^{(1)}) \\
 \tilde{z}_2^{(3)} &= \tilde{z}_2^{(2)} + \frac{\tau}{2} \left(I - \frac{\tau}{2} T_{2b} \right)^{-1} f_{2b}(\tilde{z}_2^{(2)}) \\
 \tilde{z}_2 &= \tilde{z}_2^{(3)} + \frac{\tau}{2} \left(I - \frac{\tau}{2} T_{2a} \right)^{-1} f_{2a}(\tilde{z}_2^{(3)}),
 \end{aligned}$$

where

$$T_{2a} = \frac{\partial f_{2a}}{\partial u}(\tilde{z}_2^{(3)}) + \mathcal{O}(\tau) = \frac{\partial f_{2a}}{\partial u}(u_k) + \mathcal{O}(\tau),$$

$$T_{2b} = \frac{\partial f_{2b}}{\partial u}(\tilde{z}_2^{(2)}) + \mathcal{O}(\tau) = \frac{\partial f_{2b}}{\partial u}(u_k) + \mathcal{O}(\tau)$$

are first-order approximations to the Jacobians J_{2a} and J_{2b} , respectively. This method has a consistency order two and, applied to the test equation $u'(t) = A_1 u(t) + A_2 u(t)$ with real matrices A_1 and A_2 , we obtain the amplification matrix

$$R(\tau A_1, \tau A_2) = \left(I - \frac{\tau}{2} A_1\right)^{-1} \left(I - \frac{\tau}{2} A_2\right)^{-1} \left(I + \frac{\tau}{2} A_2\right) \left(I + \frac{\tau}{2} A_1\right),$$

which is A -acceptable, if the matrices A_1 and A_2 commute, and if they have a nonpositive logarithmic matrix norm, see [19].

Automatic Time-Step Size Control An automatic time-step size control is advisable for the practical application of the method in order to avoid time-steps that are too large (which could lead to instability of the method or inaccurate solutions) on one hand and time-steps that are too small (which lead to unnecessary costs in the computation of the numerical solution) on the other. The aim is to select a sequence of time-step sizes such that the local error from one step of the method to the next is bounded by some given tolerance tol . We use an approach via Richardson extrapolation for the time-step size selection, see, e.g., [20, p. 55]. (Embedding worked equally well for a similar splitting method applied to semi-discretizations of a spatially one-dimensional version of (1.1a-1.1b), but we expect Richardson extrapolation to give more reliable results in general.)

Let p be the order of the ODE solver ($p = 2$ in our case), m be the dimension of the ODE system (2.15), τ be the current time-step size, and let $rtol_i$ and $atol_i$ be given relative and absolute tolerances for each component of the solution. We compute a time-step τ with the splitting scheme (2.16) with initial values u_k and obtain u_τ as an approximation of $u(t_k + \tau)$. We next compute the solution of the splitting scheme (2.16) and initial condition u_k with two time-steps $\tau/2$, which gives a second approximation $u_{2 \times \frac{\tau}{2}}$ of $u(t_k + \tau)$. A scaled error measure is then defined by

$$\rho = \frac{1}{2^p - 1} \sqrt{\frac{1}{m} \sum_{i=1}^m \left(\frac{(u_{2 \times \frac{\tau}{2}})_i - (u_\tau)_i}{atol_i + rtol_i |(u_k)_i|} \right)^2}. \quad (2.20)$$

The step is rejected if $\rho > 1$ and accepted otherwise. The new time-step size τ_{new} (for the following step or for the recomputation of the rejected step) is computed by

$$\tau_{new} = \tau \cdot \min\{2, \max\{0.8\rho^{-\frac{1}{2p-1}}, 0.25\}\}. \quad (2.21)$$

If the step is accepted, then we set $u_{k+1} = u_{2 \times \frac{\tau}{2}}$, the more accurate p th order approximation. Extrapolation to order $p + 1$ would be possible, but one would have to reassess the stability (and positivity) properties of the resulting method. Furthermore, numerical experiments indicated a poorer performance of the method if used with extrapolation.

The Jacobians J_{2a} and J_{2b} are computed with respect to u_k (as indicated in the section on the implicit method) at the beginning of a splitting step. They can, therefore, be reused in rejected steps, which reduces the computational cost.

III. NUMERICAL EXPERIMENTS

We solve the equations of Example I. on a 150×150 spatial grid. The PDEs are discretized in space as described in the previous section (using van Leer's limiter (2.13)) and give rise to an

ODE system with 45149 equations. This system is solved up to final times $t = 0.5$ and $t = 0.9$. For the convection in the x -direction, we assume *a priori* a negative velocity. Numerical tests on the actual numerical velocities proved this assumption to be valid. Our reference solution is computed with the explicit RK code DOPRI5 [21] applied with $tol = 10^{-12}$ to the ODE system, and we, therefore, study the temporal error (including the time-splitting error) only. We use the splitting algorithm based on (2.16) for the solution of the ODE system and compare the results with those obtained by the Krylov–ROW code ROWMAP [22] and the Krylov–BDF method VODPK [23] applied to the nonsplit system. Test results obtained with the explicit code DOPRI5 are included for comparison.

Figure 3 shows the solution n at times $t = 0.5$ (left) and $t = 0.9$ (right) obtained with our splitting method. The plots of c are not included, because the development of n in time and space is more important than that of c from a biological point of view. The results obtained are in qualitative agreement with experimental observations and with the computational results in [24, 25].

Numerical experiments with fixed time-step sizes confirm the second-order accuracy of the splitting method, but the details are not reported here.

Table I summarizes the results obtained for the three codes under consideration for a final time 0.9. The splitting algorithm performs best from the point of view of positivity. The solution is positive (to within rounding errors) even for low tolerance requirements (10^{-4}). This cannot be matched by ROWMAP or VODPK though the accuracy of these codes is better for more stringent tolerances. Figure 4 shows that the splitting scheme performs better than the ROWMAP and VODPK codes for a final time $t = 0.9$ and an error (measured in the scaled l_2 -norm $\|u\|_2 = (m^{-1}u^T u)^{1/2}$ for all $u \in \mathbb{R}^m$) of up to about 10^{-4} (results for final time 0.5 are also included in this figure). The time-step in the explicit code DOPRI5 is, of course, restricted by stability, and it is found that its positivity results are not very good for large tolerances.

If the less smooth maximum limiter [7] is used instead of van Leer’s limiter, then not only is the positivity of the solution violated more significantly (for both the splitting method and the Krylov codes) but the codes require more steps (as well as having more rejected steps).

Finally, we look at the numerical wave speed of the approximation to n in the discretization with van Leer’s limiter and the splitting algorithm on different grids. Figure 5 suggests that the wave speed of the solution does not depend significantly on the spatial resolution. In both cases

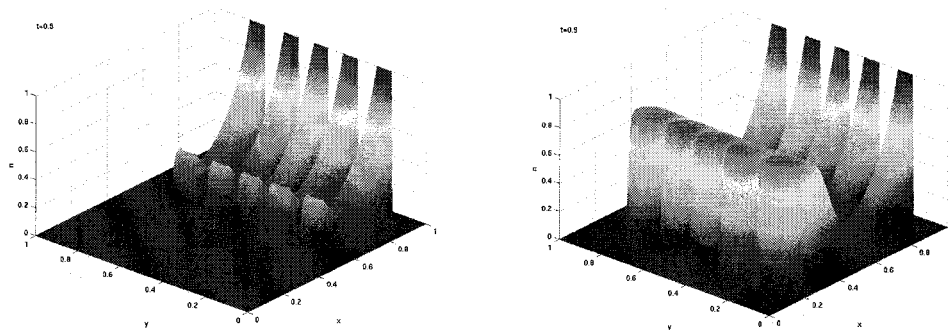


FIG. 3. MOL approximation of n of Example I. at times $t = 0.5$ (left) and 0.9 (right).

TABLE I. Results for Example I. solved up to $t = 0.9$ with the splitting algorithm, ROWMAP, and VODPK for required tolerances $tol = 10^{-2}, \dots, 10^{-8}$ (if results are not given for a certain value of tol then the code did not return a solution because of too small time steps).

Method	$\log(tol)$	steps	rej. steps	$\log(\ error\ _2)$	$\min_i \{U_i\}$	cpu time (s)
SPLITTING	-2	29	3	-1.54	-0.903E-03	83
	-3	42	0	-1.79	-0.190E-04	116
	-4	86	0	-2.11	-0.982E-20	234
	-5	181	1	-2.62	-0.641E-20	487
	-6	388	0	-3.31	-0.488E-20	1049
	-7	936	35	-4.08	-0.430E-20	2498
	-8	2843	470	-4.87	-0.260E-20	7339
ROWMAP	-5	260	26	-2.88	-0.118E-02	1534
	-6	933	196	-3.82	-0.208E-03	2868
	-7	2063	549	-4.58	-0.343E-09	4184
	-8	5484	1833	-5.52	-0.142E-14	7255
VODPK	-3	737	0	-0.98	-0.230E-02	612
	-4	1003	0	-1.68	-0.434E-04	801
	-5	1156	0	-2.89	-0.413E-05	979
	-6	1369	5	-3.88	-0.919E-10	1237
	-7	2227	2	-5.12	-0.137E-18	1792
	-8	4248	137	-6.20	-0.518E-20	3011

(100 and 200 grid points in each spatial direction), the wave peak is at the same location at $t = 0.5$, although there is a small discrepancy in the location of the leading edge of the wave.

With these observations in mind we draw the following conclusions:

- Discretization with van Leer’s limiter appears to be best with regard to both positivity and efficiency. The maximum limiter might be more accurate, but the resulting ODE system is not sufficiently positive and this leads to difficulties in the numerical solution of the ODE system (oscillations, rejected steps).
- Methods directly employing the Jacobian are not suitable for the solution of ODE systems whose right-hand sides are generated from nonsmooth limiters when only low accuracy is required. Better results are obtained for smaller tolerances, but this also leads to long computation times (for all codes). The effects of the nonsmooth limiter are more pronounced

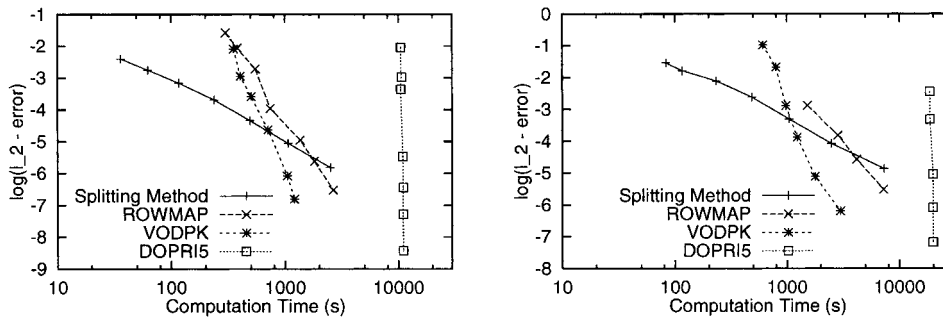


FIG. 4. Accuracy Achieved vs Computation Time plots for Example I. at $t = 0.5$ (left) and $t = 0.9$ (right).

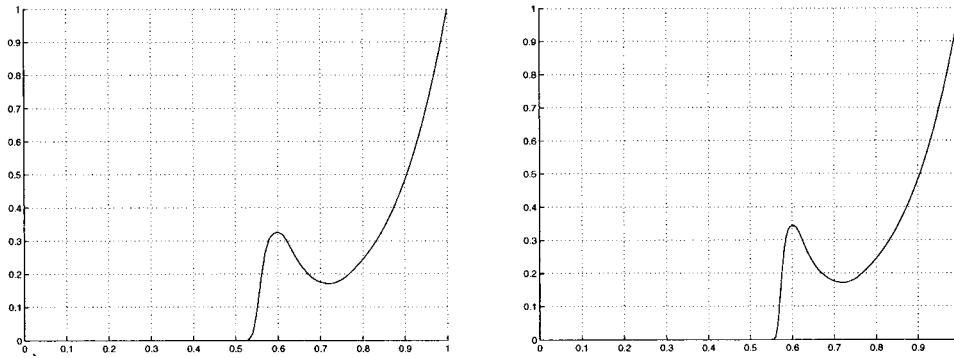


FIG. 5. Solution n of Example I. along $x \in [0, 1]$, $y = 0.5$ at $t = 0.5$ with 100 grid points in each spatial direction (left) and 200 grid points (right); splitting algorithm with $tol = 10^{-6}$.

when employing the ROWMAP code compared to VODPK. The reason is that ROWMAP requires exact Jacobian vector products to attain the order of consistency four [26], whereas VODPK uses these products in the Newton iteration only and they do not influence the order of consistency.

- The splitting scheme developed here works better than standard codes in the accuracy range 10^{-2} to 10^{-4} . This accuracy range is usually sufficient in simulations of biological processes. The code based on our approach is considerably faster than DOPRI5, ROWMAP, and VODPK. Furthermore, the splitting method satisfies positivity requirements better than the other codes.

IV. DISCUSSION AND CONCLUSIONS

In this article, we have presented a numerical method for the solution of a system of coupled hyperbolic–parabolic partial differential equations arising from a biomathematical model in which cells migrate in response to the concentration field of a chemical. The numerical scheme used the method of lines technique with special attention being given to preserving the positivity of the solution. In order to achieve this, we employed a flux limiter in the discretization of the hyperbolic equation. The resulting system of stiff ODEs was then solved using appropriate splitting techniques—an explicit scheme for those terms arising from the convection terms of the hyperbolic PDE and a linearly implicit splitting method for those terms arising from the reaction-diffusion terms of the PDE system.

The MOL approach appears to be a good option for the class of PDE problems under consideration here. The splitting method developed is of order two in time and possesses good stability and positivity properties. A further advantage is that we can easily use a time-step size control, which keeps the local temporal error below a user-defined tolerance. Furthermore, the computational costs for the solution of the large ODE system are reduced significantly by the time-splitting procedure described. Numerical tests have demonstrated the efficiency and reliability for the problem considered.

ACKNOWLEDGMENT

The authors thank Zóltan Hórvath for helpful discussions on the subject of positivity.

References

1. A. R. A. Anderson and M. A. J. Chaplain, Continuous and discrete mathematical models of tumor-induced angiogenesis, *Bull Math Biol* 60 (1998), 857–899.
2. A. R. A. Anderson, M. A. J. Chaplain, E. L. Newman, R. J. C. Steele, and A. M. Thompson, Mathematical modeling of tumour invasion and metastasis, *J Theor Med*, to appear.
3. T. Höfer, J. A. Sherratt, and P. K. Maini, Cellular pattern formation during *Dictyostelium* aggregation, *Physica D* 85 (1995), 425–444.
4. M. A. J. Chaplain and A. M. Stuart, A model mechanism for the chemotactic response of endothelial cells to tumour angiogenesis factor, *IMA J Math Appl Med Biol* 10 (1993), 149–168.
5. A. Gerisch, Finite difference methods for coupled nonlinear hyperbolic and parabolic partial differential equations in one and two dimensions, Master's thesis, U Dundee, 1997.
6. A. Gerisch, D. F. Griffiths, R. Weiner, and M. A. J. Chaplain, Splitting methods for mixed hyperbolic-parabolic systems, Tech Rept NA/186, U Dundee, 1998.
7. W. Hundsdorfer, B. Koren, M. van Loon, and J. G. Verwer, A positive finite-difference advection scheme, *J Comp Phys* 117 (1995), 35–46.
8. Zoltán Horváth, Positivity of Runge–Kutta and diagonally split Runge–Kutta methods, *Appl Numer Math* 28 (1998), 309–326.
9. W. H. Hundsdorfer, Numerical solution of advection-diffusion-reaction equations, Note NM-N9603, Dept Numer Math, CWI, 1996.
10. D. F. Griffiths, A. M. Stuart, and H. C. Yee, Numerical wave propagation in an advection equation with nonlinear source term, *SIAM J Numer Anal* 29 (1992), 1244–1260.
11. P. K. Sweby, High resolution schemes using flux limiters for hyperbolic conservation laws, *SIAM J Numer Anal* 21 (1984), 995–1011.
12. B. Koren, A robust upwind discretization method for advection, diffusion and source terms, Numerical methods for advection–diffusion problems, V 45 of Notes on numerical fluid mechanics, C. B. Vreugdenhill and B. Koren (Editors), Vieweg, Braunschweig, 1993, pp. 117–138.
13. U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, Implicit-explicit methods for time-dependent partial differential equations, *SIAM J Numer Anal* 32 (1995), 797–823.
14. U. M. Asher, S. J. Ruuth, and R. J. Spiteri, Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, *Appl Numer Math* 25 (1997), 151–167.
15. J. Frank, W. Hundsdorfer, and J. G. Verwer, On the stability of implicit-explicit linear multistep methods, *Appl Numer Math* 25 (1997), 193–205.
16. J. F. B. M. Kraaijevanger, Absolute monotonicity of polynomials occurring in the numerical solution of initial value problems, *Numer Math* 48 (1986), 303–322.
17. C. Bolley and M. Crouzeix, Conservation de la positivité lors de la discrétisation des problèmes d'évolution paraboliques, *RAIRO Anal Numer* 12 (1978), 237–245.
18. W. Hundsdorfer, Trapezoidal and midpoint splittings for initial-boundary value problems, *Math Comp* 67 (1998), 1047–1062.
19. C. Eichler-Liebenow, N. H. Cong, R. Weiner, and K. Strehmel, Linearly implicit splitting methods for higher space-dimensional parabolic differential equations, *Appl Numer Math* 28 (1998), 259–274.
20. K. Strehmel and R. Weiner, *Numerik gewöhnlicher Differentialgleichungen*, Teubner, Stuttgart, 1995.

21. E. Hairer, S. P. Nørsett, and G. Wanner, Solving ordinary differential equations I, Springer, New York, 1987.
22. R. Weiner, B. A. Schmitt, and H. Podhaisky, ROWMAP—a ROW-code with Krylov techniques for large stiff ODEs, *Appl Numer Math* 25 (1997), 303–319.
23. G. D. Byrne, Pragmatic experiments with Krylov methods in the stiff ODE setting, *Computational ordinary differential equations*, J. Cash and I. Gladwell (Editors), Oxford Univ. Press, Oxford, 1992, pp. 323–356.
24. M. A. J. Chaplain, The mathematical modeling of tumour angiogenesis and invasion, *Acta Biotheor* 43 (1995), 387–402.
25. M. A. J. Chaplain, Avascular growth, angiogenesis and vascular growth in solid tumours: The mathematical modeling of the stages of tumour development, *Math Comp Model* 23 (1996), 47–87.
26. R. Weiner and B. A. Schmitt, Order results for Krylov-W-methods, *Comp* 61 (1998), 69–89.