Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

MAS

Modelling, Analysis and Simulation

Modelling, Analysis and Simulation

Simulation of 3D Phytoplankton Dynamics: Competition in Light-Limited Environments

N.N.Pham Thi, J.Huisman, B.P.Sommeijer

REPORT MAS-R0314 NOVEMBER 25, 2003

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

Copyright © 2003, Stichting Centrum voor Wiskunde en Informatica
P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3703

Simulation of 3D Phytoplankton Dynamics: Competition in Light-Limited Environments.

ABSTRACT
In this paper, we develop computational methods for a 3D model of competition for light between phytoplankton species. The competing phytoplankton populations are exposed to both horizontal and vertical mixing. The vertical light-dependence of phytoplankton photosynthesis implies that the 3D-model is formulated in terms of integro-partial differential equations that require an efficient numerical solution technique. Due to the stiffness of the discretized system we select an implicit integration method. However, the resulting implicit relations are extremely expensive to solve, caused by the strong coupling of the components. This coupling originates from the three spatial dimensions, the interaction of the various species and the integral term. To reduce the amount of work in the linear algebra part, we use an Approximate Matrix Factorization technique. The performance of the complete algorithm is demonstrated on the basis of two test examples. It turns out that the property of A-stability is very useful for this application.

2000 Mathematics Subject Classification: Primary: 65M12, 65M20.
1998 ACM Computing Classification System: G.1.7, G.1.8.
Keywords and Phrases: Advection-diffusion-reaction equations, time integration, stiffness, approximate matrix factorization, phytoplankton dynamics, competition.
Note: Work carried out under subtheme MAS1.1-Applications from the Life Sciences.

Simulation of 3D Phytoplankton Dynamics: Competition in Light-Limited Environments *

N.N. Pham Thi [a,†], J. Huisman [b], and B.P. Sommeijer [a]

[a] *Center for Mathematics and Computer Science (CWI), P.O. Box 94079, 1090 GB Amsterdam, The Netherlands*

[b] *Aquatic Microbiology, Institute for Biodiversity and Ecosystem Dynamics, University of Amsterdam, Nieuwe Achtergracht 127, 1018 WS Amsterdam, The Netherlands*

Abstract

In this paper, we develop computational methods for a 3D model of competition for light between phytoplankton species. The competing phytoplankton populations are exposed to both horizontal and vertical mixing. The vertical light-dependence of phytoplankton photosynthesis implies that the 3D-model is formulated in terms of integro-partial differential equations that require an efficient numerical solution technique.

Due to the stiffness of the discretized system we select an implicit integration method. However, the resulting implicit relations are extremely expensive to solve, caused by the strong coupling of the components. This coupling originates from the three spatial dimensions, the interaction of the various species and the integral term. To reduce the amount of work in the linear algebra part, we use an Approximate Matrix Factorization technique.

The performance of the complete algorithm is demonstrated on the basis of two test examples. It turns out that the property of A-stability is very useful for this application.

2000 Mathematics Subject Classification: Primary: 65M12, 65M20.
1998 ACM Computing Classification System: G.1.7, G.1.8.
Keywords and Phrases: Advection-diffusion-reaction equations, time integration, stiffness, approximate matrix factorization, phytoplankton dynamics, competition.
Note: Work carried out under subtheme MAS1.1 - Applications from the Life Sciences.

1 Introduction

Lakes, seas, and oceans are inhabited by large numbers of free-floating microorganisms called phytoplankton. Like grass, trees and other plants, phytoplankton utilize

*The investigations were supported by the Computational Science Program, which is subsidized by the Netherlands Organization for Scientific Research (NWO).

†Corresponding author. Email address: N.N.Pham.Thi@cwi.nl.

solar energy and carbon dioxide to produce biomass, in a process known as photosynthesis ([15]). Phytoplankton photosynthesis forms the basis for nearly all aquatic foodwebs, and thereby has a major impact on the productivity of aquatic ecosystems including fish production. Furthermore, because phytoplankton absorb carbon dioxide during photosynthesis, on a global scale phytoplankton remove nearly as much of the greenhouse gas CO_2 from the atmosphere as all land plants do. As a result, phytoplankton photosynthesis has a major influence on climate change ([5, 6]). For these reasons, studies of the growth and population dynamics of phytoplankton is of great interest.

Phytoplankton is the generic name for many different species from a wide variety of taxonomic groups, including cyanobacteria, prochlorophytes, diatoms, coccolithophores, and dinoflagellates ([8]). The species composition of the phytoplankton plays an important role. Some species are sinking species, i.e., they have a higher specific weight than water. As a result, they transport their biologically fixed carbon into the deep ocean ([1, 10]). Other species are buoyant species since their specific weight is smaller than that of water. These species remain near the surface. All these species essentially require the same environmental resources (light, mineral nutrients, and carbon dioxide) and each phytoplankton species faces competition from the other phytoplankton species when one or more of these essential resources are available in low quantities only. This paper concentrates on competition for light between phytoplankton species. Competition for light is a major determinant of the species composition of phytoplankton communities ([11, 12, 13]), as light is the energy source that drives phytoplankton photosynthesis.

In the context of competition for light, physical mixing processes that affect the spatial distributions of the phytoplankton species play a prominent role. In particular, phytoplankton species that manage to stay in the upper water layer have plenty of light available for photosynthesis and have the additional advantage to shade other species at deeper levels. During recent years, several 3D models that combine physical mixing processes and phytoplankton growth have been developed ([7, 16, 17, 18]). These biological-physical models have advanced the general understanding of the productivity of marine ecosystems, and play an increasingly important role in oceanographic research. However, in many of the numerical applications of these models the special structure that stems from the vertical light-dependence of phytoplankton photosynthesis has not been fully recognized. Owing to shading, the decrease of light intensity with depth appears in the model as an integral over the dynamic phytoplankton concentrations. The resulting phytoplankton model is therefore framed in terms of integro-partial differential equations. Competition for light results in coupling of the population dynamics of the phytoplankton species through shading, that is, the integro-PDEs are coupled through this integral term. For this complicated model structure, efficient numerical solution techniques that avoid numerical artifacts are indispensable.

We recently outlined an efficient simulation technique for the 1D-vertical model formulation of phytoplankton competition in light-limited environments [13]. In this paper, we extend our approach by incorporation of horizontal water flow. The aim of the paper is to come up with an efficient numerical technique for the simulation of three-dimensional phytoplankton models that include competition for light between

phytoplankton species.

The paper is organized as follows. In Section 2 we formulate our model system which is based on [12] and [13]. Section 3 deals with the numerical technique to solve this system. Two applications are described in Section 4. Section 5 is devoted to final remarks and some discussion.

2 The model system

We consider a model of competition for light between n species, where we assume that the species interact with one another indirectly, via shading.

Let $\omega_s(x, y, z, t)$, $s = 1, \ldots, n$, denote phytoplankton population densities (cells per unit volume) of n species (the subscripts s indicate the different species) at position (x, y, z) in a three-dimensional domain Ω at time t ($t \geq 0$). Here z is expressed as the depth of the water column from the surface ($z = 0$) to the bottom ($z = Z$), x varies between 0 and X, y runs from 0 to Y.

Phytoplankton use energy in sunlight for photosynthesis. In the water column, light intensity L decreases with depth according to

$$L(x, y, z, t) \;=\; L_{in}\, e^{-K_{bg} z} e^{-\int_0^z \left(\sum_{s=1}^{n} r_s \omega_s(x,y,\sigma,t)\right) d\sigma}. \tag{1}$$

At a particular depth, light intensity depends on the incident light intensity L_{in}, the background turbidity K_{bg} due to all non-phytoplankton components in the water and on the total light attenuation of all phytoplankton species above that depth. Here r_s denotes the specific light attenuation coefficient of the s-th species.

The formulation (1) explicitly involves light absorption by all phytoplankton species. Thus, the light gradient changes with a change in any species density distribution.

The change in concentration (density distribution) for each species is determined by growth and the local transport process through the partial differential equation (PDE)

$$\frac{\partial \omega_s}{\partial t} \;=\; g_s\left(L\right)\, \omega_s \;-\left(\frac{\partial \mathcal{I}_s}{\partial x} \;+\; \frac{\partial \mathcal{J}_s}{\partial y} \;+\; \frac{\partial \mathcal{K}_s}{\partial z} \right). \tag{2}$$

Here $g_s\left(L(x, y, z, t)\right)$ is the specific growth rate of the s-th species driven by light availability. $\mathcal{I}_s(x, y, z, t)$, $\mathcal{J}_s(x, y, z, t)$ and $\mathcal{K}_s(x, y, z, t)$ are defined below, and are respectively the horizontal and vertical fluxes of the s-th species at position (x, y, z) and time t.

The specific growth rate $g_s\left(L(x, y, z, t)\right)$ in the above equation depends on the balance between the production rate $p_s\left(L(x, y, z, t)\right)$ and the specific loss rate ℓ_s

$$g_s(L) \;=\; p_s(L) \;-\; \ell_s. \tag{3}$$

Here $p_s(L)$ is represented by the product of $p_{s_{max}}$, the maximum specific production rate, and the ratio of the realized growth rate to the maximum one ([11])

$$p_s(L) \;=\; \frac{p_{s_{max}} L}{H_s + L}, \tag{4}$$

where H_s denotes the half-saturation constant.

The fluxes in equation (2) depend on the dynamics of the system as determined by the horizontal flow, the vertical velocity, and the transport of phytoplankton by turbulent diffusion

$$\begin{aligned}
\mathcal{I}_s(x,y,z,t) &= a(x,y,z)\,\omega_s(x,y,z,t) - D_H(z)\,\frac{\partial \omega_s}{\partial x}(x,y,z,t),\\
\mathcal{J}_s(x,y,z,t) &= b(x,y,z)\,\omega_s(x,y,z,t) - D_H(z)\,\frac{\partial \omega_s}{\partial y}(x,y,z,t),\\
\mathcal{K}_s(x,y,z,t) &= c_s\,\omega_s(x,y,z,t) - D_V(z)\,\frac{\partial \omega_s}{\partial z}(x,y,z,t),
\end{aligned} \tag{5}$$

where $a(x,y,z)$, $b(x,y,z)$ are the horizontal velocity components of the water flow, c_s is the vertical velocity of the s-th species (which is positive for sinking phytoplankton and negative for buoyant phytoplankton), and $D_H(z)$ and $D_V(z)$ are the horizontal and the vertical turbulent diffusion coefficients at depth z. The minus sign in the second terms on the right hand side indicates that turbulent diffusion is in the direction opposite to the concentration gradient. In the above formulation the spatial-dependent velocities $a(x,y,z)$, $b(x,y,z)$ and the depth-dependent diffusion coefficients $D_H(z)$ and $D_V(z)$ are assumed to be constant in time. The characteristic velocity c_s is taken constant. However, an extension to more general functions is straightforward.

Our key system, *the system of integro-partial differential equations*, follows now straightforwardly from substituting (1), (3), (4) and (5) into (2)

$$\begin{aligned}
\frac{\partial \omega_s}{\partial t} &= p_s\left(L_{in}e^{-K_{bg}z}e^{-\int_0^z\left(\sum_{j=1}^n r_j\omega_j(x,y,\sigma,t)\right)d\sigma}\right)\omega_s - \ell_s\,\omega_s -\\
&\quad\left[(a\omega_s)_x + (b\omega_s)_y + c_s(\omega_s)_z - (D_H\omega_{s_x})_x - (D_H\omega_{s_y})_y - (D_V\omega_{s_z})_z\right],
\end{aligned} \tag{6}$$

where $s = 1,\ldots,n$. Here, the subscripts x, y and z denote the spatial differentiation in the various directions. From this formula, one can see that a change in any of the phytoplankton densities $\omega_j(x,y,\sigma,t)$ $(\sigma < z)$, within the integral term, causes a change in the light intensity, which in turn, impacts on the population density $\omega_s(x,y,z,t)$ of all species. In other words, the species struggle with one another for light by a changing light gradient over depth.

The boundaries of the system are assumed to be closed. That is, there is no phytoplankton entering or leaving the domain. Therefore, the fluxes $\mathcal{I}_s(x,y,z,t)$, $\mathcal{J}_s(x,y,z,t)$ and $\mathcal{K}_s(x,y,z,t)$ all vanish at the boundaries of the domain Ω, defining the boundary conditions for our PDE system (6).

3 Numerical approach

In order to find the numerical solution of the system (6), we use a technique which is based on the popular Method of Lines (MOL) approach, where space and time discretizations are considered separately ([14]). That is, first we derive a large system of ordinary differential equations (ODEs), which is still continuous in time, from the

discrete approximations of the spatial differential operators as well as the integral term (Section 3.1). Then, that ODE system will be integrated in time numerically (Section 3.2).

Using this approach is motivated by the fact that it is easy to combine various discretizations for advection and diffusion with the treatment of the reaction term. Another attractive, practical point is that there exist nowadays many well developed ODE methods and for these methods sophisticated software is freely available.

3.1 Spatial discretization

There are many ways to discretize the differential operators on the domain Ω. The purpose is to approximate the solution at a desired accuracy level, with as few grid points as possible. Most simple to use is the equidistant grid

$$\begin{aligned} x_0 = 0, \quad x_i = (i - \frac{1}{2})\Delta x, \quad i = 1, \ldots, N_1, \quad x_{N_1+1} = X, \\ y_0 = 0, \quad y_j = (j - \frac{1}{2})\Delta y, \quad j = 1, \ldots, N_2, \quad y_{N_2+1} = Y, \\ z_0 = 0, \quad z_k = (k - \frac{1}{2})\Delta z, \quad k = 1, \ldots, N_3, \quad z_{N_3+1} = Z, \end{aligned} \tag{7}$$

where $\Delta x = X/N_1, \Delta y = Y/N_2, \Delta z = Z/N_3$. Each grid point is imaginarily surrounded by a cell, at the boundaries of which we approximate the derivative of the fluxes (the terms inside the bracket in (2)). In the internal intervals, the cell faces lie halfway the grid points. For the end intervals, the grid points are positioned on the boundary of Ω and coincide with the cell faces. This way of discretizing is based on the so-called finite-volume method ([14, 19]). In this way, we obtain *conservation* of the flux quantity since all contributions of the fluxes along the interior cell faces cancel ([19]).

To be more precise, $\partial \mathcal{I}_s / \partial x$ in the internal points (x_i, y_j, z_k) is approximated by $(\mathcal{I}_{s_{ijk}} - \mathcal{I}_{s_{(i-1)jk}})/\Delta x$, where $\mathcal{I}_{s_{ijk}}$ denotes the flux $\mathcal{I}_s$ at $(x_{i+1/2}, y_j, z_k)$ with $x_{i+1/2} := x_i + \frac{1}{2}\Delta x$, i.e.,

$$\mathcal{I}_{s_{ijk}} = a(x_{i+\frac{1}{2}}, y_j, z_k)\, \omega_s(x_{i+\frac{1}{2}}, y_j, z_k, t) - D_H \frac{\partial \omega_s}{\partial x}(x_{i+\frac{1}{2}}, y_j, z_k, t). \tag{8}$$

Here, for simplification, we consider the model with uniform turbulent diffusion coefficients. The approximation to $\mathcal{I}_{s_{ijk}}$ is obtained by using the approach that is nowadays standard in the field of Computational Fluid Dynamics for the numerical solution of advection-diffusion equations ([14, 19]). That is, the diffusion term is discretized symmetrically

$$D_H \frac{\partial \omega_s}{\partial x}(x_{i+\frac{1}{2}}, y_j, z_k, t) \approx D_H \frac{w_{s_{(i+1)jk}}(t) - w_{s_{ijk}}(t)}{\Delta x}, \tag{9}$$

where $w_{s_{ijk}}(t)$ denotes an approximation to the population density of the s-th species at (x_i, y_j, z_k) and time t. For the advection term, the third-order upwind-biased discretization is used ([14])

$$a(x_{i+\frac{1}{2}}, y_j, z_k)\, \omega_s(x_{i+\frac{1}{2}}, y_j, z_k, t) \approx a(x_{i+\frac{1}{2}}, y_j, z_k)\, w_{s_{(i+\frac{1}{2})jk}}(t), \tag{10}$$

where [1]

$$w_{s_{(i+\frac{1}{2})jk}} = \begin{cases} \frac{1}{6}\left(-w_{s_{(i-1)jk}} + 5w_{s_{ijk}} + 2w_{s_{(i+1)jk}}\right) & \text{if } a(x_{i+\frac{1}{2}}, y_j, z_k) > 0, \\ \frac{1}{6}\left(2w_{s_{ijk}} + 5w_{s_{(i+1)jk}} - w_{s_{(i+2)jk}}\right) & \text{if } a(x_{i+\frac{1}{2}}, y_j, z_k) < 0. \end{cases} \tag{11}$$

This upwind discretization is preferred to the more simple second-order symmetric discretization as the symmetric discretization of the advection term more easily leads to "wiggles" in the numerical solution, which may result in negative solution components. A negative population density is of course not realistic. To reduce this unwanted property one can use the current upwind scheme. Higher order upwind schemes may give still better results. However, for these we need a larger stencil of grid points which makes such methods impractical in simulation with boundary conditions (see e.g. [14] for more details).

We note that, according to the boundary condition, the fluxes $\mathcal{I}_{s_{0jk}}, \mathcal{I}_{s_{N_1 jk}}$ vanish. Since we lack sufficient upstream information, a symmetric discretization for $\mathcal{I}_{s_{1jk}}$ (or $\mathcal{I}_{s_{(N_1-1)jk}}$) has been used in case of $a(x_{1+\frac{1}{2}}, y_j, z_k) > 0$ (or $a(x_{N_1-\frac{1}{2}}, y_j, z_k < 0$)

$$\begin{aligned} w_{s_{(1+\frac{1}{2})jk}} &= \frac{w_{s_{2jk}} + w_{s_{1jk}}}{2} && \text{if } a(x_{1+\frac{1}{2}}, y_j, z_k) > 0, \\ w_{s_{(N_1-\frac{1}{2})jk}} &= \frac{w_{s_{N_1 jk}} + w_{s_{(N_1-1)jk}}}{2} && \text{if } a(x_{N_1-\frac{1}{2}}, y_j, z_k) < 0. \end{aligned} \tag{12}$$

A complete approximation to $\mathcal{I}_{s_{ijk}}$ is then obtained by the combination of (9), (10), (11) and (12). In the same way we obtain approximations for the fluxes $\mathcal{J}_{s_{ijk}}$ and $\mathcal{K}_{s_{ijk}}$.

Using the repeated trapezoidal rule for the integral term within the light function (L in (1)) the light intensity at (x_i, y_j, z_k) is approximated by

$$L_{ijk} = L_{in} e^{-K_{bg} z_k} e^{-\sum_{s=1}^{n} r_s \left[\frac{1}{4}w_{s_{ij0}} + \frac{3}{4}w_{s_{ij1}} + w_{s_{ij2}} + \cdots + w_{s_{ij(k-1)}} + \frac{1}{2}w_{s_{ijk}}\right] \Delta z} \tag{13}$$

with the solution at the surface, $w_{s_{ij0}}$, extrapolated as $w_{s_{ij0}} = (3w_{s_{ij1}} - w_{s_{ij2}})/2$. The corresponding specific growth rate is then $g_{s_{ijk}} := g_s(L_{ijk}) = p_s(L_{ijk}) - \ell_s$.

Finally, we arrive at the following set of ODEs

$$\frac{dw_{s_{ijk}}(t)}{dt} = g_{s_{ijk}} w_{s_{ijk}} - \frac{\mathcal{I}_{s_{ijk}} - \mathcal{I}_{s_{(i-1)jk}}}{\Delta x} - \frac{\mathcal{J}_{s_{ijk}} - \mathcal{J}_{s_{i(j-1)k}}}{\Delta y} - \frac{\mathcal{K}_{s_{ijk}} - \mathcal{K}_{s_{ij(k-1)}}}{\Delta z}, \tag{14}$$

where s, i, j and k respectively run from 1 to n, N_1, N_2 and N_3.

3.2 Time integration

This section deals with the numerical integration of the above derived ODE system, which is still continuous in time and can be written in the form

$$\frac{d\,\mathbf{w}(t)}{d\,t} = \mathbf{F}\left(\mathbf{w}(t)\right), \quad t \geq 0, \tag{15}$$

[1] Here we omit the explicit time-dependence in the notation.

where the vector $\mathbf{w}(t) \in \mathbb{R}^N$, $N = n\,N_1\,N_2\,N_3$, contains the components $w_{s_{ijk}}$. This system is *stiff* (has widely spread eigenvalues) due to the fact that the spectral radius of the diffusion part is proportional to the square of the number of grid points ([14]). To cope with the stiffness of this ODE system, we use an *implicit* BDF integration method. Consequently, in each time step, we need to solve a system of implicit relations to find the solution at the next point in time, $\mathbf{W}_{k+1}$, using previously computed values $\mathbf{W}_k, \mathbf{W}_{k-1}, \ldots, \mathbf{W}_{k+1-m}$. We denote this system by

$$\mathbf{R}(\mathbf{W}_{k+1}) := \mathbf{W}_{k+1} - \gamma \Delta t \mathbf{F}(\mathbf{W}_{k+1}) - \sum_{i=1}^{m} \alpha_i \mathbf{W}_{k+1-i} = 0, \tag{16}$$

where $\mathbf{W}_{k+1}$ is an approximation to $\mathbf{w}(t)$ at $t = t_{k+1}$, with Δt the current step size and the coefficients α_i and γ are defined by the method in use.

System (16) is iteratively solved by the modified Newton method, that is

$$[I - \gamma \Delta t\, \partial \mathbf{F}/\partial \mathbf{w}] \left[\mathbf{W}_{k+1}^{j} - \mathbf{W}_{k+1}^{j-1}\right] = -\mathbf{R}(\mathbf{W}_{k+1}^{j-1}), \quad j = 1, 2, \ldots, \tag{17}$$

where I denotes the identity matrix and the Jacobian matrix $\partial \mathbf{F}/\partial \mathbf{w}$ is occasionally evaluated at certain values $\mathbf{W}_k$. The superscript j denotes the iteration index.

The Jacobian matrix has a huge number of entries (Figure 1) since it has the structure of a matrix which is a tensor product of a $N_1 \times N_1$ 5-diagonal band matrix and a $N_2 \times N_2$ 5-diagonal band matrix (both originating from the variable advection and the diffusion parts), a $N_3 \times N_3$ 4-diagonal matrix (due to the constant vertical velocity of each species) plus a lower triangular matrix (due to the integral term), and a full $n \times n$ matrix (due to multi-species competition). This pattern in the Jacobian matrix results in a very time-consuming process in solving the linear systems in (17).

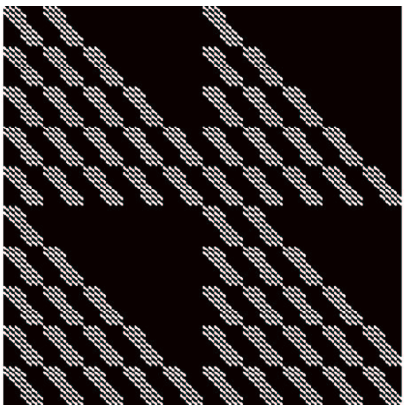


Figure 1: Structure of the Jacobian matrix in case of 2 species.

For that reason, we approximate the first term in the left-hand side of (17) as follows

$$[I - \gamma \Delta t\, \partial \mathbf{F}/\partial \mathbf{w}] \approx [I - \gamma \Delta t\, \partial \mathbf{F}_1/\partial \mathbf{w}]\,[I - \gamma \Delta t\, \partial \mathbf{F}_2/\partial \mathbf{w}]\,[I - \gamma \Delta t\, \partial \mathbf{F}_3/\partial \mathbf{w}], \tag{18}$$

where $\mathbf{F}_1(\mathbf{w}(t))$, $\mathbf{F}_2(\mathbf{w}(t))$ and $\mathbf{F}_3(\mathbf{w}(t))$ correspond to the three last terms (derivative of fluxes) in the right hand side of (14). Thus, we have removed the derivatives of the specific growth rates $g_{s_{ijk}}$ in the Jacobian matrix and have approximately factorized what remains in $[I - \gamma\Delta t\, \partial\mathbf{F}/\partial\mathbf{w}]$ in the way as shown in (18). The latter simplification is usually termed Approximate Matrix Factorization ([2, 4, 9, 14]) and can be seen as a form of dimension splitting. The growth rates can be omitted since these terms are non-stiff. Loosely speaking, the growth rates are now handled by simple successive substitution as in Jacobi iteration. All diffusion and advection terms are kept in the Jacobian because all these terms can cause stiffness. Removing them could cause stability problems.

Using the above simplified Jacobian technique we still get the required modified Newton convergence, but, with a larger number of iterations. In compensation, the cost to solve our problem is significantly lower since we only solve three simple band structured systems instead of the original complicated one.

The time integration technique based on (16) has been implemented by Brown et al. [3] in the code VODE. The results presented in this paper were obtained by adapting VODE w.r.t. the Approximate Matrix Factorization technique defined in (18). All strategies in VODE have been left unchanged. This widely used stiff ODE solver is very robust in the sense that it includes all kind of strategies, necessary for automatic integration. VODE is freely available from http://www.netlib.org/ode/ (both in Fortran and C).

4 Application

4.1 Model structure

We consider competition for light between three typical phytoplankton species: a sinking species ($c_s > 0$), a neutrally buoyant species ($c_s = 0$), and a buoyant species ($c_s < 0$). All these three species have similar growth characteristics. However, we assume that the sinking species has a higher maximal specific production rate than the neutrally buoyant species, which in turn has a higher specific production rate than the buoyant species. We note that a high specific growth rate allows proliferation under rather low light conditions. In Table 1 we specify all parameters characterizing the three species.

Table 1: Species parameters (See also Table 2 in [13]).

Species	c_s $\left(\frac{cm}{h}\right)$	r_s $\left(\frac{cm^2}{cells}\right)$	$p_{s_{max}}$ $\left(\frac{1}{h}\right)$	H_s $\left(\frac{\mu mol\text{photons}}{cm^2 \cdot s}\right)$	ℓ_s $\left(\frac{1}{h}\right)$	ω_{s_0} $\left(\frac{cells}{cm^3}\right)$
Sinking species	+4.2	0.30e-6	0.04	10.0e-4	0.01	50
Neutral species	+0.0	0.15e-6	0.03	10.0e-4	0.01	5000
Buoyant species	-8.3	0.15e-6	0.02	20.0e-4	0.01	5000

Our experiments are performed on the domain Ω with $X = 100m$, $Y = 100m$ and $Z = 10m$ using an equidistant grid of $50 \times 50 \times 20$ cells. Hence, the total system consists of 150000 ODEs.

The simulations are carried out with constant turbulences $D_H = 100, D_V = 10$, both in cm^2/s. All system parameter values [2] are summarized in Table 2.

Table 2: System parameters (see also Table 1 in [13]).

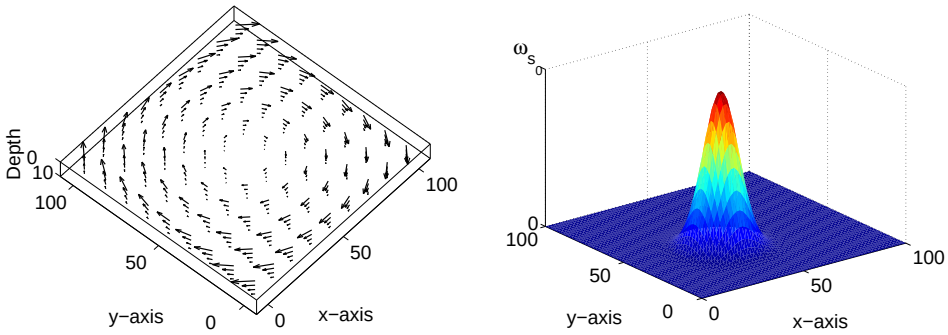
X (cm)	Y (cm)	Z (cm)	D_H $\left(\frac{cm^2}{s}\right)$	D_V $\left(\frac{cm^2}{s}\right)$	K_{bg} (cm^{-1})	L_{in} $\left(\frac{\mu mol\text{photons}}{cm^2 \cdot s}\right)$
10^4	10^4	10^3	100	10	$2.0 \cdot 10^{-3}$	$3.5 \cdot 10^{-2}$

We will use two different velocity fields for the water flow. In accordance with the incompressibility of water, both fields are divergence free.

Example 1 We start with the so-called Molenkamp velocity field (see Figure 2a)

$$a(x,y,z) = \frac{2\pi}{5Y}\left(y - \frac{Y}{2}\right) e^{-2z/Z}, \quad b(x,y,z) = -\frac{2\pi}{5X}\left(x - \frac{X}{2}\right) e^{-2z/Z}. \qquad (19)$$

This velocity field is a clockwise rotation around the center water column $(X/2, Y/2, z)$, with amplitude decreasing over depth. In the horizontal, the amplitude increases from the center towards the boundary of the domain. It takes $5Xe^{2z/Z}$ (s) for species at depth z to rotate for one cycle.



a) Molenkamp velocity field. b) Initial values of species.

Figure 2: Input for the Molenkamp test.

For this velocity field, we start with uniform population densities for each species over depth. At every depth, phytoplankton species are all densely distributed in the location of $x = 40m$ and $y = 40m$ (Figure 2b) according to

$$\omega_s(x,y,z,0) = \omega_{s_0} e^{-10^{-6}\left((x-0.4X)^2+(y-0.4Y)^2\right)},$$

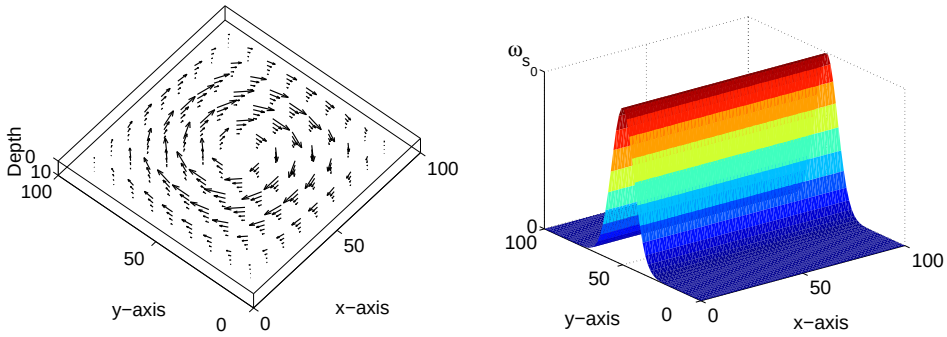
where the amplitude ω_{s_0} is given in Table 1. Notice that the neutral and the buoyant species are both initialized 100 times more abundant than the sinking species.

[2] The spatial-intervals and diffusion coefficients are chosen to be in the critical region of "bloom development" found in [13].

Example 2 In the second test example we use the velocity field defined by (see Figure 3a)

$$a(x,y,z) = -\frac{\pi}{5}\,10^{-4}X\,\sin^2(\pi\frac{x}{X})\,\sin(2\pi\frac{y}{Y})\,e^{-2z/Z},$$
$$b(x,y,z) = +\frac{\pi}{5}\,10^{-4}Y\,\sin^2(\pi\frac{y}{Y})\,\sin(2\pi\frac{x}{X})\,e^{-2z/Z}. \qquad (20)$$

This velocity field is a clockwise rotation around the center water column $(X/2, Y/2, z)$, with amplitude decreasing over depth. In the horizontal, the amplitude is minimal in the center and at the boundary of the domain.



a) The velocity field. b) Initial values of species.

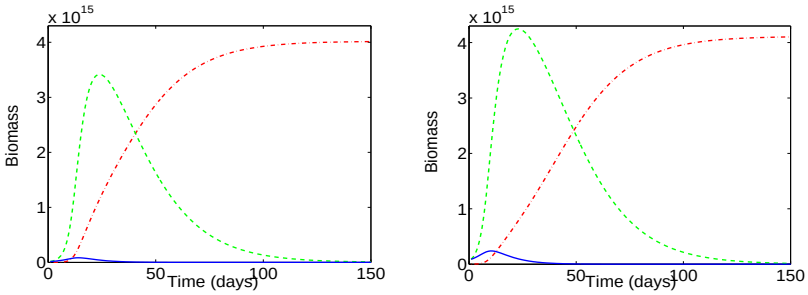
Figure 3: Input for the second test example.

For this second test example, we also start with uniform population densities for each species over depth. At every depth, phytoplankton species are, however, all densely distributed in the location of $y = 50m$ (Figure 3b) according to

$$\omega_s(x,y,z,0) = \omega_{s_0} e^{-10^{-6}(y-0.5Y)^2}.$$

4.2 Biological observations

Competition behaviour Since the depth of the water column is not large and the initial concentrations are quite low, the light availability is sufficient in the whole water column. As a result, in the first 10 days the concentrations of all three species increase (see Figure 4). The larger phytoplankton concentrations result in a quite steep light gradient. Then, the buoyant species starts to decrease because its lowest specific growth rate implies the highest light-condition requirement. Consequently, the population of the neutral species has a chance to rapidly increase until its maximal value (at about 25 days). The light availability at that time is not sufficient for such a large amount of neutral species. Thus, the neutral species no longer grows and starts to decrease towards the steady state (obtained at about 150 days). As the neutral and the buoyant species populations are getting smaller, the sinking species makes use of its high specific growth rate property and the relatively high mixing, to proliferate. Eventually, at the steady state, the sinking species dominates (see also the Figures 5, 6 and 7).

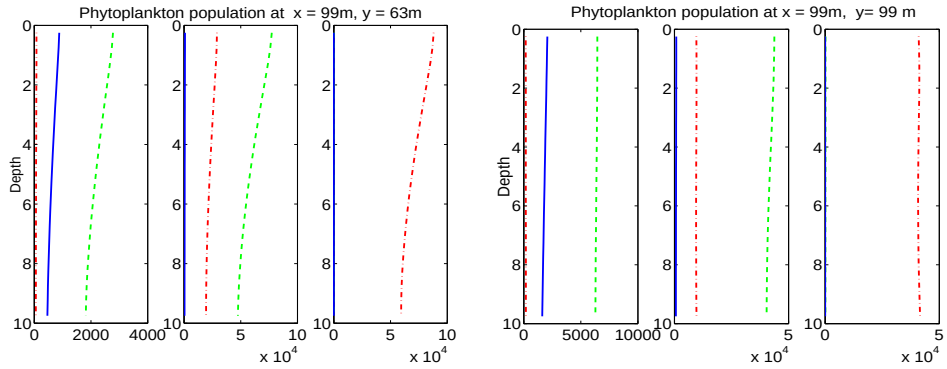


a) Example 1. b) Example 2.

Figure 4: Biomasses (integral over space) of species in time. Sinking species: dash-dot line. Neutral species: dash line. Buoyant species: solid line.

We remark that the above competition behaviour is similar to what has been observed in the one-dimensional competition model ([13]).

Vertical distribution Even though phytoplankton species are distributed uniformly over depth at the onset, all species live close to the surface where ample light is available, while the populations usually decrease towards the bottom, because of darkness (Figure 5). This behaviour was already observed in [13].



a) Example 1. b) Example 2.

Figure 5: Distribution over depth (x and y fixed) of Sinking species (dash-dot line), Neutral species (dash line) and Buoyant species (solid line) at 5 days (left), 25 days (middle) and 150 days (right).

Horizontal distribution In contrast to the aforementioned two aspects (competition behaviour and vertical distribution), the two test examples show a substantial difference with respect to the horizontal distribution. For the Molenkamp test example we plotted in Figure 6 the horizontal distribution of the three species just below the surface (at $z = 0.25m$) after 1 day of simulation (left column) and at steady

state (right column). We observe that the shapes of the various species are quite similar, whereas the amplitudes largely differ. The change of these amplitudes is in accordance with the time-behaviour of the biomasses (see Figure 4a).

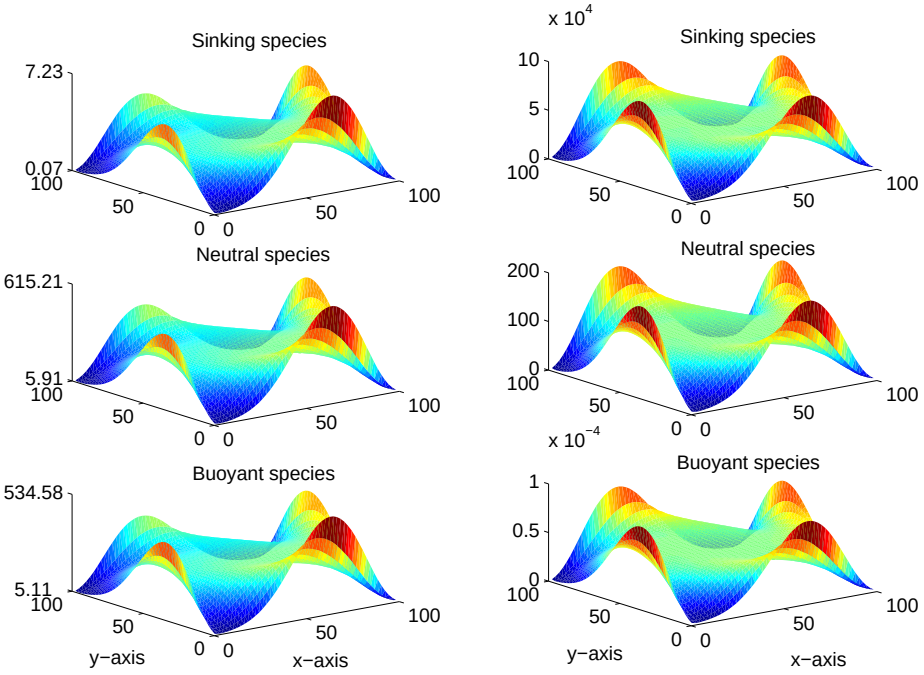


Figure 6: First test example: Populations of phytoplankton species at $z = 0.25m$ at $t = 1$ day (left) and $t = 150$ days (right).

We remark that the particular horizontal shape of the species shown in the right column of Figure 6 was already observed after a few days of simulation and did not change significantly during the remaining part of the integration. Apparently, already after a short period of time all terms involving spatial derivatives are in balance and the only contribution to the right-hand side in (6) comes from the growth term.

Finally, the particular shape of the horizontal distribution can be partly explained by taking into account the boundary conditions that we imposed. At all boundaries we use a condition of the form $v\,\omega - D\partial\omega/\partial x = 0$, with v the particular velocity component (see (5)). Hence the sign of $\partial\omega/\partial x = v\,\omega/D$ is determined by the sign of v. Using the Molenkamp velocity components as defined in (19) the correct slopes at the boundaries can be recognized in the plots.

Moreover, we consider an interior point close to a corner point. Making a Taylor series expansion of the solution in this interior point around the solution in the corner point and using the same reasoning as above for the sign of the derivatives, it can be proved that the solution in the corner point must vanish as shown in Figure 6.

For the second test example the same information is given in Figure 7. Again, the three species show a horizontal distribution which is quite similar and the mutual amplitudes are in accordance with the time-behaviour of the biomasses (Figure 4b). The main difference with the first test example is that eventually the horizontal

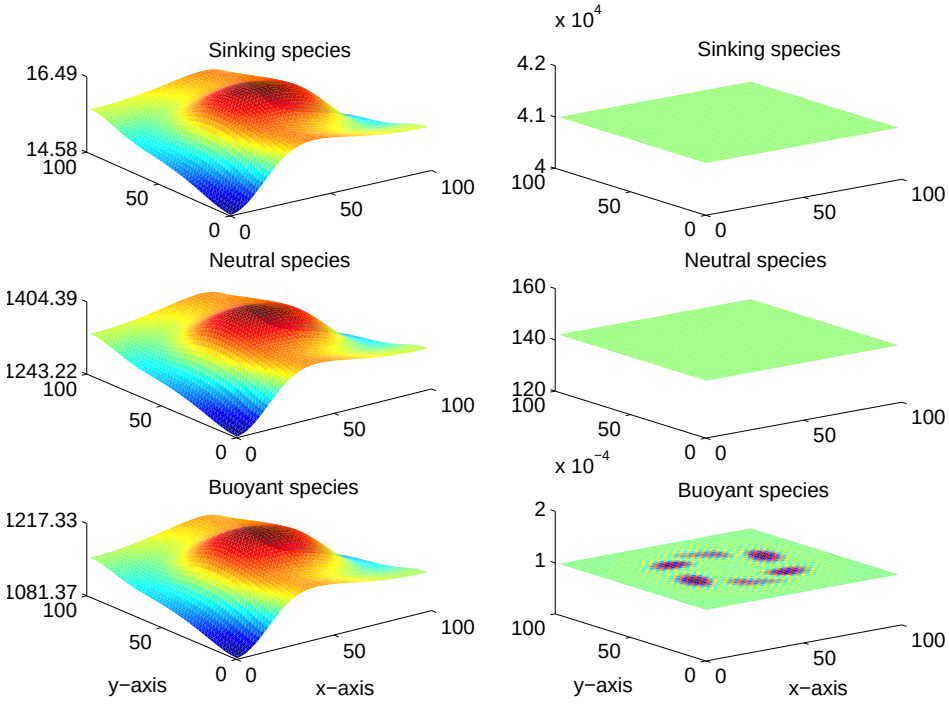


Figure 7: Second test example: Populations of phytoplankton species at $z = 0.25m$ at $t = 1$ day (left) and $t = 150$ days (right).

structure has disappeared: the solutions are completely flat in the horizontal (except for the Buoyant species; the dark regions in the plot are in fact "wiggles", i.e. numerical artifacts which will be discussed in Section 5). This horizontal solution "profile" is in agreement with the observation that all derivatives (in normal direction) of the solution at the boundary vanish for this velocity field (see (20)).

4.3 Numerical observations

To motivate the choice of the numerical techniques described in Sections 3.1 and 3.2, it is useful to look at certain characteristic numbers, such as the Cell Péclet number, the Courant-Friedrichs-Lewy (CFL) number and the stiffness of our problem. These numbers, which are discussed in many numerical text books on PDEs (see e.g. [14]), are listed in Table 3 (notice that both test examples have the same maximal velocities). Here we used the time step $\Delta t = 200(s)$, which turns out to be a realistic value.

Table 3: $|a|$ and $|b|$ are the maximal velocities in the horizontal direction, $|c_s|$ is the largest velocity of the three species, h: mesh size, D: diffusion coefficient, Δt: time step.

Characteristic number	Horizontal	Vertical
Péclet number	$\|a\|h/D = 1.2566$	$\|c_s\|h/D = 0.0115$
CFL number	$(\|a\|/h + \|b\|/h)\Delta t = 1.2566$	$\|c_s\|\Delta t/h = 0.0092$
Stiffness number	$8\,\Delta t\, D/h^2 = 4.0000$	$4\,\Delta t\, D/h^2 = 3.2000$

The diffusion coefficient, which plays an important role in the population dynamics of phytoplankton ([12],[13]), varies in a wide range from $10^{-1}(cm^2/s)$ in poorly mixed water, to $10^3(cm^2/s)$ in well mixed water. This results in a wide range for the Péclet number as well as for the stiffness number. Hence, for small D-values, the Péclet number will be large which motivates the choice for the third-order upwind-biased discretization. On the other hand, for large D-values, the stiffness number enforces to use an implicit method. Since we want to capture the whole spectrum of parameter values in one code, we decided to include upwind discretization as well as an implicit method.

As said in Section 3.2, for the time integration we have used VODE (extended with the Approximate Matrix Factorization technique). We observed that the behaviour of VODE is a bit erratic, especially when the code tries to integrate with a high order formula (orders 1 until 5 are available). Since only order 1 and 2 lead to an A-stable method, this behaviour suggests that the code encountered stability problems, caused by the complex eigenvalues originating from the (discretization of the) advection terms. Therefore, we also applied VODE with the maximal order set to 2. In the results described below, these two modes are denoted by VODE and BDF2, respectively.

We will now discuss the behaviour of both solvers when applied to our test examples on the time interval $[0, T]$, with $T = 2.5 \cdot 10^5(s)$. For the first test example, this interval corresponds to 5 rotations for each point at the surface. The results for the first test example are listed in Table 4.

From this table one can see that BDF2 is more efficient for the large tolerances, both in terms of CPU time and number of steps. Only for the very small tolerances VODE is more efficient, but those tolerance values are not realistic for our application (the spatial discretization is of order two).

Figure 8 presents an accuracy/cost plot. This figure confirms the better performance of BDF2 in the low accuracy range. The global relative error GRERR is the time integration error for the semi-discrete system (15). This error has been obtained by comparing the numerical solution with a reference solution, obtained with a very small tolerance value.

For a better understanding of these results, we will have a closer look at the time integration statistics for one particular RTOL value, i.e. 10^{-3}. Initially, VODE increases both the step size and the order as we can see in Figure 9a and Figure 9b. Then at steps 125 and 126 two consecutive convergence failures occur. VODE reacts with a reduction of the step size with a factor 4 for each failure, maintaining order

Table 4: Test example 1. Output for BDF2 (upper part) and VODE (lower part). RTOL: Relative tolerance, NST: Number of steps, NNI: Number of Newton iterations, NJE: Number of Jacobian evaluations, CFN: Number of non-linear convergence failures, ETF: Number of error test failures, Q: Order used in the final step, CPU: CPU time (seconds), GRERR: Global relative error in L_2 norm.

RTOL	NST	NNI	NJE	CFN	ETF	Q	CPU	GRERR
10^{-2}	855	1583	26	7	6	2	466	6.23e-3
10^{-3}	1219	1920	24	2	15	2	601	8.79e-4
10^{-4}	1767	2588	32	1	21	2	767	9.55e-5
10^{-5}	2950	3747	53	2	46	2	1233	8.46e-6
10^{-6}	5268	6532	93	2	78	2	2161	2.46e-6
10^{-2}	1547	1989	30	3	4	2	608	8.42e-3
10^{-3}	1789	2487	56	17	14	3	797	5.41e-4
10^{-4}	3471	5302	59	1	10	4	1642	4.58e-5
10^{-5}	2592	4080	48	2	14	3	1239	9.48e-6
10^{-6}	4584	5766	77	0	33	4	1904	1.02e-6

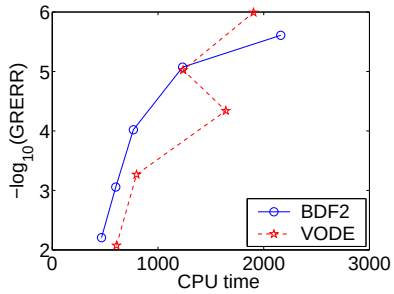
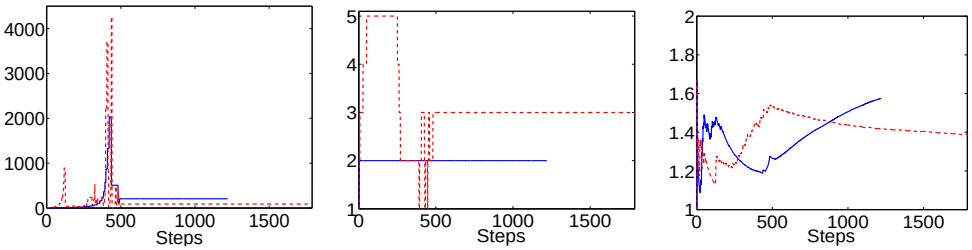


Figure 8: Test example 1. Efficiency plot for BDF2 and VODE.



a) The time step Δt. b) Order used. c) Averaged NNI/NST.

Figure 9: Test example 1. BDF2 (solid line) and VODE (dashed line) for RTOL=10^{-3}.

5. Next, several error test failures occur which cause a further reduction of the step size. Then VODE decides to lower the order to 4 and then to 3, due to an error test

failure. After a next convergence failure, the order is further reduced to 2.

As can be seen in Figures 9a and 9b, VODE seems to have trouble in finding an appropriate time step and an appropriate order, especially from step number $390 - 500$, where other convergence failures occur. Finally, beyond step number 500, the time step and order are fixed at $89(s)$ and 3, respectively. With this choice, VODE successfully reaches the end of the integration interval without any failure (see Figure 9a).

The behaviour of BDF2 is different. Here we observe a modest increase of the step size for the first 436 steps. Then, after a sudden increase of the step size, also BDF2 encounters two convergence failures at steps 436 and 481. After a reduction by a factor 4 for each failure, the step size is now appropriate to reach the end of the interval without any failure.

Observe that it is remarkable that the lower order BDF2 mode completes the integration with the constant step size $\Delta t = 206(s)$ while VODE, using order 3, seems to feel comfortable with the constant step size $\Delta t = 89(s)$ to satisfy the same tolerance criterion for the local error.

Finally, we will discuss the convergence behaviour which may have suffered from the fact that we replaced the Newton matrix in (17) by the right-hand side of (18).

As discussed in Section 3.2, the linear system in (17) has a huge number of entries. Therefore, we solved this complicated system by removing the growth term contributions and by successively solving three band-structured systems (see (18)) within each modified Newton iteration. In spite of this simplification, Newton's process still works very well: in average (taken over the steps), both modes need less than 2 Newton iterations per time step. This is shown in Figure 9c.

For the second test example, the codes show a behaviour which resembles the behaviour that we obtained for the first test example: again, VODE behaves rather erratic, in the sense that the global error is far from a monotone function of the costs. Furthermore, we again found that BDF2 is more efficient unless a very stringent

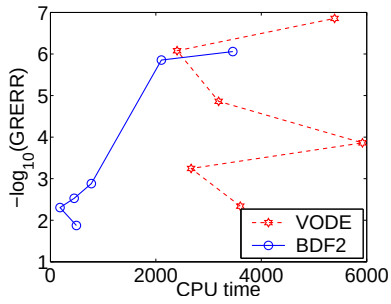


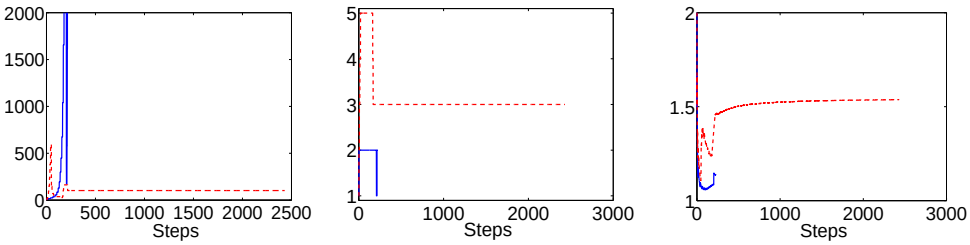
Figure 10: Test example 2. Efficiency plot for BDF2 and VODE.

tolerance-value is used. As can be seen in Figure 10, these two properties are even more pronounced than in the first test example. These observations indicate that VODE encounters stability problems when using a high-order formula, due to the

Table 5: Test example 2. Output for BDF2 (upper part) and VODE (lower part). The entries in this Table have the same meaning as in Table 4.

RTOL	NST	NNI	NJE	CFN	ETF	Q	CPU	GRERR
10^{-2}	422	815	12	3	0	2	493	1.34e-2
10^{-3}	226	258	7	2	1	2	182	4.96e-3
10^{-4}	557	684	15	3	3	2	450	2.97e-3
10^{-5}	1034	1180	24	5	3	2	776	1.32e-3
10^{-6}	2636	3374	52	5	1	2	2105	1.40e-6
10^{-2}	3248	5128	55	0	11	4	3602	4.58e-3
10^{-3}	2431	3737	42	1	6	3	2669	5.67e-4
10^{-4}	4009	5346	69	1	7	4	5919	1.37e-4
10^{-5}	3100	5199	52	0	9	4	3191	1.38e-5
10^{-6}	2589	3915	47	2	8	3	2403	8.35e-7

advection terms in the model. This conclusion is supported by the statistical data collected in Table 5: for RTOL=$10^{-2}, 10^{-4}$ and 10^{-5}, VODE integrates the last part of the integration interval using a fourth-order method and the resulting number of steps (and CPU time) are significantly higher than in case of a third-order formula (as VODE did for RTOL=10^{-3}, 10^{-6}). A close inspection of the performance of both solvers for RTOL=10^{-3} is shown in Figure 11.



a) The time step Δt. b) Order used. c) Averaged NNI/NST.

Figure 11: Test example 2. BDF2 (solid line) and VODE (dashed line) for RTOL=10^{-3}.

In Figure 11a we see that both codes try to substantially increase the step size, in particular BDF2. However, then a convergence failure reduces the step size to a more realistic value. VODE settles at a constant step size of $102(s)$ until the end of the integration interval. After an initial increase of the order to 5, VODE completes the integration with order 3. BDF2 prefers the second-order formula, except for a few steps at the end where Backward Euler has been used (see Figure 11b).

Finally, from Figure 11c we conclude that the use of Approximate Matrix Factorization in the Newton process requires not more than 1.5 iteration (in average) which is a quite satisfactory convergence behaviour.

5 Discussion

In summary, the integro-PDE system modelling the 3D phytoplankton dynamics is solved in two steps.

First, we discretized the spatial differential operators as well as the integral term. The diffusion term has been discretized symmetrically. For the advection term, we used the third-order upwind-biased discretization. The integral term is approximated using the repeated trapezoidal role.

Next, the resulting ODE system has been integrated in time. Due to the stiffness, we selected an implicit approach, viz. the family of BDF methods. The automatic integrator VODE, which has actually been used, is based on this family of implicit methods. Since the structure in the Jacobian gives rise to a laborious linear algebra owing to coupling of the competing species through the integral term of the integro-PDEs, we implemented the Approximate Matrix Factorization technique in VODE. In this way the expensive linear system is replaced by a sequence of three linear systems, each possessing a band structure, thus significantly reducing the computing time. The convergence behaviour of the resulting modified Newton process is still quite satisfactory.

Experiments with two test examples indicate that A-stability is a very useful property for the efficient and reliable solution of the phytoplankton dynamics model. Therefore, in the range of realistic (i.e., low) accuracies, the version of VODE in which we restricted the order to 2 shows a superior behaviour. However, based on what we experienced with this code, we believe that the numerical approach to solve this particular application can be improved upon. For example, as already mentioned in Section 4.2, the second test problem shows "wiggles", i.e., small oscillations superimposed on a smooth solution (see the right panel in Figure 7). The origin of such oscillations can be twofold: (i) spatial discretization and (ii) time integration.

For the spatial discretization of the advection terms we used the third-order upwind-biased scheme. Although rather accurate and better than a symmetric discretization, this choice does not guarantee that "wiggles" will be absent. In fact, this is only achieved with the first-order upwind discretization. This choice, however, has the disadvantage of low accuracy and the introduction of a large amount of artificial diffusion. A possible remedy to avoid the unwanted oscillations and to maintain a high-order is to combine the third-order upwind-biased discretization with limiters (see e.g. [14], p. 215). A disadvantage is, however, that such a technique introduces additional nonlinearity in the scheme which is a drawback when implemented in a fully implicit method such as used in VODE.

The second source of oscillations stems from the time integration method. Also here we encounter an order 1-barrier. Hence, the A-stability of the BDF2 method is not a sufficient condition to suppress "wiggles". Indeed, the only so-called positive method possessing this property is the BDF of order 1, i.e., Backward Euler. In passing, we remark that we also applied the Backward Euler scheme to the second test example and observed that the "wiggles" were still present, although to a much smaller extent. Hence, in our application, the spatial discretization seems to be the main reason for the "wiggles".

Based on the above considerations and the characteristics given in Table 3, an

implicit-explicit approach might result in a more efficient algorithm. With this approach we mean that the non-stiff terms in the model (i.e., growth and advection) are treated by an explicit time integration method and the (stiff) diffusion terms are treated implicitly. In this setting, a third-order upwind-biased discretization combined with limiters only marginally complicates the algorithm and Approximate Matrix Factorization can still be used to solve the 3D diffusion part. In this way we can avoid the Newton process since the diffusion terms are linear. This approach will be subject of future research.

Acknowledgements

The authors are grateful to Jan Verwer and Barry Koren for their constructive remarks.

References

[1] K.R. Arrigo, D.H. Robinson, D.L. Worthen, R.B. Dunbar, G.R. DiTullio, M. van Woert and M.P. Lizotte, *Phytoplankton community structure and the drawdown of nutrients and CO_2 in the Southern Ocean*, Science 283, 365-367, 1999.

[2] R.M. Beam, R.F. Warming, *An implicit finite-difference algorithm for hyperbolic systems in conservation-law form*, J. Comput. Phys. 22, 87-110, 1976.

[3] P.N. Brown, G.D. Byrne and A.C. Hindmarsh, *VODE: a variable-coefficient ODE solver*, SIAM Journal on Scientific and Statistical Computing 10, 1038-1051, 1989.

[4] E.G. D'Yakonov, *Difference systems of second order accuracy with a divided operator for parabolic equations without mixed derivatives*, USSR Comput. Math. Math. Phys. 4(5), 206-216, 1964.

[5] P.G. Falkowski, *The ocean's invisible forest: marine phytoplankton play a critical role in regulating the earth's climate. Could they also help stop global warming?*, Scientific American 287, Number 2, 38-45, August 2002.

[6] P.G. Falkowski, R.T. Barber and V. Smetacek, *Biogeochemical controls and feedbacks on ocean primary production*, Science 281, 200-206, 1998.

[7] P.J.S. Franks and C.S. Chen, *A 3-D prognostic numerical model study of the Georges Bank ecosystem (II). Biological-physical model*, Deep-Sea Research II 48, 457-482, 2001.

[8] C. van den Hoek, D.G. Mann and H.M. Jahns, *Algae: an Introduction to Phycology.* Cambridge University Press, 1995.

[9] P.J. van der Houwen, B.P. Sommeijer, *Approximate factorization for time-dependent partial differential equations*, Journal of Computational and Applied Mathematics 128, 447-466, 2001.

[10] J. Huisman, M. Arrayás, U. Ebert and B. Sommeijer, *How do sinking phytoplankton species manage to persist?*, American Naturalist 159, 245-254, 2002.

[11] J. Huisman, R.R. Jonker, C. Zonneveld and F.J. Weissing, *Competition for light between phytoplankton species, experimental tests of mechanistic theory*, Ecology 80, 211-222, 1999.

[12] J. Huisman, P. van Oostveen and F.J. Weissing, *Species dynamics in phytoplankton blooms: incomplete mixing and competition for light*, American Naturalist 154, 46-68, 1999.

[13] J. Huisman, B.P. Sommeijer, *Population dynamics of sinking phytoplankton in light-limited environments: simulation techniques and critical parameters*, J. Sea Research 48, 83-96, 2002.

[14] W. Hundsdorfer, J.G. Verwer, *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*, Springer Series in Computational Mathematics 33, Springer-Verlag, 2003.

[15] J.T.O. Kirk, *Light and Photosynthesis in Aquatic Ecosystems*, Second edition, Cambridge University Press, 1994.

[16] P. Korpinen, M. Kiirikki, P. Rantanen, A. Inkala and J. Sarkkula, *High resolution 3D-ecosystem model for the Neva Bay and estuary: model validation and future scenarios*, Oceanologia 45, 67-80, 2003.

[17] D.J. McGillicuddy Jr, A.R. Robinson and J.J. McCarthy, *Coupled physical and biological modeling of the spring bloom in the North Atlantic (II). 3-Dimensional bloom and post-bloom processes*, Deep-Sea Research I 42, 1359-1398, 1995.

[18] J.R. Moisan, E.E. Hofmann and D.B. Haidvogel, *Modeling nutrient and plankton processes in the California Coastal Transition Zone (II). A three-dimensional physical-bio-optical model*, Journal of Geophysical Research-Oceans 101, 22677-22691, 1996.

[19] A.E.P. Veldman, *Computational Fluid Dynamics*, Lecture Notes, University of Groningen, The Netherlands, 2001.