Time adaptive numerical solution of a highly non-linear degenerate cross-diffusion system arising in multi-species biofilm modelling†

MARYAM GHASEMI[1], STEFANIE SONNER[2]
and HERMANN J. EBERL[3]

[1]*Department Applied Mathematics, University of Waterloo, Waterloo Ontario, Canada*
email: m23ghase@uwaterloo.ca
[2]*Department Mathematics, Radboud University, Nijmegen, the Netherlands*
email: stefanie.sonner@uni-graz.at
[3]*Department Mathematics and Statistics, University of Guelph, Guelph Ontario, Canada*
email: heberl@uoguelph.ca

We propose a numerical method for the simulation of a quasi-linear parabolic biofilm model that exhibits three non-linear diffusion effects: (i) a power law degeneracy, (ii) a super diffusion singularity and (iii) non-linear cross-diffusion. The method is based on a spatial Finite Volume discretisation in which cross-diffusion terms are formally treated as convection terms. Time-integration of the resulting semi-discretised system is carried out using an error-controlled, time-adaptive, embedded Rosenbrock–Wanner method. We compare several variants of the method and two variants of the model to investigate how details such as the choice cross-diffusion coefficients, and specific variants of the time integrator affect simulation time.

1 Introduction

Bacterial biofilms are microbial communities attached to an immersed surface. Depending on the context, they can be formed by one or multiple species. A characteristic of biofilms is that they produce gel-like extracellular polymeric substances in which they are embedded and which offers them protection against chemical and mechanical washout and against antimicrobial agents [13, 17, 35]. Biofilms are prevalent in natural, industrial, and hospital settings. Depending on the situation, they can be harmful or beneficial. Biofilms can cause corrosion and clogging in drinking water pipelines [31]. Dental plaque, which provokes tooth decay and gum disease, is a consequence of biofilm formation on teeth. Biofilm formation in the body can lead to failure of medical implants and

too difficult to eradicate bacterial infections [2, 5, 28]. On the other hand, the adsorption and absorption properties and enhanced mechanical stability of biofilms are beneficially applied in wastewater engineering to remove harmful components from contaminated water [26].

Although the term biofilm suggests homogeneous film-like layers, biofilms on the meso-scale (10 μm $\sim$ 1 mm, the actual biofilm scale) can be rather heterogeneous structures formed by colonies that may merge as they grow and expand [18]. Biological factors such as nutrient availability, maximum cell density, and specific growth rate determine the architecture of biofilms [6, 18].

In addition to the architectural heterogeneity, many biofilms, in particular multi-species systems, are also heterogeneous with respect to internal biomass distribution, for example, anaerobic pockets within otherwise aerobic biofilms can be found, etc. Such properties have been used for example in wastewater engineering to design biofilm reactors in which processes can happen concurrently which in batch reactors would need to be performed subsequently and often in different vessels; an example for this is simultaneous nitrification and denitrification in biofilm reactors, e.g., [36].

A mesoscopic, multi-dimensional mathematical model for multi-species biofilms has been proposed in [22]. It has been derived both from the view point of a biofilm as a spatially structured population, and as a mechanical object. The model is a highly non-linear diffusion–reaction system with three non-linear diffusion effects: (i) a power law degeneracy as in the porous medium equation when the biomass vanishes, (ii) a super-diffusion singularity where the biomass attains the maximum cell density, and (iii) cross-diffusion. The interplay of effects (i) and (ii) is responsible for the formation of a sharp biofilm/water interface that moves with finite speed and to ensure that the maximum cell density is not exceeded. They have already been included in the single-species model in [6], which this model generalizes. Cross-diffusion effects (iii) come into play in multi-species systems and describe the resistance to spatial spreading of one due to the presence of other components, and subsequent mixing of species.

Each of the three non-linear diffusion effects (i), (ii), and (iii) presents challenges for the numerical simulation. Due to (i), at the biofilm/water interface, biomass gradients blow up, an effect that is prone to introduce interface smearing or mathematically and physically unrealistic oscillations in numerical schemes. Since this effect leads to solutions with only very little regularity, it is not clear that methods that have been originally designed and proved to be suitable for problems with sufficiently high regularity can be reliably applied. The super-diffusion singularity (ii) leads to blow-up of the diffusion coefficients and the associated numerical problems. Even if it can be proved that solutions never attain this singularity, numerical approximations are prone to overshoot the singularity in its vicinity, in particular if the time steps are chosen too large. Finally, (iii), cross-diffusion equations do normally not have maximum principles, and positivity of solutions must be established by different means. In the straightforward numerical discretisation of this type of problems this can result in schemes that are not positivity preserving, for example, because in the discretisation system matrices might be obtained that do not have M-matrix properties.

In [10], based on a regularisation approach, it was shown that a semi-discrete approximation of the single-species biofilm model of [6] with properties (i) and (ii) [but not (iii)] has sufficient regularity to allow the application of higher order, error controlled time

adaptive schemes, such as embedded Rosenbrock–Wanner (ROW) methods. In [9], this was generalized to a multi-species biofilm model without cross-diffusion. Our objective is now to investigate whether the approach can be extended to the cross-diffusion biofilm model from [22] that also includes effect (iii). This would overcome the restriction of the semi-implicit numerical method that was introduced in [21], which requires small enough fixed time-steps.

Although there have been several numerical methods for cross-diffusion problems proposed in the literature, these cannot be readily adapted to the problem at hand without further investigation. Most of these methods have been developed for problems of Shigesada–Kawasaki–Teramoto type, i.e., models with polynomial fluxes, e.g., [1,3,4,11,12]. The biofilm model that we study is highly degenerate and the fluxes have singularities. Very little is known about the numerical treatment of such equations and whether existing methods can deal with these effects reliably requires additional investigation. To contribute to this question is the overarching goal of our study.

The outline of the paper is as follows: In the next Section 2, we will state the model problem. Subsequently, we propose in Section 3, a semi-discrete approximation that is obtained from spatial discretisation with a Finite Volume method. In Section 4, we will investigate the applicability of time-adaptive embedded ROW methods to solve the semi-discrete problem; this includes a convergence study by grid refinement and a comparison against a special solution of the model that is obtained from the single-species model. The density dependent cross- and self-diffusion coefficients that we use here differ from those that were introduced and used in [21, 22]. We conduct a comparative simulation of both approaches with view on computational efficiency. As there are several ROW methods, we compare three of them, of the same order and with the same number of stages, with the goal to investigate how strongly the choice of a specific representative of this class of methods can affect computation time performance. ROW methods require repeated evaluation of the Jacobian of the underlying system. To provide these can be a cumbersome task. Therefore, we include also a comparison of the performance of the method with a numerical approximation of the Jacobian against the method with an analytical expression.

The focus of our work is on practical computational aspects of the numerical method, rather than on a thorough and rigorous theoretical analysis of the algorithm and the underlying model.

2 Mathematical model

A cross-diffusion model for the competition of two biomass species in a biofilm for a shared growth limiting substrate was introduced in [22]. It is an extension of the single-species biofilm model in [6], in which competition for space between both species leads to cross-diffusion terms that describe the resistance to the movement of one species due to the presence of the other species. The model is a density-dependent degenerate diffusion–reaction equation over domain $\Omega \subset \mathbb{R}^2$. The dependent variables are the volume fractions a and b occupied by biomass of the two species, and the concentration of nutrient, c. The total volume fraction occupied by the biofilm is denoted by $m := a + b$. Domain Ω can be divided into sub-domain $\Omega_1(t) = \left\{ (x, y) \in \Omega \subset \mathbb{R}^2 : m(t, x, y) = 0 \right\}$ that describes

the aqueous phase (bulk liquid, channels, and pores of a biofilm) without biomass, and sub-domain $\Omega_2(t) = \{(x, y) \in \Omega \subset \mathbb{R}^2 : m(t, x, y) > 0\}$, which is the actual biofilm with positive biomass volume fraction. The independent variables $t \geqslant 0$ and $(x, y) \in \Omega$ denote here time and spatial location, respectively. Both regions are separated by the biofilm/water interface $\Gamma(t) = \partial\Omega_1(t) \cap \partial\Omega_2(t)$ that changes as the biofilm grows. Note that neither Ω_1 nor Ω_2 need to be connected, and Γ can consist of several disjoint segments. As is usual in mathematical models of biofilms, the extracellular polymeric substances that is produced by the bacteria is subsumed in the biofilm volume fraction in this biofilm model. The governing equations read [22]

$$\frac{\partial a}{\partial t} = \nabla\big(D_{11}(a, b)\nabla a + D_{12}(a, b)\nabla b\big) + \mu_a \frac{ca}{\kappa_a + c} - K_a a, \tag{2.1}$$

$$\frac{\partial b}{\partial t} = \nabla\big(D_{21}(a, b)\nabla a + D_{22}(a, b)\nabla b\big) + \mu_b \frac{cb}{\kappa_b + c} - K_b b, \tag{2.2}$$

$$\frac{\partial c}{\partial t} = \nabla\big(D_c(a + b)\nabla c\big) - \frac{\mu_a a^\infty}{\gamma_a} \frac{ca}{\kappa_a + c} - \frac{\mu_b b^\infty}{\gamma_b} \frac{cb}{\kappa_b + c}, \tag{2.3}$$

where $a^\infty [gm^{-3}]$ and $b^\infty [gm^{-3}]$ are the maximum cell densities, usually measured in equivalents of chemical oxygen demand. In (2.1) and (2.2), $\mu_{a,b}[d^{-1}]$ are the maximum growth rates of species a and b, the corresponding decay rates are $K_{a,b}[d^{-1}]$. Growth of both components is described by standard Monod kinetics, where $\kappa_{a,b}[gm^{-3}]$ are the half saturation concentrations and the yield coefficients for nutrient uptake rates are given by $\gamma_{a,b}[-]$. The self- and cross-diffusion coefficients $D_{ij}, [m^2 d^{-1}], i, j \in \{1, 2\}$ in (2.1), (2.2) are defined as [21]

$$\begin{cases} D_{11}(a, b) = p(m)q(m) + a\big(p(m)q'(m) - q(m)p'(m)\big), \\ D_{12}(a, b) = a\big(p(m)q'(m) - q(m)p'(m)\big), \\ D_{21}(a, b) = b\big(p(m)q'(m) - q(m)p'(m)\big), \\ D_{22}(a, b) = p(m)q(m) + b\big(p(m)q'(m) - q(m)p'(m)\big), \end{cases} \tag{2.4}$$

where $p(m)$ and $q(m)$ are transfer functions describing the local movement of the species a and b from one site to a neighbouring site.

The transfer function p describes the attractivity of the current site to incoming individuals and q is a measure of the incentive for individuals to leave their current position. Therefore, they typically have the following properties:

$$p' \leqslant 0, \qquad\qquad p(1) = 0,$$
$$q' \geqslant 0, \qquad\qquad q(0) = 0.$$

The first property encodes that biomass cannot be deposited into an already full site, the second one that at current low biomass densities newly produced biomass is not transferred into neighbouring sites. An additional requirement on the functions p and q is that the non-linearities must be strong enough to avoid overcrowding, i.e., they must guarantee that under equations (2.1)–(2.3), the biomass density $a + b$ is bounded by unity. Furthermore, we require that in the absence of one species, the cross-diffusion model reduces to the single-species biofilm model of [6] and, likewise, that if both species have

the exact same growth behaviour, that $a + b$ is a solution of that single-species model. This leads to

$$D(m) = \delta \frac{m^4}{(1-m)^4} = p(m)q(m) + m\big(p(m)q'(m) - q(m)p'(m)\big), \tag{2.5}$$

where function $D(m)$ is the density dependent diffusion-coefficient of the single-species model in [6]. There are infinitely many pairs p and q that satisfy these requirements for $0 \leqslant m < 1$. In [22], which follows the derivation for the single-species model in [15] closely, instead of specifying p and q, the product $\Phi(m) = p(m) \cdot q(m)$ and the function $\Psi(m) = p(m)q'(m) - p'(m)q(m)$ are determined, leading to terms involving Hypergeometric Functions, see also Section 4.4. We aim here to establish instead a pair of functions p, q that has the same monotonicity properties and satisfies (2.5), but is numerically less expensive to evaluate. Prompted by equation (2.5), we make the ansatz

$$p(m) = c_1(1-m)^\gamma, \qquad q(m) = c_2 \frac{m^\alpha}{(1-m)^\beta},$$

where c_1, c_2, γ, α, and β are positive constants. These choices of p and q lead to

$$
\begin{aligned}
p' &= -c_1\gamma(1-m)^{\gamma-1}, \\
q' &= c_2 \frac{m^{\alpha-1}}{(1-m)^{\beta+1}}(\alpha + m(\beta - \alpha)), \\
D(m) &= c_1 c_2 \frac{m^\alpha}{(1-m)^{\beta-\gamma+1}}(1 + \alpha + m(\beta + \gamma - 1 - \alpha)),
\end{aligned}
\tag{2.6}
$$

Choosing α, β and γ such that $\beta + \gamma = 1 + \alpha$ results in

$$D(m) = c_1 c_2 (\beta + \gamma) \frac{m^{\beta+\gamma-1}}{(1-m)^{\beta-\gamma+1}},$$

Comparing with $D(m) = \delta \frac{m^4}{(1-m)^4}$, we obtain

$$c_1 c_2 (\beta + \gamma) = \delta, \qquad \beta + \gamma - 1 = 4, \qquad \beta - \gamma + 1 = 4.$$

A set of coefficients that satisfies this is

$$\alpha = 4, \qquad \beta = 4, \qquad \gamma = 1, \qquad c_1 = \frac{\delta}{5}, \qquad c_2 = 1.$$

The transfer functions are then

$$p(m) = \frac{\delta}{5}(1 - m), \qquad q(m) = \frac{m^4}{(1-m)^4},$$

and we obtain

$$
\begin{cases}
D_{11}(a,b) = \frac{\delta}{5}\left(\frac{m^4}{(1-m)^3} + a\frac{m^3(4+m)}{(1-m)^4}\right), \ D_{12}(a,b) = \frac{\delta}{5}a\frac{m^3(4+m)}{(1-m)^4}, \\[2mm]
D_{21}(a,b) = \frac{\delta}{5}b\frac{m^3(4+m)}{(1-m)^4}, \ D_{22}(a,b) = \frac{\delta}{5}\left(\frac{m^4}{(1-m)^3} + b\frac{m^3(4+m)}{(1-m)^4}\right).
\end{cases}
\tag{2.7}
$$

These self- and cross-diffusion functions are rational, involving only polynomial evaluations and divisions, but not the more costly Hypergeometric Functions, which were required for the diffusion coefficients in [22]. We will compare the solutions obtained by both approaches below in Section 4.4 in a simulation experiment.

The diffusion of substrate in a biofilm is slower than that in the surrounding aqueous phase [32]. To account for this, we use for the substrate diffusion coefficient D_c in (2.3) linear interpolation between the values of the diffusion coefficient of the dissolved substrate in the fully developed biofilm, $D_c(1)$, and in the liquid region, $D_c(0)$, which results in

$$D_c(m) = D_c(0) + m\big(D_c(1) - D_c(0)\big) = D_c(0)\big(1 + m(\rho - 1)\big), \qquad (2.8)$$

in which $\rho := \frac{D_c(1)}{D_c(0)}$ is the ratio of diffusivity of nutrient in biofilm and liquid. Although the nutrient diffusion coefficient depends on m, its boundedness from above and below by constants, $0 < D_c(1) \leqslant D_c(m) \leqslant D_c(0) < \infty$, indicates that it behaves essentially Fickian [22].

To study the mathematical model (2.1)–(2.3), we consider a rectangular domain $\Omega = [0, L] \times [0, H]$. The substratum, on which biofilm colonies form is the bottom boundary, $y = 0$. We assume that the substratum is impermeable to biomass and dissolved substrate. A symmetry boundary condition at the lateral boundaries, $x = 0$ and $x = L$, is imposed for all dependent variables. This allows us to view the domain as a part of a continuously repeating, symmetrically mirrored larger domain, or alternatively as a small pocket in which a biofilm forms. At the top boundary, $y = H$, a homogeneous Dirichlet condition for both biomass species and a Robin condition for the nutrient supply is considered that reflects that the substrate is added through the top boundary. Thus, the imposed boundary conditions on domain $\Omega = [0, L] \times [0, H]$ are defined as follows:

$$\begin{cases} \partial_n a = \partial_n b = \partial_n c = 0 & \text{at } x = 0, x = L \text{ and } y = 0, \\ a = b = 0, \quad c + \lambda \partial_n c = c^\infty & \text{at } y = H, \end{cases} \qquad (2.9)$$

where c^∞ is the bulk substrate concentration and ∂_n denotes the outward normal derivative. Moreover in (2.9), λ is the concentration boundary layer thickness (i.e., essentially the reciprocal of the external mass transfer coefficient) which is related to the bulk flow velocity. A small bulk flow velocity that brings less nutrient supply via convective transport in the (not explicitly considered) bulk phase implies a thick concentration boundary layer, while a thin concentration boundary layer that provides more substrate represents fast bulk flow [7]. Default values of the model parameters and their description are shown in Table 1.

The Partial Differential Equation (PDE) model in (2.1)–(2.2) shows three non-linear diffusion effects: (i) degeneracy as in the porous medium equation, i.e., D_{ij}, $i, j \in \{1, 2\}$ vanish as the dependent variables a, b vanish, (ii) a super diffusion singularity as $a + b$ approaches to unity, and (iii) cross-diffusion. The effects (i) and (ii) have already been present in the underlying single-species prototype biofilm growth model of [6], and in an earlier multi-species model without cross-diffusion effects [14, 20]. The porous medium degeneracy (i) guarantees that the biofilm does not spread notably if there is locally space available for biomass to accumulate new biomass, and it is also responsible for the formation of a sharp interface $\Gamma(t)$ between biofilm and surrounding liquid, i.e.,

Table 1. *Model parameters in the simulation of model (2.1–2.3)*

Parameter	Symbol	Value	Unit	Source
Maximum specific growth rate of species a	μ_a	6	d^{-1}	[22]
Maximum specific growth rate of species b	μ_b	9	d^{-1}	[22]
Half saturation concentration for species a	κ_a	0.4	gm^{-3}	[22]
Half saturation concentration for species b	κ_b	0.8	gm^{-3}	[22]
Lysis rate	K_a, K_b	0.4	d^{-1}	[22]
Maximum biomass density	a^∞, b^∞	10^4	gm^{-3}	[22]
Yield coefficient	γ_a, γ_b	0.63	−	[22]
Bulk substrate concentration	c^∞	40	gm^{-3}	[22]
Substrate diffusion coefficient in water	$D_c(0)$	10^{-4}	$m^2 d^{-1}$	[22]
Substrate diffusion coefficient in biofilm	$D_c(1)$	0.8×10^{-4}	$m^2 d^{-1}$	[22]
Biomass motility coefficient	δ	10^{-12}	$m^2 d^{-1}$	[6]

initial data with compact support lead to solutions with compact support. The super-diffusion singularity (ii) forces the total biomass fraction to be bounded by a constant strictly less than 1. To see that this behaviour carries over to the dual-species model with cross-diffusion (2.1)–(2.2) assume that a, b, c is a non-negative solution. Then, due to the comparison principle, $0 \leqslant c \leqslant c^\infty$. We have

$$
\begin{aligned}
\frac{\partial(a+b)}{\partial t} &= \nabla(D_{11}(a,b)\nabla a + D_{12}(a,b)\nabla b) + \mu_a \frac{ca}{\kappa_a + c} - K_a a \\
&+ \nabla(D_{21}(a,b)\nabla a + D_{22}(a,b)\nabla b) + \mu_b \frac{cb}{\kappa_b + c} - K_b b \\
&\leqslant \nabla([D_{11}(a,b) + D_{22}(a,b)]\nabla a + [D_{12}(a,b) + D_{21}(a,b)]\nabla b) \\
&+ \mu \frac{c(a+b)}{\kappa + c} - K(a+b),
\end{aligned}
$$

where $\mu := \max\{\mu_a, \mu_b\}$, $\kappa := \min\{\kappa_a, \kappa_b\}$, $K := \min\{K_a, K_b\}$. Defining now $m := a + b$ and considering the definitions (2.4) and (2.5), this inequality can be rewritten as

$$
\frac{\partial m}{\partial t} \leqslant \nabla(D(m)\nabla m) + \mu \frac{cm}{\kappa + c} - Km. \tag{2.10}
$$

Thus, the solution of the single-species biofilm model

$$
\frac{\partial \tilde{m}}{\partial t} = \nabla(D(\tilde{m})\nabla \tilde{m}) + \mu \frac{c^\infty \tilde{m}}{\kappa + c^\infty} - K\tilde{m}. \tag{2.11}
$$

is an upper solution for total biomass fraction $m := a + b$. The results of [8] on the single-species model imply that $\tilde{m}$ and therefore m remains separated from the fast diffusion singularity, i.e., $a + b \leqslant 1 - \xi$ for some constant $0 < \xi \ll 1$. Moreover, initial data with compact support lead to solutions with compact support, i.e., to a finite speed of interface propagation.

3 Numerical methods

3.1 Preliminaries

Each of the non-linear diffusion effects (i)–(iii) brings its own numerical challenge. Due to the power law degeneracy (i), biomass gradients blow up at the biofilm/water interface $\Gamma(t)$. This effect is prone to introduce interface smearing or mathematically and physically unrealistic oscillations in numerical discretisation schemes. Since this effect leads to solutions with only very little regularity, it is not clear that methods that have been originally designed and proved to be useful for problems with sufficiently high regularity can be reliably applied. The super-diffusion singularity (ii) leads to blow-up of the diffusion coefficients if $a + b = 1$ and the associated numerical problems if the biomass comes close to this singularity, $a + b \approx 1$. Although it can be proved that solutions of the continuous model never attain this singularity, numerical approximations might overshoot the singularity in its vicinity, in particular if the time steps are chosen too large, leading to a breakdown of the method. This is in particular a problem for methods with fixed time-steps, and also for explicit time-adaptive methods because of the stiffness induced by the singularity, cf. [10] and the discussion therein for the single-species case.

In [10], using regularisation techniques, a semi-discrete approximation of the underlying single-species biofilm model was analysed, which is obtained by spatial discretisation with a Finite Volume method. It was shown that adaptive, error controlled implicit solvers for the resulting initial value problem reliably keep the numerical solution separated from the singularity, i.e., prevent its overshooting. Furthermore, it was shown that the interface smearing effects introduced are negligibly small, i.e., the method was found suitable for problems with properties (i) and (ii). In [9], this was extended to the multi-species model without cross-diffusion. Per (2.10) the single-species model provides an upper bound on the solution of the cross-diffusion model, which suggests the application of the same time-integration approach to the problem at hand.

For cross-diffusion problems, i.e., problems with effect (iii), maximum principles do not exist. This is reflected in the numerical treatment by instabilities that arise if the cross-diffusion terms are discretised by the same second-order approximations that are routinely applied for the self-diffusion terms. One approach to overcome this difficulty, which was employed in [21] for the cross-diffusion biofilm model at hand together with a semi-implicit, time-integration scheme with fixed time-steps, is to formally treat the cross-diffusion terms as convection terms and apply upwinding techniques. It was shown that this spatial treatment avoids spurious oscillations, keeps interface smearing effects low and describes the biomass gradient blow-up effect with acceptable accuracy.

In order to obtain an improved numerical method for the cross-diffusion biofilm model (2.1)–(2.3), we propose to combine the spatial discretisation strategy of [21] with the time-integration method of [10]. To this end, we rewrite the model as

$$\frac{\partial a}{\partial t} = \nabla \big(D_{11}(a, b) \nabla a - \omega_a a \big) + \mu_a \frac{ca}{\kappa_a + c} - K_a a, \tag{3.1}$$

$$\frac{\partial b}{\partial t} = \nabla \big(D_{22}(a, b) \nabla b - \omega_b b \big) + \mu_b \frac{cb}{\kappa_b + c} - K_b b, \tag{3.2}$$

$$\frac{\partial c}{\partial t} = \nabla\left(D_c(a,b)\nabla c\right) - \frac{\mu_a a^\infty}{\gamma_a}\frac{ca}{\kappa_b + c} - \frac{\mu_b b^\infty}{\gamma_b}\frac{cb}{\kappa_b + c}, \tag{3.3}$$

where $\omega_{a,b}$ are the cross-diffusion velocities

$$\omega_a = -\psi(m)\nabla b, \qquad \omega_b = -\psi(m)\nabla a,$$

with function $\psi(m)$ defined via (2.5) as

$$\psi(m) := p(m)q'(m) - q(m)p'(m).$$

For the numerical treatment, we first derive a dimensionless version of the model using the scalings

$$\tilde{x} = \frac{x}{L}, \qquad \tilde{t} = t\mu_a, \qquad \tilde{c} = \frac{c}{c^\infty},$$

where L is the length of the computational domain and $\frac{1}{\mu_a}$ is the characteristic time scale of biomass growth of type a. Note that the biomass fractions a and b are originally defined as non-dimensional variables. By these choices, the non-dimensionalized model reads

$$\frac{\partial a}{\partial \tilde{t}} = \tilde{\nabla}(\tilde{D}_{11}(a,b)\tilde{\nabla}a - \tilde{\omega}_a a) + \frac{\tilde{c}a}{\tilde{\kappa}_a + \tilde{c}} - \tilde{K}_a a,$$
$$\frac{\partial b}{\partial \tilde{t}} = \tilde{\nabla}(\tilde{D}_{22}(a,b)\tilde{\nabla}b - \tilde{\omega}_b b) + \tilde{\mu}_b\frac{\tilde{c}b}{\tilde{\kappa}_b + \tilde{c}} - \tilde{K}_b b,$$
$$\frac{\partial \tilde{c}}{\partial \tilde{t}} = \tilde{\nabla}(\tilde{D}_c(m)\tilde{\nabla}\tilde{c}) - v_a\frac{\tilde{c}a}{\tilde{\kappa}_a + \tilde{c}} - v_b\frac{\tilde{c}b}{\tilde{\kappa}_b + \tilde{c}}, \tag{3.4}$$

with

$$\tilde{\kappa}_a = \frac{\kappa_a}{c^\infty}, \qquad \tilde{\kappa}_b = \frac{\kappa_b}{c^\infty}, \qquad \tilde{K}_a = \frac{K_a}{\mu_a} \qquad \tilde{K}_b = \frac{K_b}{\mu_a},$$

$$\tilde{\mu}_b = \frac{\mu_b}{\mu_a} \qquad v_a = \frac{a^\infty}{\gamma_a c^\infty}, \qquad v_b = \frac{\mu_b b^\infty}{\mu_a \gamma_b c^\infty},$$

$$\tilde{D}_{ii} = \frac{D_{ii}}{\mu_a L^2}, \quad i \in \{1,2\} \qquad \tilde{\omega}_{a,b} = \frac{\omega_{a,b}}{\mu_a L^2} \qquad \tilde{D}_c = \frac{D_c}{\mu_a L^2}.$$

For the numerical study and computer simulation, we will consider the non-dimensionalized model and drop the ($\tilde{\ }$) for simplicity.

3.2 Spatial discretization

In order to discretize the PDE model (3.4), we define a uniform grid of size $N \times M$ for the rectangular domain $[0,1] \times [0, H/L]$, i.e., we make implicitly the assumption $H = L \cdot M/N$. Integrating the first equation of (3.4) over each grid cell and using the Divergence Theorem gives for $i = 1, \ldots, N, j = 1, \ldots, M$

$$\frac{d}{dt}\int_{v_{i,j}} a\, dxdy = \int_{\partial v_{i,j}} J_n ds + \int_{\partial v_{i,j}} F_n ds + \int_{v_{i,j}} R(c)\, a\, dxdy, \tag{3.5}$$

where $v_{i,j}$ denotes the domain of the cell with grid index (i,j), $J_n = D_{11}(a,b)\partial_n a$ and $F_n = D_{12}(a,b)\partial_n b = -\omega_a a$ denote the outward normal self- and cross-diffusive fluxes across the grid cell boundary, and $R(c) = c/(\kappa_a + c) - K_a$ represents the reaction term. To find the area integral in (3.5), the dependent variables are evaluated at the centre of the grid cells,

$$A_{i,j} := a(t, x_i, y_j) \approx a\left(t, \left(i - \frac{1}{2}\right)\triangle x, \left(j - \frac{1}{2}\right)\triangle x\right),$$

$$B_{i,j} := b(t, x_i, y_j) \approx b\left(t, \left(i - \frac{1}{2}\right)\triangle x, \left(j - \frac{1}{2}\right)\triangle x\right),$$

and similarly for the shared nutrient concentration

$$C_{i,j} := c(t, x_i, y_j) \approx c\left(t, \left(i - \frac{1}{2}\right)\triangle x, \left(j - \frac{1}{2}\right)\triangle x\right).$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, M$ with $\triangle x = 1/N = H/LM$. For later usage, we define also $M_{i,j} := A_{i,j} + B_{i,j}$. Integrals in (3.5) are thus approximated by the midpoint rule, including the line integral which is obtained by considering every edge of the grid cell separately. To compute the self-diffusive flux between cells (i,j) and $(i+1,j)$, the self-diffusion coefficient $D_{11}(a,b)$ needs to be evaluated in the midpoint of the cell edge, for which we use the arithmetic averaging from the neighbouring grid cell centre points. The derivative of a and b across the cell edge are approximated by a central finite difference. Consequently, the self-diffusive flux across the edges of cell (i,j) for biomass fraction a, accounting for the boundary conditions is obtained as

$$J_{i,j+\frac{1}{2}} = \begin{cases} \frac{1}{2\triangle x}(D_{11}(A_{i,j+1}, B_{i,j+1}) + D_{11}(A_{i,j}, B_{i,j}))(A_{i,j+1} - A_{i,j}) & \text{for} \quad j < M, \\ -\frac{2}{\triangle x}D_{11}(0)A_{i,M} & \text{for} \quad j = M, \end{cases} \tag{3.6}$$

$$J_{i,j-\frac{1}{2}} = \begin{cases} 0 & \text{for} \quad j = 1, \\ \frac{1}{2\triangle x}(D_{11}(A_{i,j}, B_{i,j}) + D_{11}(A_{i,j-1}, B_{i,j-1}))(A_{i,j-1} - A_{i,j}) & \text{for} \quad j > 1, \end{cases} \tag{3.7}$$

$$J_{i+\frac{1}{2},j} = \begin{cases} \frac{1}{2\triangle x}(D_{11}(A_{i+1,j}, B_{i+1,j}) + D_{11}(A_{i,j}, B_{i,j}))(A_{i+1,j} - A_{i,j}) & \text{for} \quad i < N, \\ 0 & \text{for} \quad i = N, \end{cases} \tag{3.8}$$

$$J_{i-\frac{1}{2},j} = \begin{cases} 0 & \text{for} \quad i = 1, \\ \frac{1}{2\triangle x}(D_{11}(A_{i-1,j}, B_{i-1,j}) + D_{11}(A_{i,j}, B_{i,j}))(A_{i-1,j} - A_{i,j}), & \text{for} \quad i > 1. \end{cases} \tag{3.9}$$

Similarly, we can obtain the expressions for the self-diffusive flux for species b.

For the cross-diffusion terms, which we represented in (3.4) as a convection term, the velocity component v in the y-direction of velocity vector ω_a at the edge between cells (i,j) and $(i, j+1)$ is computed using the arithmetic averaging from the neighbouring grid

cell centre points as

$$
v_{i,j+\frac{1}{2}} = \begin{cases} \frac{1}{2\triangle x} \left(\psi(M_{i,j+1}) + \psi(M_{i,j}) \right) \left(B_{i,j+1} - B_{i,j} \right) & \text{for} \quad j < M, \\ -\frac{2}{\triangle x} \psi(0) B_{i,M} & \text{for} \quad j = M. \end{cases} \tag{3.10}
$$

The velocity component of ω_a in y-direction at the edge between cells $(i, j-1)$ and (i, j) can be obtained as

$$
v_{i,j-\frac{1}{2}} = \begin{cases} 0 & \text{for} \quad j = 1, \\ \frac{1}{2\triangle x} \left(\psi(M_{i,j}) + \psi(M_{i,j-1}) \right) \left(B_{i,j-1} - B_{i,j} \right) & \text{for} \quad j > 1, \end{cases} \tag{3.11}
$$

The velocity components of ω_a in x-direction, u, at the edges between cells $(i+1, j)$ and (i, j) and $(i-1, j)$ and (i, j) are calculated in the same manner as

$$
u_{i+\frac{1}{2},j} = \begin{cases} \frac{1}{2\triangle x} \left(\psi(M_{i+1,j}) + \psi(M_{i,j}) \right) \left(B_{i+1,j} - B_{i,j} \right) & \text{for} \quad i < N, \\ 0 & \text{for} \quad i = N, \end{cases} \tag{3.12}
$$

$$
u_{i-\frac{1}{2},j} = \begin{cases} 0 & \text{for} \quad i = 1, \\ \frac{1}{2\triangle x} \left(\psi(M_{i-1,j}) + \psi(M_{i,j}) \right) \left(B_{i-1,j} - B_{i,j} \right), & \text{for} \quad i > 1. \end{cases} \tag{3.13}
$$

Similarly, we compute the velocity components of ω_b in x- and y-directions.

Applying the first-order upwind scheme to discretise the cross-diffusion flux results

$$
F_{i+\frac{1}{2},j} = \begin{cases} u_{i+\frac{1}{2},j} A_{i,j} & \text{if} \quad u_{i+\frac{1}{2},j} \geqslant 0, \\ u_{i+\frac{1}{2},j} A_{i+1,j} & \text{if} \quad u_{i+\frac{1}{2},j} < 0, \end{cases} \tag{3.14}
$$

$$
F_{i-\frac{1}{2},j} = \begin{cases} u_{i-\frac{1}{2},j} A_{i-1,j} & \text{if} \quad u_{i-\frac{1}{2},j} \geqslant 0, \\ u_{i-\frac{1}{2},j} A_{i,j} & \text{if} \quad u_{i-\frac{1}{2},j} < 0, \end{cases} \tag{3.15}
$$

$$
F_{i,j+\frac{1}{2}} = \begin{cases} v_{i,j+\frac{1}{2}} A_{i,j} & \text{if} \quad v_{i,j+\frac{1}{2},j} \geqslant 0, \\ v_{i,j+\frac{1}{2}} A_{i,j+1} & \text{if} \quad v_{i,j+\frac{1}{2}} < 0, \end{cases} \tag{3.16}
$$

$$
F_{i,j-\frac{1}{2}} = \begin{cases} v_{i,j-\frac{1}{2}} A_{i,j-1} & \text{if} \quad v_{i,j-\frac{1}{2},j} \geqslant 0, \\ v_{i,j-\frac{1}{2}} A_{i,j} & \text{if} \quad v_{i,j-\frac{1}{2}} < 0. \end{cases} \tag{3.17}
$$

The cross-diffusion flux for b is approximated in the same way.

Putting the above together, we get the following ordinary differential equation for the biomass species a in the grid cell centre

$$
\frac{d}{dt} A_{i,j} = \frac{1}{\triangle x} \left(J_{i+\frac{1}{2},j} + J_{i-\frac{1}{2},j} + J_{i,j+\frac{1}{2}} + J_{i,j-\frac{1}{2}} \right)
$$

$$
- \frac{1}{\triangle x} \left(F_{i+\frac{1}{2},j} + F_{i-\frac{1}{2},j} + F_{i,j+\frac{1}{2}} + F_{i,j-\frac{1}{2}} \right) + R_{i,j} A_{i,j}, \tag{3.18}
$$

in which $R_{ij} = \frac{C_{i,j}}{\kappa_a + C_{i,j}} - K_a$ stands for the reaction term in the equation for a.

The equation for the nutrient concentration can be discretized in space following the same principle. The major difference in spatial discretization of nutrient is that we have a Robin boundary condition at the top boundary instead of homogeneous Dirichlet condition. We have then

$$\frac{d}{dt}C_{i,j} = \frac{1}{\triangle x}\left(\hat{J}_{i+\frac{1}{2},j} + \hat{J}_{i-\frac{1}{2},j} + \hat{J}_{i,j+\frac{1}{2}} + \hat{J}_{i,j-\frac{1}{2}}\right) - R_{1_{i,j}}A_{i,j} - R_{2_{i,j}}B_{i,j}, \qquad (3.19)$$

with

$$R_{1_{i,j}} = v_a \frac{C_{i,j}}{\kappa_a + C_{i,j}}, \quad R_{2_{i,j}} = v_b \frac{C_{i,j}}{\kappa_b + C_{i,j}}, \qquad (3.20)$$

and

$$\hat{J}_{i,j+\frac{1}{2}} = \begin{cases} \frac{1}{2\triangle x}(D_c(M_{i,j+1}) + D_c(M_{i,j}))(C_{i,j+1} - C_{i,j}) \text{ for } & j < M, \\ \frac{D_c(0)}{\triangle x}\left(\frac{2\triangle x}{2\lambda + \triangle x} - C_{i,M}\left(1 + \frac{\triangle x - 2\lambda}{\triangle x + 2\lambda}\right)\right) & \text{for } j = M, \end{cases} \qquad (3.21)$$

$$\hat{J}_{i,j-\frac{1}{2}} = \begin{cases} 0 & \text{for } j = 1, \\ \frac{1}{2\triangle x}(D_c(M_{i,j-1}) + D_c(M_{i,j}))(C_{i,j-1} - C_{i,j}) \text{ for } & j > 1, \end{cases} \qquad (3.22)$$

$$\hat{J}_{i+\frac{1}{2},j} = \begin{cases} \frac{1}{2\triangle x}(D_c(M_{i+1,j}) + D_c(M_{i,j}))(C_{i+1,j} - C_{i,j}) \text{ for } & i < N, \\ 0 & \text{for } i = N, \end{cases} \qquad (3.23)$$

$$\hat{J}_{i-\frac{1}{2},j} = \begin{cases} 0 & \text{for } i = 1, \\ \frac{1}{2\triangle x}(D_c(M_{i-1,j}) + D_c(M_{i,j}))(C_{i-1,j} - C_{i,j}), \text{ for } & i > 1. \end{cases} \qquad (3.24)$$

Introducing the lexicographical grid ordering

$$\pi : \{1,\ldots,N\} \times \{1,\ldots,M\} \to \{1,\ldots,NM\}, \ (i,j) \mapsto p = (i-1)N + j, \qquad (3.25)$$

and the vector notation $\mathbf{A} = (A_1,\ldots,A_{NM})$, $\mathbf{B} = (B_1,\ldots,B_{NM})$, and $\mathbf{C} = (C_1,\ldots,C_{NM})$ with $A_p := A_{\pi(i,j)} = A_{i,j}$, $B_p := B_{\pi(i,j)} = B_{i,j}$ and $C_p := C_{\pi(i,j)} = C_{i,j}$ for $i = 1,\ldots,N$, $j = 1,\ldots,M$, we obtain the coupled system of $3 \cdot N \cdot M$ ordinary differential equations

$$\begin{cases} \frac{d\mathbf{A}}{dt} = D_A\mathbf{A} + F_A\mathbf{A} + R_A(\mathbf{C})\mathbf{A} + \mathbf{b}_A, \\ \frac{d\mathbf{B}}{dt} = D_B\mathbf{B} + F_B\mathbf{B} + R_B(\mathbf{C})\mathbf{B} + \mathbf{b}_B, \\ \frac{d\mathbf{C}}{dt} = D_C\mathbf{C} - R_{1_c}(\mathbf{C})\mathbf{A} - R_{2_c}(\mathbf{C})\mathbf{B} + \mathbf{b}_C. \end{cases} \qquad (3.26)$$

Remark 3.1 Matrices D_A and D_B are symmetric, and weakly diagonally dominant with non-positive main diagonals and non-negative off-diagonals that contain the contributions of the self-diffusion terms. Matrices F_A and F_B are weakly diagonally dominant matrices with non-positive main diagonals and non-negative off-diagonals that carry the cross-diffusion contribution. The matrices $R_A(\mathbf{C}), R_B(\mathbf{C}), R_{1_c}(\mathbf{C})$, and $R_{2_c}(\mathbf{C})$ are diagonal matrices that contain the contribution of the reaction terms. Vectors $\mathbf{b}_A$, $\mathbf{b}_B$, and $\mathbf{b}_C$ are obtained from the boundary conditions. For $\mathbf{b}_C$, its entries are zero for all grid cells (i, j) with $j < M$, and $b_{\pi(i,j)} = \frac{D_c(0)}{\triangle x}\frac{2}{2\lambda + \triangle x} > 0$ for grid points with $j = M$. Entries of $\mathbf{b}_A$ and $\mathbf{b}_B$ are zero for all grid cells (i, j) with $j < M$ and for grid points with $j = M$ can be obtained

by (3.6), (3.10), and (3.14). These properties ensure non-negativity of the solution of the semi-discrete system, by standard invariance arguments.

3.3 Time integration: Embedded Rosenbrock–Wanner methods

We have shown in Section 2 that the solution of the PDE system (2.1)–(2.3) is separated from unity i.e., the singularity in the diffusion coefficients is not attained. This, together with the simulation experiments in [9, 10], suggests that using error controlled adaptive time-stepping methods for (3.26) should prevent the numerical solution from reaching or overshooting the singularity, which is a breakdown scenario for fixed time-step methods, such as the semi-implicit method [21]. Among the error-controlled time adaptive initial value problem solvers, we use embedded ROW methods. More specifically, we use for the bulk of our work ROS3PRL, a third-order method with four stages [25], which we will compare against similar variants below in Section 4.5.

In each time step of a ROW method several linear systems need to be solved, the matrix of which requires the Jacobian of the right-hand side of the ODE system (3.26), see also Appendix A. For our problem, these matrices are sparse and non-symmetric. To solve this linear system, we use the stabilised bi-conjugate gradient method [33]. More specifically, we use a routine from the SPARSKIT library [27], that is prepared for parallel execution using OpenMP as in [19].

4 Results

The results presented in this section can be grouped as follows: In Section 4.1, we illustrate the use of the method that we described in Sections 2 and 3 by documenting a typical simulation. In Sections 4.2 and 4.3, we provide as quantitative tests of the method a grid convergence study, and the application of the method in a special case where the solution of the cross-diffusion model can be compared against the solution of a simpler single-species model, that we solve with a previously established method for that simpler problem. Finally in Sections 4.4–4.6, we compare variants of the method, with a focus on computation times. This includes a comparison of different choices of biomass transfer functions p and q, of embedded ROW methods for time integration, and of two different approaches to compute the Jacobian that is needed for that.
Most of the simulations reported here, including all for which timing data are provided, were carried out on a Lenovo P700 ThinkStation with a single Intel Xeon E5-2667 v3 Processor (20MB Cache, 3.20 GHz, 8 cores/16 threads).

4.1 Illustrative simulation

For a first illustrative numerical simulation, we assume that the substratum is initially inoculated by two semi-spherical colonies, one of species a, the other one of species b, both with an initial volume fraction of 0.9. The colonies are arranged symmetrically around the centre of the substratum. The parameter values used are listed in Table 1. We assume that species b has higher maximum growth rate and half saturation concentration, which gives it a growth advantage over species a. Nutrients are supplied from the top boundary.
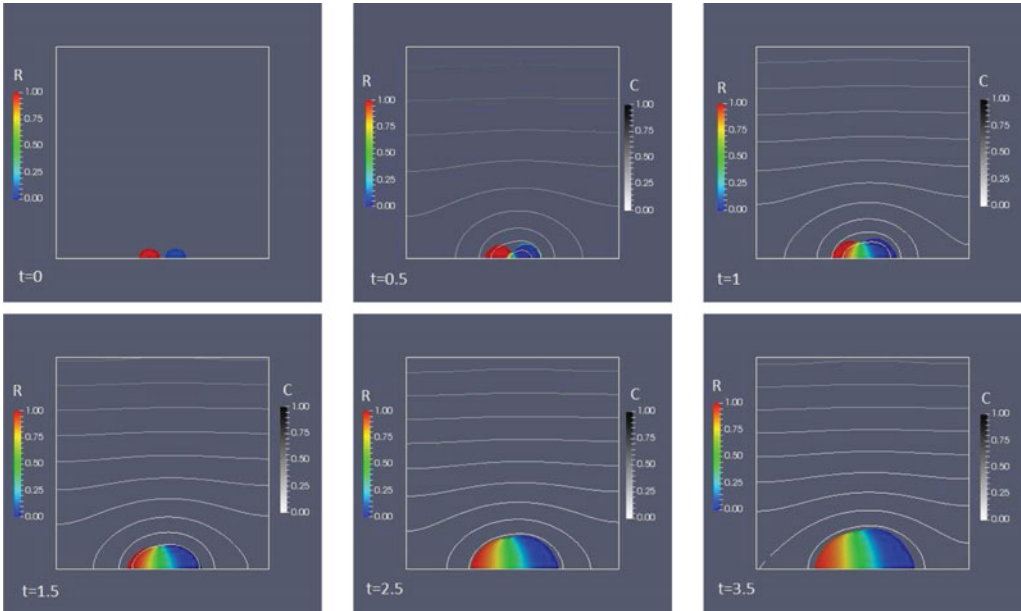
FIGURE 1. Biofilm formation and interaction between two different species of biofilm with cross-diffusion at different times. The colour coding refers to the relative fraction of the biomass of type a, $R = \frac{A}{A+B}$, and the grey-scale isolines are drawn to depict the substrate concentration C.

The computational domain is square of size $[0, 1] \times [0, 1]$, uniformly discretised by a mesh with $N \times M = 256 \times 256$ grid cells. The simulation was terminated when the final time $t = 3.5$ is reached.

Snapshots of biofilm structure at different times are shown in Figure 1. Initially, the individual colonies grow and expand, which leads to the distance between them decreasing. At $t = 0.5$, the colonies start merging and mixing of the species begins, leading to a region in the centre of the biofilm where both species are present. Species b, due to its growth advantage occupies more space than its competitor, cf. the biofilm structure at $t = 1$. The layer in which both species overlap increases, i.e., the biomass gradients in the inner regions of the colony become less steep, as the biofilm grows with time. Since both species have different growth parameters symmetry is broken, both in the biofilm structure and in the substrate concentration field. In this simulation substrate does not become severely limiting, i.e., the colony that forms after merging remains rather compact.

Whereas Figure 1 shows the structure of the biofilm colony that forms after merging and the relative distribution of biomass in the colony, we include Figure 2 to illustrate the mixing of both species in the colony, and to provide information about the biomass density levels. To this end, we plot for the same time instances as in Figure 1 the biomass volume fractions as well as the substrate concentration in the grid cells along the substratum. In the first two time instances after merging, at $t = 1, t = 1.5$, we observe that the outer layers of the colony are still occupied by the original inhabitant only. At $t = 2.5$ species b with growth advantage has completely penetrated the colony, at $t = 3.5$ also species a. Due to its growth advantage species b begins to build up faster in the overlapping
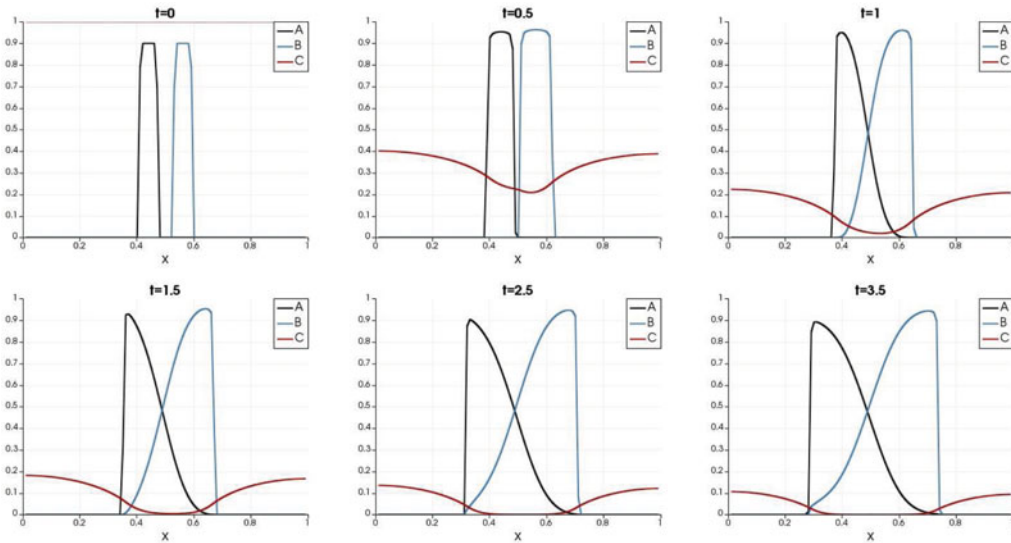
FIGURE 2. Biofilm formation and interaction between two different species including cross-diffusion effect at different times. Plotted are the biomass densities A, B, and substrate concentration C along the line $y = 0.03$.

domain. Although the substrate concentration in the inner core of the colony becomes very small, hollowing effects due to starvation are not observed, and the biomass density remains close to the maximum density throughout the simulation.

In Figure 3, we show how the time step $h(t)$ changes over time. The time-step size initially increases, indicating that the initial guess for the time-step could have been chosen larger. After a short plateau phase, it begins to decrease. Around the time at which both colonies merge the time step size oscillates briefly. Afterwards, starting from time $t \approx 0.6$ the time-step undergoes a steady increase again, although it is occasionally disrupted by sudden drops, from which the method recovers quickly. This demonstrates not only the adaptability of the numerical method, but also suggests that the time-adaptive method allows simulations with significantly larger time steps than the one required by a fixed-time step method to achieve the same accuracy.

4.2 Grid refinement

We perform a grid refinement study to show the convergence of the numerical method. For this purpose, we consider a special case of the considered model in which both biomass types have the same reaction kinetics, namely $\mu_a = \mu_b =: \mu_m = 6[d^{-1}]$, $\kappa_a = \kappa_b =: \kappa_m = 0.4[gm^{-3}]$, $K_a = K_n =: K_m = 0.4[d^{-1}]$, $\gamma_a = \gamma_b =: \gamma_m = 0.63[-]$, $a^\infty = b^\infty =: m^\infty$. The initial condition and tolerance for the ROW method are the same as those in Section 4.1. We use grids of size $N \times M$ with $N = M = 2^\kappa$, κ is an integer. We compute the least square norm of the difference between two subsequent solutions. The least square errors are defined as $E_A^\kappa = \frac{\|\mathbf{A}_\kappa - \mathbf{A}_{\kappa-1}\|_2}{2^{2(\kappa-1)}}$, $E_B^\kappa = \frac{\|\mathbf{B}_\kappa - \mathbf{B}_{\kappa-1}\|_2}{2^{2(\kappa-1)}}$, and $E_C^\kappa = \frac{\|\mathbf{C}_\kappa - \mathbf{C}_{\kappa-1}\|_2}{2^{2(\kappa-1)}}$, where subscript κ indicates the grid size. These errors which are reported in Table 2 are computed at $t = 2$. This is the time at which the gap between colonies is closed, the two biomass species

Table 2. *Results for grid refinement: Least square norms of the difference between solutions for grids with $2^\kappa \times 2^\kappa$ and $2^{\kappa-1} \times 2^{\kappa-1}$ cell resolution, $E_{A,B,C}^\kappa$, at $t = 2$*

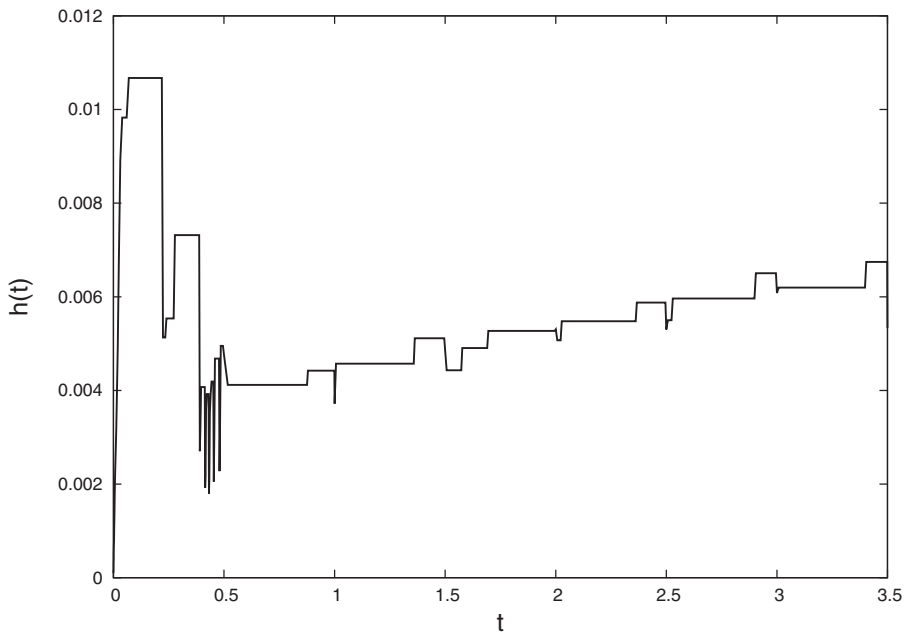| κ	E_A^κ	E_B^κ	E_C^κ
5	$0.13554882 \times 10^{-2}$	$0.13554882 \times 10^{-2}$	$0.79305360 \times 10^{-3}$
6	$0.12369284 \times 10^{-2}$	$0.12369284 \times 10^{-2}$	$0.57804700 \times 10^{-3}$
7	$0.12924492 \times 10^{-3}$	$0.12924492 \times 10^{-3}$	$0.27762389 \times 10^{-4}$
8	$0.91885915 \times 10^{-4}$	$0.91885915 \times 10^{-4}$	$0.21128131 \times 10^{-4}$
9	$0.33678108 \times 10^{-4}$	$0.33678108 \times 10^{-4}$	$0.38231383 \times 10^{-5}$

FIGURE 3. Time course of the value of time-step size.

penetrate into each other and the overlapped region is formed. The computed data shows a steady decrease in errors E_A^κ, E_B^κ, and E_C^κ as the grid is refined.

A key question for degenerate problems like the one at hand is how the method performs at interface $\Gamma(t)$, where biomass gradients blow-up. To illustrate this, we show for the grid resolutions with $N = M = 2^\kappa$, $\kappa = 4, \ldots, 9$ the profile of the biomass fractions in the grid layer $y = 0.03$ adjacent to the substratum at time $t = 2$. This is the time for which also the results in Table 2 were reported. As expected, the simulations show symmetry since the same reaction kinetics are assumed for both species. For coarse resolutions, $N = 16, 32$, substantial interface smearing occurs, which however vanishes for finer grids with $N \geqslant 128$. This suggests that the method is able to describe gradient blow-up effects. The location of the interface, and thus the size of colony, converges quickly as the grid

Table 3. *Least square norms of difference between the solution of the cross-diffusion problem,* $A + B$, *and the solution of the single-species problem* M, E^κ, *at* $t = 2$ *for different number of grid cells* $N \times N = 2^\kappa \times 2^\kappa$

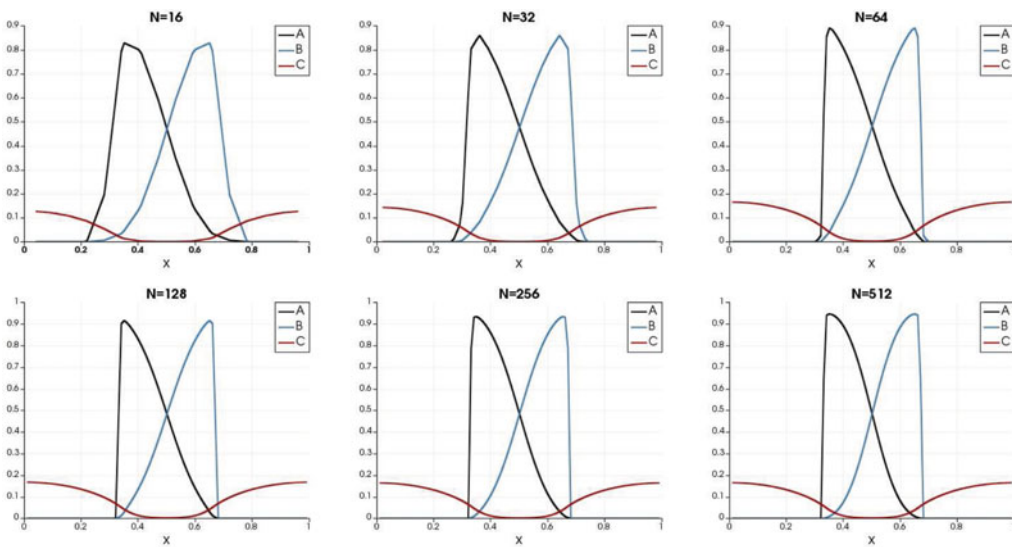| κ	E^κ
4	$0.25638711 \times 10^{-3}$
5	$0.49430582 \times 10^{-4}$
6	$0.18812184 \times 10^{-4}$
7	$0.86763930 \times 10^{-5}$
8	$0.27333315 \times 10^{-5}$
9	$0.63403576 \times 10^{-6}$

FIGURE 4. Profiles of biomass volume fractions A, B, and substrate concentration C at $t = 2$ for different grid resolutions. Plotted are the biomass densities A, B, and substrate concentration C along the line $y = 0.03$.

is refined, suggesting that a grid resolution of $N = 256$ suffices for the simulation of this problem to achieve a good trade-off between accuracy and computational time.

4.3 Comparison against a solution of the single-species model

If identical reaction parameters are specified for both species, as in Section 4.2, the total biomass $m := a + b$ is a solution of the corresponding single-species biofilm model

$$\frac{\partial m}{\partial t} = \nabla(D(m)\nabla m) + \mu_m \frac{cm}{\kappa_m + c} - K_m m,$$

$$\frac{\partial c}{\partial t} = \nabla(D_c(m)\nabla c) - \frac{\mu_m m^\infty}{\gamma_m} \frac{c}{\kappa_m + c}, \tag{4.1}$$

with $D(m) = \delta m^{\alpha}(1 - m)^{-\beta}$. In the absence of a non-trivial analytical solution to the cross-diffusion model, against which we could compare our numerical solution we use the single-species model as a test case. We compare the numerical solution of the cross-diffusion equations **A,B,C** against the numerical solution **M** of the single-species model obtained by the adaptive method in [10] which our method generalizes. We compute the least square errors between $\mathbf{A} + \mathbf{B}$ and **M** for different number of grid cells n as $E^n = \frac{\|(\mathbf{A}+\mathbf{B})-\mathbf{M}\|_2}{n^2}$. The initial condition and tolerance of the ROW method are the same as those in Section 4.1 and again the difference in solutions is reported for $t = 2$. The value of E^n for coarse grids is relatively small and decreases steadily further by refining the number of grid cells, see Table 3. This shows that the solution of the method which uses first-order upwinding for cross-diffusion terms converges to the expected solution as the grid is refined, i.e., that the discretisation errors introduced by treatment of cross-diffusion terms by simple first-order upwinding become negligible for sufficiently fine grids. In particular, the characteristic smearing by numerical diffusion is kept small around the interface.

4.4 Comparison of the newly obtained diffusion coefficients against those based on Hypergeometric Functions

The self- and cross-diffusion coefficients that we derived in Section 2 are different from those previously used in [21, 22]. We compare here the solutions obtained by both approaches. In [21, 22], the coefficients were written as

$$\begin{cases} D_{11}(a,b) = \Phi(m) + a\Psi(m), \ D_{12}(a,b) = a\Psi(m), \\ D_{21}(a,b) = b\Psi(m), \ D_{22}(a,b) = \Phi(m) + b\Psi(m), \end{cases} \tag{4.2}$$

where $\Phi(m)$ and $\Psi(m)$ are defined via the density dependent diffusion coefficient of the underlying single-species model, $D(m)$, as

$$D(m) = \delta m^{\alpha}(1 - m)^{-\beta} = \Phi(m) + m\Psi(m),$$

$$\Phi(m) = \left(1 - \int_0^m D(s)ds\right)\frac{\int_0^m D(s)ds}{m}. \tag{4.3}$$

For general exponents α and β, the integral is expressed as

$$\int_0^m D(s)ds = \frac{m^{\alpha+1}}{\alpha + 1} \, {}_2F_1(\alpha + 1, \beta; \alpha + 2, s), \tag{4.4}$$

where ${}_2F_1$ is a Hypergeometric Function. For specific values of α and β, in particular integers as in our case, these integrals can be expressed in terms of elementary functions [30], but still require in addition to polynomials and divisions the evaluations of logarithms. For comparison, we assume that the initial condition is the same as in Section 4.1. The least square difference between solutions obtained by these two methods are defined as: $E_A = \frac{\|\mathbf{A}_H - \mathbf{A}\|_2}{N^2}$, $E_B = \frac{\|\mathbf{B}_H - \mathbf{B}\|_2}{N^2}$, and $E_C = \frac{\|\mathbf{C}_H - \mathbf{C}\|_2}{N^2}$ where index H stands for using the Hypergeometric Function as in (4.3) and (4.4). The differences between the solution and a comparison of the CPU time for each method to find the solution at $t = 2$ are shown in Tables 4 and 5 for different number of grid cells. The differences between

Table 4. *Least square norms of the difference between solutions obtained by using Hypergeometric Function and the proposed scheme in Section 2 for grids with $2^\kappa \times 2^\kappa$ cell resolution, E, at $t = 2$*

κ	E_A	E_B	E_C
5	0.649999×10^{-4}	0.649998×10^{-4}	0.178283×10^{-3}
6	0.326207×10^{-4}	0.326206×10^{-4}	0.864977×10^{-4}
7	0.251905×10^{-4}	0.251905×10^{-4}	0.425207×10^{-4}
8	0.183171×10^{-4}	0.183171×10^{-4}	0.211366×10^{-4}

The tolerance of each ROW method is $TOL = 1e - 7$.

Table 5. *Elapsed CPU time of the simulation with diffusion coefficients of [22] according to (4.2), (4.3), (4.4), relative to the simulation time of the model with diffusion coefficients according to Section 2, evaluated at $t = 2$ for different grid resolutions $N \times N = 2^\kappa \times 2^\kappa$*

κ	rel. CPU time
5	1.1412
6	1.1550
7	1.2068
8	1.2882

The tolerance of the ROW method is $TOL = 1e - 7$.

the solutions are small and decrease slowly as the grid is refined. This indicates that the solutions obtained depend not strongly on the specifics of the choices of functions p and q, as long as they have the properties laid out in Section 2. However, the simulations with the self- and cross-diffusion coefficients that were derived in Section 2 appear faster than the previous coefficients of [22], by approximately 15~30% for all grid sizes.

4.5 Comparison of different ROW embeddings

In this section, we compare the numerical solution obtained by ROS3PRL with the solutions obtained by two other embedded ROW methods of the same order, namely ROS34PW2 and ROS34PRW. ROS34PW2 is a strongly A-stable method which is obtained by assuming that the ROW method satisfies the stiffly accurate condition [23]. ROS34PRW is based on a new order condition for the small local error in solving the Prothero–Robinson equation [24]. We refer to Appendix A for the Butcher tableaus of these three methods. While we expect that all three time integrators will give similar results for the problem at hand, our focus will be on the variability of compute times achieved.

For our comparisons, we assume that the initial condition is the same as in Section 4.1 and the computational domain is uniformly discretised by a mesh with $N \times M = 256 \times 256$ grid cells. For proper comparison, we set the tolerance of all methods to the same value, namely $TOL = 1e - 7$. Assuming, therefore, that all methods achieve similar accuracy,

Table 6. *Elapsed CPU time for ROS34PW2, ROS34PRW relative to ROS3PRL for solving ODE system (3.26), evaluated at t = 2*

	ROS34PW2	ROS34PRW
relative CPU time vis-a-vis ROS3PRL	1.1101	1.2927

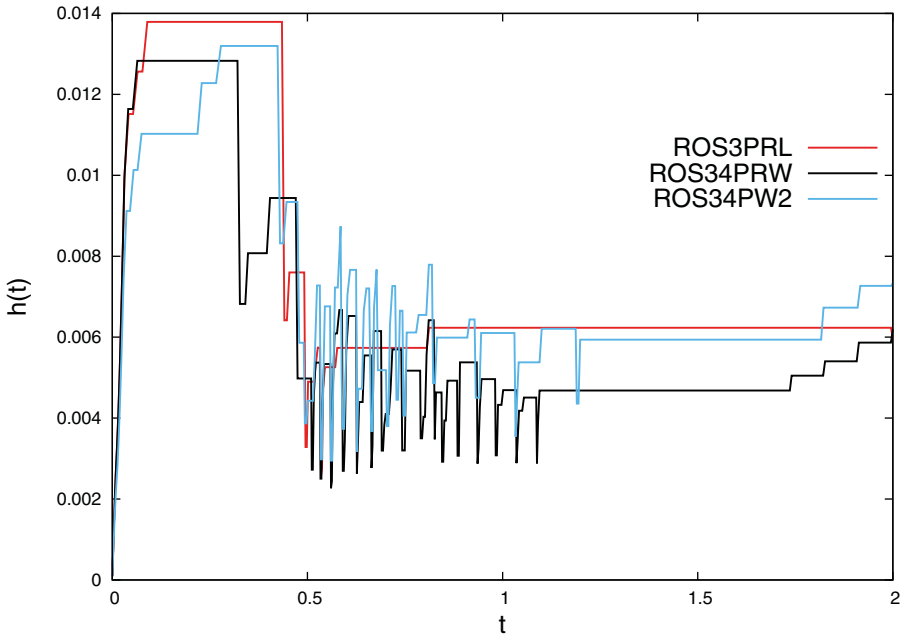The cell resolution is $N = M = 256$ and the tolerance of each ROW method is $TOL = 1e - 7$.

FIGURE 5. Time-step size, $h(t)$, for three different types of ROW method: ROS34PW2, ROS34PRW, and ROS3PRL.

the comparison criterion is the compute time required. The value of time-step size and a comparison of CPU time for these three types of ROW methods for solving ODE system (3.26) are shown in Figure 5 and Table 6. The time-step initially increases for all three methods indicating that the initial value of $h(t)$ could have been chosen larger. After a plateau phase, which is longest for ROS3PRL, at about the time that merging of colonies takes place, time-step oscillation occurs in all three methods, but the oscillation phase in ROS34PW2 and ROS34PRW last longer than for ROS3PRL. Afterwards, the time-step reaches a plateau phase at a higher level and remains on there until the end of the simulation, whereas for ROS34PW2 and ROS34PRW time-step fluctuations are observed. While the CPU time for the three methods are of the same order of magnitude, ROS3PRL is about 10% faster than ROS34PW2, which in turn is about 10% faster than ROS34PRW.

Table 7. *Least square error between solutions obtained with an analytical expression of the Jacobian (obtained using Maple) and a finite difference approximation, $E_{A,B,C}^{AN}$, for different grid sizes $2^\kappa \times 2^\kappa$ at $t = 2$*

κ	E_A^{AN}	E_B^{AN}	E_C^{AN}
5	$0.57219704 \times 10^{-8}$	$0.57513159 \times 10^{-8}$	$0.35210175 \times 10^{-8}$
6	$0.93127701 \times 10^{-8}$	$0.93106705 \times 10^{-8}$	$0.51523097 \times 10^{-8}$
7	$0.79606632 \times 10^{-8}$	$0.79604554 \times 10^{-8}$	$0.24516744 \times 10^{-8}$
8	$0.10302779 \times 10^{-7}$	$0.11174099 \times 10^{-7}$	$0.10777266 \times 10^{-8}$

For the numerical approximation, a step size is chosen relative to the value of the dependent variable i.e for derivatives with respect to a the step size is $\max\{\Delta \cdot a, 10^{-10}\}$ and similar for b and c.

4.6 Comparison of analytical Jacobian versus finite difference approximation

ROW methods need in every time step the evaluation of the Jacobian matrix of the right-hand side of the underlying ODE system. For highly non-linear systems like ours this can be cumbersome, even if computer algebra systems are used. An alternative is to approximate the Jacobian numerically, e.g., by finite differencing. The latter approach is easy to implement but can become computationally expensive.

To compare both approaches, we compute the least square norm of the difference between solutions obtained by analytical and finite difference approximations of the Jacobian matrix for various number of grid cells, and various finite difference step sizes. The initial condition in these tests is the same as in Section 4.1 and the tolerance of ROW method is set at $TOL = 1e - 7$ and results are computed at $t = 2$.

The difference between solutions obtained by analytical and numerical schemes to compute the Jacobian matrix is computed as $E_A^{AN} = \frac{\|\mathbf{A}_{ANA} - \mathbf{A}_{NUM}\|_2}{N^2}$, $E_B^{AN} = \frac{\|\mathbf{B}_{ANA} - \mathbf{B}_{NUM}\|_2}{N^2}$ and $E_C^{AN} = \frac{\|\mathbf{C}_{ANA} - \mathbf{C}_{NUM}\|_2}{N^2}$, where indices ANA and NUM stand for analytical and numerical scheme, respectively. We use for the numerical approximation of the Jacobian a step size that is chosen relative to the value of the dependent variable. Where the dependent variable vanishes, we use a small absolute step size, i.e., we have for derivatives with respect to a the step size $\max\{\Delta \cdot a, 10^{-10}\}$ and similar for b and c. We first choose the value of Δ to be 10^{-4}. Later on we will vary Δ.

For all grid resolutions the differences between the solutions obtained with analytical and numerical Jacobians are small. The quality of the approximation appears independent of the grid resolution in the sense that finer grids do not necessarily lead to better agreement, cf. Table 7. Using the numerical approximation is computationally more expensive than using the analytical expression of the Jacobian, cf Table 8. The difference in compute time increases faster than the spatial step-size. For the smallest grid with $N = 32$, the simulation with numerical approximation of the Jacobian requires twice as long as the simulation with the analytical expression. For the finer grid, the difference between both approaches is approximately factor 10.

The finite difference approximation of the Jacobian introduces an additional degree of freedom in the numerical method, namely the step-size Δ used for the approximation. To study how this parameter affects accuracy, we compute the least square norm of

Table 8. *Elapsed CPU Time of the simulation of the solution of ODE system (3.26) using a finite difference approximation of the Jacobian, relative to an analytical expression (obtained using Maple), for different numbers of grid cells $N \times N = 2^\kappa \times 2^\kappa$ at $t = 2$*

κ	rel. CPU time
5	1.9179
6	4.7335
7	10.770
8	8.7405

Table 9. *Least square error between solutions obtained by finite difference approximation of the Jacobian with different values of step-size Δ, and the ones obtained with the analytical expression, $E_{A,B,C}^\Delta$, for $N \times M = 256 \times 256$ as the grid resolution at $t = 2$*

Δ	E_A^Δ	E_B^Δ	E_C^Δ
10^{-2}	$0.33782351 \times 10^{-6}$	$0.33782618 \times 10^{-6}$	$0.96109200 \times 10^{-8}$
10^{-3}	$0.56486691 \times 10^{-7}$	$0.56482495 \times 10^{-7}$	$0.25513969 \times 10^{-8}$
10^{-4}	$0.10302779 \times 10^{-7}$	$0.10301568 \times 10^{-7}$	$0.10878556 \times 10^{-8}$
10^{-5}	$0.22194184 \times 10^{-8}$	$0.22270281 \times 10^{-8}$	$0.10424320 \times 10^{-8}$
10^{-6}	$0.29534966 \times 10^{-8}$	$0.29590839 \times 10^{-8}$	$0.99652974 \times 10^{-9}$

differences between the solutions obtained by finite differencing for various values of Δ and the one resulted by using the analytical expression. It is defined as $E_A^\Delta = \frac{\|\mathbf{A}_{ANA} - \mathbf{A}_\Delta\|_2}{N^2}$, $E_B^\Delta = \frac{\|\mathbf{B}_{ANA} - \mathbf{B}_\Delta\|_2}{N^2}$, and $E_C^\Delta = \frac{\|\mathbf{C}_{ANA} - \mathbf{C}_\Delta\|_2}{N^2}$. The error decreases steadily by decreasing Δ up to $\Delta = 10^{-5}$, but further decreases in Δ increase the error, cf Table 9. This suggests that too small step sizes for the finite difference approximation can introduce numerical artefacts, e.g., via cancellation.

5 Conclusion

The main objective of our study was to investigate the applicability of a time-adaptive, error-controlled method to solve a highly non-linear partial differential equation system that arises in biofilm modelling. This PDE system has three non-linear diffusion effects, all of which bring their own numerical challenges: (i) porous medium degeneracy as the dependent variables vanish, $m := a + b = 0$ vanishes, (ii) super-diffusion singularity as m approaches to unity, and (iii) non-linear cross-diffusion. A semi-discrete ODE approximation is obtained by spatial discretisation using a standard Finite Volume Method. Since the solutions to the PDE have low regularity, previously only low-order time integration schemes have been used, typically with fixed time steps.

The method for the cross-diffusion biofilm model generalizes an earlier method for the single-species biofilm model and inherits many of its properties with respect to (i) and (ii) above. The method is able to describe biomass gradient blow-up at the biofilm/water

interface well. It keeps interface smearing effects small, despite the low-order treatment of cross-diffusion terms by first order upwinding. This first-order upwinding, despite its proneness to numerical diffusion, was employed to ensure the preserving of non-negativity. We demonstrated practically that error controlled adaptive methods can prevent the numerical solution from overshooting the singularities of the equation. Nevertheless, a rigorous proof of the boundedness of solutions of the spatially discretized system by unity, $a + b \leqslant 1$, is an open problem. For the single-species case and for a simpler multi-species model without cross-diffusion this was previously shown in [9, 10] by regularization. We could not carry this over to the cross-diffusion problem studied here.

In our study, we compared several variants of the model and method with respect to simulation times:

- We proposed a new set of transition functions that describe the spatial behaviour of biomass. It gives the same results that one obtains with previously proposed self- and cross-diffusion coefficients, but the numerical simulations are approximately 20% faster, due to the simpler arithmetic operations involved. This suggests that from a modelling perspective the actual choice of the transition functions is not essential, as long as they satisfy certain properties that we described above.

- Among three different embedded ROW methods of the same order and with the same number of stages, we found computing time variations of up to 30%. This suggests that simulation time is relatively sensitive to the choice of time integrator, even within one class of methods.

- ROW methods require in every time step the computation of the Jacobian of the system. Obtaining those analytically, even with the help of computer algebra systems, can be cumbersome. The alternative is to approximate the Jacobian numerically, which is straightforward but can be computationally expensive, due to the non-linearities in the diffusion coefficients. While the differences in simulation results are small, computation times vary greatly. The actual difference depends on the grid size, and becomes larger as the grid is refined. For the smallest grid size tested, the difference was by a factor 2, for the largest one by a factor 10. This suggests that for simulation experiments that may consist of dozens or hundreds of simulations, the extra effort to form the Jacobian analytically, may pay off.

The numerical treatment of the biofilm cross-diffusion model that we discussed here is rather generally applicable and it should be straightforward to extend it to more involved multi-species degenerate cross-diffusion systems.

References

[1] ANAYA, V., BENDAHMANE, M., LANGLAIS, M. & SEPÚLVEDA, M. (2015) A convergent finite volume method for a model of indirectly transmitted diseases with nonlocal cross-diffusion. *J. Comput. Math. Appl.* **70**, 132–157.

[2] ANDERL, J. N., FRANKLIN, M. J. & STEWART, P. S. (2000) Role of antibiotic penetration limitation in *Klebsiella pneumoniae* biofilm resistance to ampicillin and ciprofloxacin. *Antimicrob. Agents Chemother.* **44**, 1818–1824.

[3] ANDREIANOV, B., BENDAHMANE, M. & BAIER, R. R. (2011) Analysis of a finite volume method for a cross-diffusion model in population dynamics. *Math. Models Methods Appl. Sci. (ISSN: 0218–2025)* **21**(2), 307–344, world scientific publishing.

[4] BARRETT, J. W. & BLOWEY, J. F. (2004) Finite element approximation of a nonlinear cross-diffusion population model. *Numer. Math.* **98**, 195221.

[5] COSTERTON, J. W., STEWART, P. S. & GREENBERG, E. P. (2017) Bacterial biofilms: A common cause of persistent infections. *Science* **248**(5418), 1318–1322.

[6] EBERL, H. J., PARKER, D. F. & VAN LOOSDRECHT, C. M. (2001) A new deterministic spatio-temporal continuum model for biofilm development. *J. Theor. Med.* **3**, 161–175.

[7] EBERL, H. J. & COLLINSON, S. (2009) A modelling and simulation study of siderophore mediated antagonsim in dual-species biofilms. *Theor. Biol. Med. Mod.* **6**, 30.

[8] EFENDIEV, M. A., ZELIK, S. V. & EBERL, H. J. (2009) Existence and longtime behaviour of a biofilm model. *Comm. Pure Appl. Anal.* **8**(2), 509–531.

[9] GHASEMI, M. & EBERL, H. J. (2017) Extension of a regularization based time-adaptive numerical method for a degenerate diffusion-reaction biofilm growth model to systems involving quorum sensing. *Proc. Comp. Sci.* **108**, 1893–1902.

[10] GHASEMI, M. & EBERL, H. J. (2018) Time adaptive numerical solution of a highly degenerate diffusion-reaction biofilm model based on regularisatio. *J. Sci. Comp.* **74**, 1060–1090.

[11] GALIANO, G., GARZON, M. L. & JUNGEL, A. (2001) Analysis and numerical solution of a nonlinear cross-diffusion system arising in population dynamics. *Rev. R. Acad. Cien. Ser. A Mat.* **95**, 281295.

[12] GALIANO, G., GARZON, M. L. & JUNGEL, A. (2003) Semi-discretization in time and numerical convergence of solutions of a nonlinear cross-diffusion population model. *Numer. Math.* **93**, 655673.

[13] HALL-STOODLEY, L., COSTERTON, J. W., & STOODLEY, P. (2004) Bacterial biofilms: From the natural environment to infectious diseases. *Nat. Rev. Microbiol.* **2**(2), 95–108.

[14] KHASSEHKHAN, H., EFENDIEV, M. A. & EBERL, H. J. (2008) Existence and simulations of solutions to a degenerate diffusion-reaction model of an amensalistic biofilm control system, *Discrete Cont. Dyn. Syst. B.* **12**(2), 371–388.

[15] KHASSEHKHAN, H., HILLEN, T. & EBERL, H. J. (2009). A nonlinear master equation for a degenerate diffusion model of biofilm growth, *LNCS.* **5544**, 735–744

[16] LANG, J. & TELEAGA, D. (2008) Towards a fully space-time adaptive FEM for magnetoquasistatics. *IEEE Trans. Magn.* **44**, 1238–1241.

[17] LEAR, G. & LEWIS, G. D. (2012) *Microbial Biofilms: Current Research and Applications*, Caister Academic Press, Poole, U.K. ISBN 978-1-904455-96-7.

[18] VAN LOOSDRECHT, M. C. M., HEIJNEN, J. J., EBERL, H., KREFT, J. & PICIOREANU, C. (2002) Mathematical modelling of biofilm structures. *Antonie van Leeuwenhoek* **81**(1), 245–256

[19] MUHAMMAD, N. & EBERL, H. J. (2010) OpenMP parallelization of a Mickens time-integration scheme for a mixed-culture biofilm model and its performance on multi-core and multi-processor computers. *LNCS* **5976**, 180–195.

[20] MUHAMMAD, N. & EBERL, H. J.(2011) Model parameter uncertainties in a dual-species biofilm competition model affect ecological output parameters much stronger than morphological ones. *Math. Biosci.* **233**(1), 1–18.

[21] RAHMAN, K. A. & EBERL, H. J. (2014) Numerical treatment of a cross-diffusion model of biofilm exposure to antimicrobials. *LNCS* **8384**, 134–144.

[22] RAHMAN, K. A., SUDARSAN, R. & EBERL, H. J. (2015) A mixed culture biofilm model with cross-diffusion. *Bull. Math. Biol.* **77**(11), 2086–2124.

[23] RANG, J. & ANGERMANN, L. (2005) New Rosenbrock W-methods of order 3 for partial differential algebraic equations of index 1. *BIT Numer. Math.* **45**, 761–787.

[24] RANG, J. (2014) An analysis of the Prothero-Robinson example for constructing new DIRK and ROW methods. *J. Comput. Appl. Math.* **262**, 105–114.

[25] RANG, J. (2015) Improved traditional Rosenbrock-Wanner methods for stiff ODEs and DAEs. *J. Comp. Appl. Math.* **286**, 128–144.

[26] RITTMANN, B. E. & MCCARTY, P. L. (2001) *Environmental Biotechnology: Principles and Applications*, McGraw-Hill, United States.

[27] SAAD, Y. (1994) SPARSKIT: A basic tool for sparse matrix computations. URL: http://www.users.cs.umn.edu/saad/software/SPARSKIT/sparskit.html. (accessed 2014).

[28] SCHWERMER, C. U. *et al.* (2008) Impact of nitrate on the structure and function of bacterial biofilm communities in pipelines used for injection of seawater into oil fields. *Appl. Environ. Microbiol.* **74**(9), 2841–51.

[29] SIRCA, S. & HORVAT, M. (2012) *Computational Methods for Physicists: Compendium for Students, Graduate Texts in Physics* (Berlin: Springer).

[30] SNEDDON, I. A. (1961) *Special Functions of Mathematical Physics and Chemistry*. Oliver and Boyd, Edinburgh.

[31] STEWART, P. S. & COSTERTON, J. W. (2001) Antibiotic resistance of bacteria in biofilms. *Lancet* **358**(9276), 135–8.

[32] STEWART, P. S. (2003) Diffusion in biofilms. *J. Bacteriol.* **185**, 1485–1491.

[33] VAN DER VORST, H. A. (1992) Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems. *SIAM J. Sci. Stat. Comput.* **13**(2), 631–644.

[34] WALTER, W. (1927) *Ordinary Differential Equations*, Springer, New York.

[35] WATNICK, P. & KOLTER, R. (2000) Biofilm – city of microbes (Minireview) *J. Bacteriol.* **182**(10), 2675–2679.

[36] WALTERS, E., HILLE, A., HE, M., OCHMANN, C. & HORN, H. (2009) Simultaneous nitrification/denitrification in a biofilm airlift suspension (BAS) reactor with biodegradable carrier material. *Water Res.* **43**(18), 4461–4468.

Appendix A Rosenbrock–Wanner method

Consider the initial value problem

$$\mathbf{u'}(t) = \mathbf{f}(\mathbf{u}(t)) \quad \mathbf{u}(t_0) = \mathbf{u}_0. \tag{A 1}$$

A s-stage ROW method to solve equation (A 1) is defined as [25]

$$\mathbf{k}_i = \mathbf{f}\left(t_m + \alpha_i t, \mathbf{u}^m + \tau \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{k}_j\right) + \tau \mathbf{f}_u \sum_{j=1}^{i} \gamma_{ij} \mathbf{k}_j + \tau \gamma_i \mathbf{f}_t \quad i = 1, \ldots, s \tag{A 2}$$

$$\mathbf{u}^{m+1} = \mathbf{u}^m + \tau \sum_{i=1}^{s} b_i \mathbf{k}_i, \tag{A 3}$$

where α_{ij}, γ_{ij}, and b_i are the parameters of the method, $\mathbf{f}_u = \mathbf{f}_u(t_m, \mathbf{u}_0)$ is the Jacobian matrix, $\mathbf{f}_t = \mathbf{f'}(t_m, \mathbf{u}_0)$ and $\alpha_i = \sum_{j=1}^{i-1} \alpha_{ij}$ [25]. In order to find the coefficients $\mathbf{k}_i$, $i = 1, \ldots, s$, a linear system needs to be solved in each time-step which is sparse and non-symmetric for our problem. To solve this linear system, we use the stabilised bi-conjugate gradient method [33]. More specifically, we use a routine from the SPARSKIT library [27], that is prepared for parallel execution using OpenMP as in [19].

The embedded method which is used to control the error has order $p - 1$ and gives the embedded solution as follows:

$$\hat{\mathbf{u}}^{m+1} = \mathbf{u}^m + \tau \sum_{i=1}^{s} \hat{b}_i \mathbf{k}_i, \tag{A 4}$$

Table A 1. *Set of coefficients for ROS3PRL [25]*

$\gamma = 0.43586652150845900$

$\alpha_{21} = +0.50000000000000000$	$\gamma_{21} = -0.50000000000000000$
$\alpha_{31} = +0.50000000000000000$	$\gamma_{31} = -0.79156480420464204$
$\alpha_{32} = +0.50000000000000000$	$\gamma_{32} = +0.35244216792751432$
$\alpha_{41} = +0.50000000000000000$	$\gamma_{41} = -0.49788969914518677$
$\alpha_{42} = +0.50000000000000000$	$\gamma_{42} = +0.38607515441580453$
$\alpha_{43} = +0.00000000000000000$	$\gamma_{43} = -0.32405197677907682$
$b_1 = +0.00211030085481324$	$\hat{b}_1 = +0.50000000000000000$
$b_2 = +0.88607515441580453$	$\hat{b}_2 = +0.38752422953298199$
$b_3 = -0.32405197677907682$	$\hat{b}_3 = -0.20949226315045236$
$b_4 = +0.43586652150845900$	$\hat{b}_4 = +0.32196803361747034$

Table A 2. *Set of coefficients for ROS34PW2 [23]*

$\gamma = 0.43586652150845900$

$\alpha_{21} = +0.87173304301691801$	$\gamma_{21} = -0.87173304301691801$
$\alpha_{31} = +0.84457060015369423$	$\gamma_{31} = -0.90338057013044082$
$\alpha_{32} = -0.11299064236484185$	$\gamma_{32} = +0.054180672388095326$
$\alpha_{41} = +0.00000000000000000$	$\gamma_{41} = +0.24212380706095346$
$\alpha_{42} = +0.00000000000000000$	$\gamma_{42} = -1.22325058390451470$
$\alpha_{43} = +1.00000000000000000$	$\gamma_{43} = +0.54526025533510214$
$b_1 = +0.24212380706095346$	$\hat{b}_1 = +0.37810903145819369$
$b_2 = -1.22325058390451470$	$\hat{b}_2 = -0.096042292212423178$
$b_3 = +1.54526025533510200$	$\hat{b}_3 = +0.50000000000000000$
$b_4 = +0.43586652150845900$	$\hat{b}_4 = +0.21793326075422950$

Table A 3. *Set of coefficients for ROS34PRW [24]*

$\gamma = 0.43586652150845900$

$\alpha_{21} = +0.87173304301691801$	$\gamma_{21} = -0.87173304301691801$
$\alpha_{31} = +0.14722022879435914$	$\gamma_{31} = -0.12855347382089872$
$\alpha_{32} = -0.31840250568090289$	$\gamma_{32} = +0.50507005541550687$
$\alpha_{41} = +0.81505192016694938$	$\gamma_{41} = -0.48201449182864348$
$\alpha_{42} = +0.50000000000000000$	$\gamma_{42} = +0.21793326075422950$
$\alpha_{43} = -0.31505192016694938$	$\gamma_{43} = -0.17178529043404503$
$b_1 = +0.33303742833830591$	$\hat{b}_1 = +0.25000000000000000$
$b_2 = +0.71793326075422947$	$\hat{b}_2 = +0.74276119608319180$
$b_3 = -0.48683721060099439$	$\hat{b}_3 = -0.31472922970066219$
$b_4 = +0.43586652150845900$	$\hat{b}_4 = +0.32196803361747034$

The advantage of the embedded ROW method is that it controls the time-step size to prevent unnecessary computational effort for small time-steps and less accurate results when the time-step is large. Furthermore, solving non-linear systems that occurs in other types of Runge–Kutta method can be omitted. In order to control the time-step size, the numerical error is defined as follows:

$$r = \frac{1}{N} \|\hat{\mathbf{u}} - \mathbf{u}\|_2, \tag{A 5}$$

where N is the number of grid cells. The standard controller coefficient is computed as follows:

$$r_t = \rho (\frac{TOL}{r})^{\frac{1}{p}}, \tag{A 6}$$

where $\rho \in (0, 1]$ is a safety factor, $TOL > 0$ is a given tolerance that determines the accuracy of the method and p is the order of the method. We compute the new time-step as follows:

$$\begin{cases} r \leqslant \frac{TOL}{10} & \to h_{\text{new}} = \min(r_t, 4) \times h_{\text{old}}, \\ r \geqslant TOL & \to h_{\text{new}} = \max(r_t, 0.3) \times h_{\text{old}}. \end{cases} \tag{A 7}$$

For $\frac{TOL}{10} < r < TOL$ the time-step is accepted and the ROW method continues with the previous time-step. The Butcher tableau of the three types of ROW methods considered in Section 4.5, ROS3PRL, ROS34PW2, and ROS34PRW, is shown below.