A spatially stabilized TDG based finite element framework for modeling biofilm growth with a multi-dimensional multi-species continuum biofilm model

D. Feng[1] · I. Neuweiler[1] · U. Nackenhorst[2]

Abstract We consider a model for biofilm growth in the continuum mechanics framework, where the growth of different components of biomass is governed by a time dependent advection–reaction equation. The recently developed time-discontinuous Galerkin (TDG) method combined with two different stabilization techniques, namely the Streamline Upwind Petrov Galerkin (SUPG) method and the finite increment calculus (FIC) method, are discussed as solution strategies for a multi-dimensional multi-species biofilm growth model. The biofilm interface in the model is described by a convective movement following a potential flow coupled to the reaction inside of the biofilm. Growth limiting substrates diffuse through a boundary layer on top of the biofilm interface. A rolling ball method is applied to obtain a boundary layer of constant height. We compare different measures of the numerical dissipation and dispersion of the simulation results in particular for those with non-trivial patterns. By using these measures, a comparative study of the TDG–SUPG and TDG–FIC schemes as well as sensitivity studies on the time step size, the spatial element size and temporal accuracy are presented.

✉ D. Feng
feng@hydromech.uni-hannover.de

I. Neuweiler
neuweiler@hydromech.uni-hannover.de

U. Nackenhorst
nackenhorst@ibnm.uni-hannover.de

[1] Institute of Fluid Mechanics and Environmental Physics in Civil Engineering, Leibniz Universität Hannover, Appelstraße 9a, 30167 Hannover, Germany

[2] Institute of Mechanics and Computational Mechanics, Leibniz Universität Hannover, Appelstraße 9a, 30167 Hannover, Germany

1 Introduction

Biofilms play an important role in many medical and industrial applications [8,17]. Even though the definition of biofilm is very diverse, some common components of bacterial biofilms are widely observed from experimental studies, such as the active bacteria, inert or dead bacteria and the extracellular polymeric substance (EPS) produced by the active bacteria [31]. The combination of those components may influence the profile of biofilms as well as the patterns of biofilms generated.

The fast development of the computational techniques opens a door to predict the biofilm growth process with numerical solutions of the models. Various numerical models and modeling strategies are discussed in literature. Generally, there are two main categories of biofilm models, namely the continuous models [1,2,5,7,9,13–15,27,47,48,52] and the discrete element based (DEB) models [30,33,49]. In the DEB models, a single bacterium or a mass of bacteria is modeled as an individual particle or a cell. In recent years, single species biofilm models based on particle methods are also presented in literature [45,51].

In contrast to the DEB models, continuous models are suitable to handle the problems of large scales. Continuous models differ depending on the assumptions made. Zhang et al. [52,53] developed several phase field based biofilm models by assuming the systern is a mixture of polymer biofilm networks and the solvent. They are suitable to study the mechanical behavior of the biofilm especially when the fluid structure interaction (FSI) is considered and

large deformation of biofilm happens. However, the sharp interface is difficult to be represented for phase field models even though one can always argue there is anyway no existence of real sharp interface in nature. Meanwhile, a phase field model usually ends up with solving a fouth order Cahn–Hilliard equation, which is quite computational challenging.

Another way is to assume that the biofilm grows diffusively [13–15] or advectively. We focus on a model that describes the growth of a biofilm as an advection process in this paper. Wanner and Gujer [47] developed a one-dimensional biofilm model (the W–G model) based on this assumption, and the biofilm growth rate which is limited by the concentration of the substrate is pre-defined as a Monod form of growth function. Dockery and Klapper [27] further extended the W–G model to multi-dimensional cases (the D–K model), in which the biofilm movement was modeled as a potential flow. With this model, the well-known "finger" pattern of the biofilm was reproduced. In order to describe the interactions between different components of the biofilm, Alpkvist and Klapper [2] further developed the D–K model to a multi-species form (the A–K model). In the A–K model, the biofilm movement was modeled as a potential flow driven by the reaction in the biofilm. A diffusive boundary layer of substrates with a constant thickness was assumed to be above the biofilm–fluid interface. The flow in the boundary layer was not captured by the model, but the diffusion of substrates through the boundary layer was taken into account. The growth limiting substrates diffused into the biofilm from the top of the boundary layer. Recencent reviews on different biofilm models can be found in [16,22,28,46].

In this paper, we present a solution strategy for the A–K model in the Finite Element framework. The mass balance in the A–K model is governed by an advection–reaction partial differential equation and the reaction term is generally non-linear. To predict reliable spatial distribution of biomass requires schemes that generate stable solutions without spurious oscillations and have little numerical dissipation. Such requirements hold for many other problems as well, such as modeling wound healing [44], tumor growth [50] and the generation of many other biological patterns [35]. In addition to the mass balance, as the biofilm–fluid interface changes over time, one has to deal with the moving boundary problem for solving the potential equation. Moreover, the location of the boundary layer also changes following the biofilm–fluid interface over time. This requires a special numerical treatment as well.

Alpkvist and Klapper presented a finite difference strategy to solve the model by using a third order weighted essentially nonoscillatory (WENO) scheme [26] in space together with a third order explicit TVD-Runge–Kutta scheme [19] in time to solve the hyperbolic system. The biofilm–fluid interface was set by a level-set function as the signed distance function [38]. Due to the weakness of the finite difference method on dealing with the irregular geometry of the computation domain and interface, the biofilm growth velocity on the interface was specially treated by a technique presented in [4]. Meanwhile, the symmetric ghost-node method [18] was applied to assure a second order accuracy of the potential solution on the biofilm–fluid interface. Finite element methods are applied to overcome these drawbacks. Duddu et al. [11,12] used the extended finite element (XFEM) method to obtain accurate numerical solutions of the potential as well as the substrate concentration on the irregular computation domain. The first order upwind finite difference scheme for both spatial and temporal discretization was used for solving the hyperbolic equations. The numerical scheme is stable. However, the first order finite difference scheme, especially for temporal discretization [10], for the nonlinear hyperbolic equations reduces the numerical accuracy of the whole model. Therefore, it is useful to develop a fully finite element method based numerical strategy to solve the model. As the finite element has drawbacks concerning the stability of the solution, it needs to be tested if and how reasonable solutions can be obtained in the finite element framework. This question is addressed in this paper.

In this paper, the A–K model is used for modeling two different components of biomass, namely the active biomass and the inert biomass within a biofilm. We present a new numerical strategy to simulate the biofilm growth process using the A–K model. The time discontinuous Galerkin (TDG) method [43] is applied to solve the transport equations of biomass with higher order (compared to the first order) time discretization. The instability property that comes from the hyperbolic PDEs is sufficiently controlled by Streamline Upwind Petrov Galerkin (SUPG) method [3,6] and finite incremental calculus (FIC) method [37]. The biofilm–fluid interface is captured by the level-set of a threshold value of the total biomass. The boundary layer is modeled explicitly by using a rolling ball algorithm. As analytical solutions for the non-linear coupled equations are not available, ad hoc measures for numerical dissipation and numerical dispersion are introduced and are used to compare the numerical simulation results obtained from the TDG method, TDG–SUPG and TDG–FIC methods. The influence of the time step as well as the mesh size is studied with these measures. Cauchy convergence studies of the simulation results of different time accuracy schemes are also presented.

This paper focus on the numerical study of the A–K model but not on the biofilm system behavior. A brief review of the A–K model is presented in the second section, and the numerical strategies for solving the model are presented in the third section. In Sect. 4, more detailed studies on the numerical stability of the solution strategies as well as numerical examples are presented.
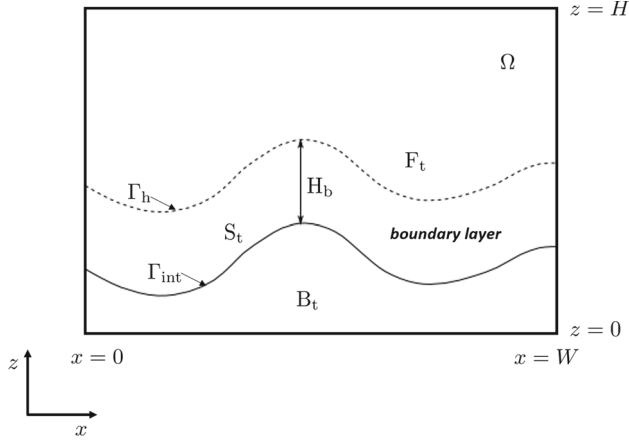
Fig. 1 Two-dimensional illustration of the computational domain Ω [2]

2 Mathematical model

The following section outlines the A–K model introduced in Section 3.1 in [2]. It is outlined at this place for completeness. For the two-dimensional A–K model, the biofilm is considered within a computational domain of $\Omega :\rightarrow \{\mathbf{x} = (x, z) : 0 \leq x \leq W, \quad 0 \leq z \leq H\}$ as illustrated in Fig. 1.

The system contains a fluid region F_t and a biofilm region B_t that are time dependent, and the biofilm–fluid interface is denoted as $\Gamma_{int} = F_t \cap B_t$. The growth of biofilm is limited by the concentration of specific substrates that are known as the growth-limiting substrates. A boundary layer with a constant thickness H_b is assumed above Γ_{int}. The growth limiting substrates are assumed to be fully mixed above the boundary layer (in the bulk fluid domain) and diffuse from the top of the boundary layer Γ_h into the biofilm. The domain below Γ_h denotes the time dependent substrates transport domain S_t.

2.1 Substrate transport

The active bacterial biomass grows by consuming the growth limiting substrate(s) that could be more than one substance interacting with each other. We assume there is only one growth limiting substrate, which is oxygen in this paper. Therefore, the concentration of the oxygen s [ML^{-3}] is one of the most vital variables in the model. The concentration of oxygen is governed by the mass balance equation

$$- D\nabla^2 s = -\upsilon_1 \bar{\rho}\frac{1}{Y}\frac{\mu s}{k_{O_2} + s} \quad \mathbf{x} \in S_t,$$
$$s = \bar{s} \quad \mathbf{x} \in \Gamma_h, \tag{1}$$
$$\frac{\partial s}{\partial \mathbf{n}_s} = 0 \quad \mathbf{x} \in \Gamma_s.$$

where D [L^{2}T^{-1}] is the diffusion coefficient of oxygen, Y [−], k_{O_2} [ML^{-3}], μ [T^{-1}] and $\bar{\rho}$ [ML^{-3}] are yield, Monod

half-saturation coefficient, maximum growth rate and the density of biofilm. The consumption of oxygen is proportional to the volume fraction of active biomass υ_1 [−] that can access the oxygen. A Dirichlet boundary of a constant oxygen concentration $\bar{s}$ is applied at the top of the boundary layer Γ_h, and no-flux boundary is applied on $\Gamma_s :\rightarrow \partial S_t \cap \partial \Omega$. $\mathbf{n}_s$ here refers to the normal vector on Γ_s.

It is noted that the transport of the oxygen is only considered as a stationary diffusion–reaction process in the model. Based on the research of Picioreanu et al. [40], the time scale of the substrate transport is much smaller than the time scale of the biofilm growth. Changing of the substrate over time happens only via the change of other variables or the boundary location, but s is assumed to be immediately at equilibrium.

2.2 Biofilm growth

In this paper, two components of biomass, namely the active biomass and the inert biomass, are considered within the biofilm. Therefore, the summation of the volume fractions of two biomass components in B_t reads

$$\upsilon_1 + \upsilon_2 = 1, \quad \mathbf{x} \in B_t. \tag{2}$$

Active biomass can be generated by consuming oxygen. Meanwhile, the active biomass will also decay with a first order rate of κ_d [T^{-1}] and transform to the inactive biomass with a first order rate of inactivation κ_i [T^{-1}]. The only source of the inert biomass in the system is from the transformation of the active biomass. The mass balance of these two components read

$$\frac{\partial \upsilon_1}{\partial t} + \nabla \cdot (\mathbf{u}\upsilon_1) = \upsilon_1 \left(\frac{\mu s}{k_{O_2} + s} - \kappa_d - \kappa_i \right),$$
$$\mathbf{x} \in \Omega, \tag{3a}$$
$$\frac{\partial \upsilon_2}{\partial t} + \nabla \cdot (\mathbf{u}\upsilon_2) = \upsilon_1 \kappa_i, \quad \mathbf{x} \in \Omega, \tag{3b}$$
$$\frac{\partial \upsilon_1}{\partial \mathbf{n}_b} = 0, \quad \frac{\partial \upsilon_2}{\partial \mathbf{n}_b} = 0, \quad \mathbf{x} \in \partial \Omega, \tag{3c}$$

where $\mathbf{n}_b$ is the normal vector of $\partial \Omega$ and the Neumann boundary is applied on the edges of the computational domain. υ_1 [−] and υ_2 [−] denote the volume fractions of the active biomass and the inert biomass respectively.

The biofilm is modeled as a fluid moving according to consumption (sink) and production (source). The growth velocity field of the biofilm $\mathbf{u}$ [LT^{-1}] is assumed to be irrotational. Therefore, there exists a potential Φ [L^{2}T^{-1}] that satisfies

$$\mathbf{u} = \nabla \Phi. \tag{4}$$

The growth of the biofilm is driven by a balance of the production and decay of the active biomass. Summing Eq. (3) by considering Eq. (2) yields the biofilm growth velocity field

$$\nabla \cdot \mathbf{u} = \nabla^2 \Phi = \upsilon_1 \left(\frac{\mu s}{k_{O_2} + s} - \kappa_d \right), \quad \mathbf{x} \in B_t. \tag{5}$$

Substituting equation (5) into (3) yields

$$\frac{\partial \upsilon_1}{\partial t} + \mathbf{u} \cdot \nabla \upsilon_1 = \upsilon_1 \left(\frac{\mu s}{k_{O_2} + s} - \kappa_d - \kappa_i \right)$$
$$- \upsilon_1^2 \left(\frac{\mu s}{k_{O_2} + s} - \kappa_d \right), \quad \mathbf{x} \in \Omega, \tag{6a}$$

$$\frac{\partial \upsilon_2}{\partial t} + \mathbf{u} \cdot \nabla \upsilon_2 = \upsilon_1 \kappa_i - \upsilon_1 \upsilon_2 \left(\frac{\mu s}{k_{O_2} + s} - \kappa_d \right),$$

$$\mathbf{x} \in \Omega, \tag{6b}$$

$$\frac{\partial \upsilon_1}{\partial \mathbf{n}_b} = 0, \quad \frac{\partial \upsilon_2}{\partial \mathbf{n}_b} = 0, \quad \mathbf{x} \in \partial \Omega. \tag{6c}$$

Equations (1–5) compose the whole A–K biofilm model used in this paper.

2.3 Dimensionless form of the governing equations

For computational convenience, the dimensionless form of the governing Eqs. (1), (5) and (6) are derived by introducing the following dimensionless variables [2]

$$\mathbf{X} = \frac{\mathbf{x}}{H}, \qquad T = \frac{t}{t_d}, \qquad \chi = \frac{t_d \Phi}{H^2},$$

$$S = \frac{s}{\bar{s}}, \qquad \Theta^2 = \frac{(H^2 \mu \bar{\rho})}{Y D \bar{s}}, \qquad \xi_1 = t_d \kappa_d, \tag{7}$$

$$\xi_2 = t_d \kappa_i, \qquad \Psi = \mu t_d, \qquad K = \frac{k_{O_2}}{\bar{s}},$$

where t_d is a typical time scale for the growth process and is here set to 24 h (86,400 s). Note that the dimensionless numbers could also be combined to yield Damköhler numbers (see Sect. 4.2). Putting all these dimensionless variables into the governing equations leads to the dimensionless form of the governing equations

$$\left. \begin{array}{l} - \nabla^2 S = -\upsilon_1 \dfrac{\Theta^2 S}{K + S}, \quad \mathbf{X} \in S_t^*, \\[2mm] S = 1, \quad \mathbf{X} \in \Gamma_h^*, \\[2mm] \dfrac{\partial S}{\partial \mathbf{n}_s^*} = 0, \quad \mathbf{X} \in \Gamma_s^*, \end{array} \right\} \tag{8a}$$

$$\left. \begin{array}{l} \nabla^2 \chi = \upsilon_1 \dfrac{\Psi S}{K + S} - \upsilon_1 \xi_1, \quad \mathbf{X} \in B_t^*, \\[2mm] \chi = 0, \quad \mathbf{X} \in \Gamma_{int}^*, \\[2mm] \dfrac{\partial \chi}{\partial \mathbf{n}_\chi^*} = 0, \quad \mathbf{X} \in \Gamma_\chi^*, \end{array} \right\} \tag{8b}$$

$$\left. \begin{array}{l} \dfrac{\partial \upsilon_1}{\partial T} + \nabla \chi \cdot \nabla \upsilon_1 = \upsilon_1 \left[\dfrac{\Psi S}{K + S} - \right. \\[2mm] (\xi_1 + \xi_2) - \upsilon_1 \left. \left(\dfrac{\Psi S}{K + S} - \xi_1 \right) \right], \\[2mm] \dfrac{\partial \upsilon_2}{\partial T} + \nabla \chi \cdot \nabla \upsilon_2 = \upsilon_1 \xi_2 - \\[2mm] \upsilon_1 \upsilon_2 \left(\dfrac{\Psi S}{K + S} - \xi_1 \right), \\[2mm] \dfrac{\partial \upsilon_i}{\partial \mathbf{n}_b^*} = 0, \quad \mathbf{X} \in \partial \Omega^*, \end{array} \right\} \tag{8c}$$

where S and χ refer to the dimensionless concentration of oxygen and dimensionless potential. S_t^*, B_t^* and Ω^* are the corresponding dimensionless domains of S_t, B_t and Ω, Γ_h^*, Γ_s^*, Γ_{int}^* and Γ_χ^* are the corresponding dimensionless boundaries of Γ_h, Γ_s, Γ_{int} and Γ_χ respectively. $\mathbf{n}_s^*$, $\mathbf{n}_\chi^*$ and $\mathbf{n}_b^*$ refer to the normal vectors of Γ_s^*, Γ_χ^* and $\partial \Omega^*$ respectively.

3 Numerical solution strategy

We present a new numerical strategy to solve the model described in the previous section. All PDEs in Eq. (8) are solved with Galerkin methods. Standard finite element method (FEM) [54] is applied to solve the Eqs. (8a) and (8b) and the spatially stabilized TDG method is applied additionally to solve Eq. (8c). As the numerical scheme for solving Eqs. (8a) and (8b) is the standard FEM scheme for solving the Poisson equation, we do not present it in detail in this paper. The first order bilinear (or trilinear) element is used for the spatial discretization of Eq. (8a) and the second order quadratic element is applied for the discretization of Eq. (8b). Solutions of second order accuracy in space and arbitrary odd order (maximum to 5th order in this paper) accuracy in time are achieved for the biofilm modeling with this fully finite element method based numerical strategy. The TDG scheme for the advection–reaction equation with vector variables is presented in Sect. 3.1 and two different stabilization methods for multidimensional problems, namely the SUPG [6] and the FIC [37] methods, are discussed in Sect. 3.2. The rolling ball algorithm is presented in Sect. 3.3 and the numerical implementation framework is summarized in Sect. 3.4.

3.1 Time-discontinuous Galerkin (TDG) method for time dependent advection–reaction equation

Time-discontinuous Galerkin (TDG) method is applied to solve the vectorized time dependent advection–reaction PDE

$$\frac{\partial \boldsymbol{\upsilon}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\upsilon} - \boldsymbol{f}(\boldsymbol{\upsilon}) = 0, \tag{9}$$
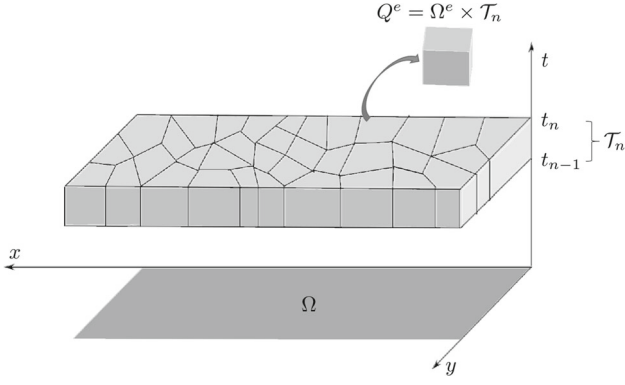
Fig. 2 TDG time–space slice and elements [23]

Fig. 3 Temporal approximation with linear shape function in TDG scheme [36]

where $\boldsymbol{v} = (v_1, v_2)^T$ is the volume fractions vector of components of biomass and the biofilm growth velocity $\mathbf{u} = -\nabla \chi$. The temporal weak form within the nth time interval $\mathcal{T}_n =]t_{n-1}, t_n[$ (as illustrated in Fig. 2) reads

$$\int_{\mathcal{T}_n} v \left(\frac{\partial \hat{\boldsymbol{v}}}{\partial t} + \mathbf{u} \cdot \nabla \hat{\boldsymbol{v}} - \boldsymbol{f}\left(\hat{\boldsymbol{v}}\right) \right) dt = 0, \tag{10}$$

where v is the temporal weight function and $\hat{\boldsymbol{v}}$ refers to the temporal approximation of the principle variables. $\boldsymbol{f}(\boldsymbol{v}) = (f_1(v_1, v_2), f_2(v_1, v_2))^T$ with

$$f_1(v_1, v_2) = v_1 \left(\frac{\Psi S}{K+S} - (\xi_1 + \xi_2) \right)$$
$$- v_1^2 \left(\frac{\Psi S}{K+S} - \xi_1 \right), \tag{11a}$$

$$f_2(v_1, v_2) = v_1 \xi_2 - v_1 v_2 \left(\frac{\Psi S}{K+S} - \xi_1 \right). \tag{11b}$$

as shown in Eq. (8c).

Both the weight function and the numerical approximation of the variables' values could be discontinuous at the temporal nodes in the TDG method. By using Green–Gauss divergence theorem, integration of Eq. (10) once yields

$$-\int_{\mathcal{T}_n} \frac{\partial v}{\partial t} \hat{\boldsymbol{v}} dt + \underbrace{v(t_n) \hat{\boldsymbol{v}}(t_n) - v(t_{n-1}) \hat{\boldsymbol{v}}(t_{n-1})}_{\text{on } \partial \mathcal{T}_n}$$
$$+ \int_{\mathcal{T}_n} v \left(\mathbf{u} \cdot \nabla \hat{\boldsymbol{v}} - f(\hat{\boldsymbol{v}}) \right) dt = 0. \tag{12}$$

Noting that $\mathcal{T}_n$ is an open time interval from t_{n-1} to t_n, the test function as well as the nodal values at t_{n-1} should use the right hand side value (+) of the interval and the left hand side values (−) for t_n. However, information can not pass from element to element and a unique solution cannot be obtained as a result of doing so. A frequently applied technique in the
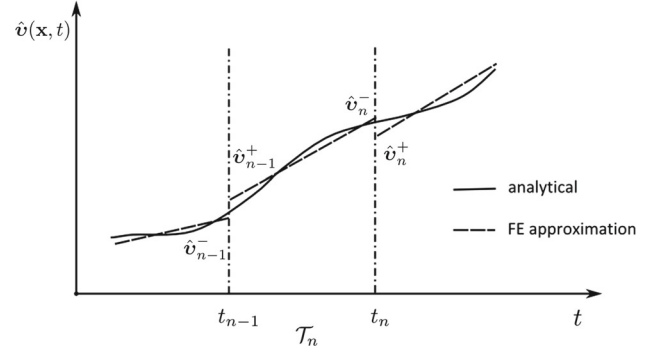
DG method is to introduce a numerical flux at the node as

$$v(t_n) \hat{\boldsymbol{v}}(t_n) - v(t_{n-1}) \hat{\boldsymbol{v}}(t_{n-1})$$
$$= v_n^- \hat{\boldsymbol{v}}_n^- - v_{n-1}^+ \left(\alpha \hat{\boldsymbol{v}}_{n-1}^- + (1-\alpha) \hat{\boldsymbol{v}}_{n-1}^+ \right), \tag{13}$$

where the parameter $\alpha \in (0, 1]$ is a constant that controls the numerical flux at the nodal point. As information always passes from the past to the future, $\alpha = 1$ when the DG method is used for time discretization (Fig. 3). Then Eq. (12) can be rewritten as

$$-\int_{\mathcal{T}_n} \frac{\partial v}{\partial t} \hat{\boldsymbol{v}} dt + v_n^- \hat{\boldsymbol{v}}_n^- - v_{n-1}^+ \hat{\boldsymbol{v}}_{n-1}^-$$
$$+ \int_{\mathcal{T}_n} v \left(\mathbf{u} \cdot \nabla \hat{\boldsymbol{v}} - f(\hat{\boldsymbol{v}}) \right) dt = 0. \tag{14}$$

Integration of Eq. (14) by using the Green–Gauss divergence theorem once again yields

$$\int_{\mathcal{T}_n} v \left(\frac{\partial \hat{\boldsymbol{v}}}{\partial t} + \mathbf{u} \cdot \nabla \hat{\boldsymbol{v}} - f(\hat{\boldsymbol{v}}) \right) dt$$
$$+ v_{n-1}^+ [\![\hat{\boldsymbol{v}}_{n-1}]\!] = 0, \tag{15}$$

in which $[\![\hat{\boldsymbol{v}}_{n-1}]\!]$ refers to the jump value of $\hat{\boldsymbol{v}}_{n-1}$

$$[\![\hat{\boldsymbol{v}}_{n-1}]\!] = \hat{\boldsymbol{v}}_{n-1}^+ - \hat{\boldsymbol{v}}_{n-1}^-, \tag{16}$$

and the discontinuous nodal values of $\hat{\boldsymbol{v}}$ are defined as

$$\hat{\boldsymbol{v}}_{n-1}^+ = \lim_{\epsilon > 0; \epsilon \to 0} \hat{\boldsymbol{v}}(t_{n-1} + \epsilon),$$
$$\hat{\boldsymbol{v}}_{n-1}^- = \lim_{\epsilon > 0; \epsilon \to 0} \hat{\boldsymbol{v}}(t_{n-1} - \epsilon). \tag{17}$$

Discretization of Eq. (15) in space and time as outlined above with the standard finite element method yields

$$\mathbf{T}_a^n \otimes \left(N^T, N\hat{v}^e \right)^e + \mathbf{T}_b^n \otimes \left[\left(-\nabla N^T \mathbf{u}, N\hat{v}^e \right)^e \right.$$
$$\left. - \left(N^T, f(N\hat{v}^e) \right)^e \right] = \mathbf{T}_c^n \otimes \left(N^T, N\hat{v}_{n-1}^{-e} \right)^e, \quad (18)$$

in which

$$(\mathrm{m}, \mathrm{n})^e = \int_{\Omega^e} \mathrm{mn} \mathrm{d}\Omega, \quad (19)$$

$\hat{v}^e$ here refers to the unknown variable vector in a single element, and N denotes the spatial shape function of the element. $\mathbf{T}_a^n$, $\mathbf{T}_b^n$ and $\mathbf{T}_c^n$ are time matrices within the nth time interval $\mathcal{T}_n$ with only temporal shape functions $\hat{N}$ involved

$$\mathbf{T}_a^n = \int_{\mathcal{T}_n} \hat{N}^T \frac{\mathrm{d}\hat{N}}{\mathrm{d}t} \mathrm{d}t + \left(\hat{N}_{n-1}^+ \right)^T \hat{N}_{n-1}^+, \quad (20a)$$

$$\mathbf{T}_b^n = \int_{\mathcal{T}_n} \hat{N}^T \hat{N} \mathrm{d}t, \quad (20b)$$

$$\mathbf{T}_c^n = \left(\hat{N}_{n-1}^+ \right)^T \hat{N}_{n-1}^-. \quad (20c)$$

Due to the nonlinear feature of the reaction term in (10), the equation is linearized by introducing an iteration process according to the Newton–Raphson method as

$$\hat{v}_{k+1} = \hat{v}_k + \Delta\hat{v}_k, \quad (21a)$$

$$f(\hat{v}_{k+1}) \approx f(\hat{v}_k) + \frac{\partial f(\hat{v}_k)}{\partial v} \Delta\hat{v}_k, \quad (21b)$$

where k is the iteration index. The linearized equation is solved iteratively and the fully discretized equation reads

$$\left[\mathbf{T}_a^n \otimes (N^T, N)^e + \mathbf{T}_b^n \otimes \right.$$
$$\left. \left((-\nabla N^T \mathbf{u}, N)^e - (N^T \frac{\partial f\left(v_k^h\right)}{\partial v}, N)^e \right) \right] \Delta v_k^h$$
$$= \mathbf{T}_c^n \otimes (v_{n-1}^{h-}, N^T)^e \quad (22)$$
$$- \mathbf{T}_a^n \otimes (N^T, N)^e v_k^h - \mathbf{T}_b^n \otimes$$
$$\left[-(-\nabla N^T \mathbf{u}, N)^e v_k^h - (f\left(v_k^h\right), N^T)^e \right],$$

where v_k^h denotes the time–space approximation of the principle variable of the kth iteration. As a remark, the global weak form is built at the level of single time–space slices (as shown in Fig. 2). The discretized Eq. (22) is always solved within a single temporal interval and marches to the next temporal interval by using the previous interval's solution of the last time point.

3.2 Spatial stabilization strategies: Streamline Upwind Petrov Galerkin (SUPG) method and finite increment calculus (FIC) method

One advantage of using the TDG method to solve the coupled advection–reaction equations is that large time steps can be applied [43]. Therefore, the time step is not limited to the most critical PDE of the coupled system. Another advantage is that higher order time schemes can be easily achieved. This is done by increasing the order of the polynomial in the temporal shape function $\hat{N}$. It has been shown that with a pth order polynomial temporal shape function, a $2p + 1$ order of temporal accuracy is achieved [24]. For instance, a linear approximation of the temporal discretization results in a 3rd order accuracy in time, and a quadratic temporal approximation leads to 5th order accuracy in time. By using higher order temporal discretization schemes, the numerical dissipation is well controlled and even the stability of the numerical scheme is improved. However, the numerical dispersion, which leads to the non-monotonic behavior of the solution, still causes problems (see later in Sect. 4.3). Therefore, using the spatially stabilized form of the TDG scheme is necessary. The stabilized form of Eq. (18) obtained by applying either the SUPG method or the FIC method can be generally written as the following equation, which introduces an additional stabilization term r^{STB}

$$\mathbf{T}_a^n \otimes \left(N^T, N\hat{v}^e \right)^e$$
$$+ \mathbf{T}_b^n \otimes \left[\left(-\nabla N^T \mathbf{u}, N\hat{v}^e \right)^e - \left(N^T, f(N\hat{v}^e) \right)^e \right.$$
$$\left. + \underbrace{r^{STB}}_{\text{stabilization term}} \right] = \mathbf{T}_c^n \otimes \left(N^T, N\hat{v}_{n-1}^{-e} \right)^e. \quad (23)$$

The stabilization term introduced by the SUPG method can be written as

$$r^{STB} = r^{SUPG} =$$
$$\left(\nabla N^T \mathbf{u}, \tau^{SUPG} \mathbf{u}^T \nabla N\hat{v}^e - f\left(N\hat{v}^e \right) \right)^e, \quad (24)$$

with the stabilization parameter $\tau^{SUPG} = \frac{h}{2\|\mathbf{u}\|}$, and h refers to the length of the longest element edge. What should be noted is that, unlike the original SUPG method, the time derivative term in r^{STB} is omitted here. Hidden behind this simplification is that the scheme is stabilized as a stationary problem in each time step. Of course, if the time derivative term is large, the simplification will lead to a different stabilized result compared to the original SUPG method. Meanwhile, the DG method ensures A-stable property for time discretization, which means the unstable problem comes from the first derivative term in space. Therefore, we have good reason to assume that the time derivative term does
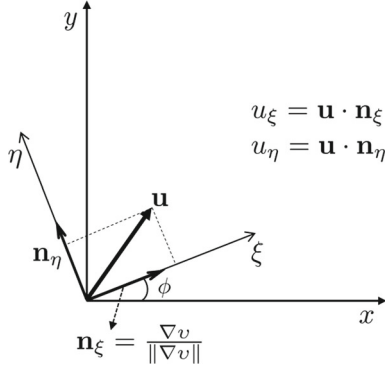
Fig. 4 The local coordinates system (ξ, η) and the global coordinates system (x, y) in two-dimensional FIC formula

Fig. 5 The local coordinates system (ξ, η, ζ) and the global coordinates system (x, y, z) in two-dimensional FIC formula

not cause instability problems in the TDG scheme. Another advantage of making such a simplification is that the spatial discretization can be fully decoupled from the temporal discretization in the stabilized TDG scheme and thus saves computational efforts.

For the FIC method, the additional stabilization term reads

$$r^{STB} = r^{FIC} = -\left(\nabla N^T D^{FIC}, \nabla N \hat{v}^e\right)^e, \tag{25}$$

and the additional balancing isotropic diffusion matrix is defined as

$$D^{FIC} = R^T \begin{pmatrix} \beta_\xi & 0 \\ 0 & \beta_\eta \end{pmatrix} R \tag{26}$$

where R is a transformation matrix from the local coordinate system to the global coordinate system, and β_ξ and β_η are the stabilization parameters along the principle directions of the solution v. The directions of the principle curvature are approximated by the gradient direction of v at the geometrical center of the element, the corresponding axes are marked as ξ in Fig. 4 and its counter orthometric direction η.

Each stabilization parameter is obtained by analyzing a one-dimensional problem, and the stabilization parameters [37] are chosen as

$$\beta_j = \gamma_j \left(\frac{\text{sign}(S_{j1})}{\text{sign}(S_{j2})}\right) + \frac{v_j}{6}\left(\frac{\text{sign}(S_0)}{\text{sign}(S_{j2})}\right), \quad (j = \xi, \eta), \tag{27a}$$

$$\gamma_j = \frac{u_j l_j}{2}, \quad v_j = a l_j^2, \tag{27b}$$

$$\text{sign}(S_0) = \text{sign}(\bar{v}) \tag{27c}$$

$$\text{sign}(S_{j1}) = \text{sign}\left(\frac{\partial \bar{v}}{\partial j}\right) \tag{27d}$$

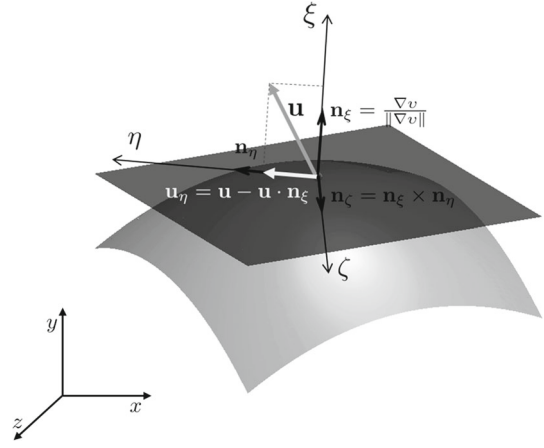$$\text{sign}(S_{j2}) = \text{sign}\left(\frac{\partial^2 \bar{v}}{\partial j^2}\right), \tag{27e}$$

where $\bar{v}$ denotes the value of v at the element center, and l_j refers to the maximum length of the projection of u_j on the element edge. In this paper, the critical stabilization parameters of $\text{sign}(S_0) = 1$, $\text{sign}(S_{j1}) = 1$ and $\text{sign}(S_{j2}) = 1$ are used. For more details we refer to [43].

The balancing diffusion matrix for three-dimensional problems in the FIC method reads

$$D^{FIC} = R^T \begin{pmatrix} \beta_\xi & 0 & 0 \\ 0 & \beta_\eta & 0 \\ 0 & 0 & \beta_\zeta \end{pmatrix} R, \tag{28}$$

and the local coordinates system is shown as in Fig. 5.

In the figure the convex surface represents the contour surface of the local solution and the flat surface on the top of the convex surface is the tangent plane at one point. The direction of axis ξ is also approximated by the gradient of the solution v at the element center. However the direction of η and ζ axes are arbitrary as long as they lie on the tangent plane. For convenience, the direction of axis η is taken the same as the projection of the velocity vector on the tangent plane $\mathbf{u}_\eta$ and the base vector along the axis is

$$\mathbf{n}_\eta = \frac{\mathbf{u}_\eta}{\|\mathbf{u}_\eta\|}, \tag{29}$$

the base vector in ζ direction is simply derived by

$$\mathbf{n}_\zeta = \mathbf{n}_\xi \times \mathbf{n}_\eta. \tag{30}$$

If the velocity vector is parallel to the gradient vector, the direction of the velocity vector rotates a constant angle along the global coordinates. What should be noted is that the artificial rotation of the velocity vector is only to determine the local coordinates, but should not change the real solution of the velocity field.
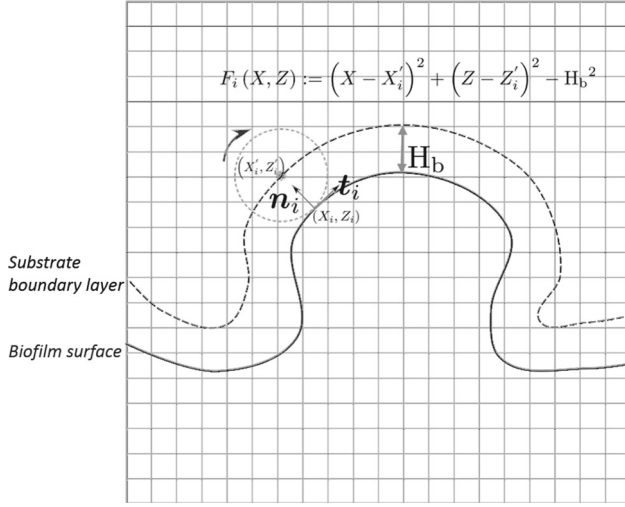
Fig. 6 Rolling ball method to determine the boundary layer in two-dimensional problems

3.3 Interface and boundary layer

The biofilm–fluid interface is determined by the contour line (or contour surface in three-dimensional problem) of the total biomass volume fraction of a constant value v^*

$$\Gamma_{\text{int}}^* :\to \text{contour}\left\{\sum_i v_i = v^*\right\}, \qquad (0 < v^* < 1). \quad (31)$$

The parameter $0 < v^* < 1$ is chosen to describe the location of the biofilm–fluid interface. However, different choices of v^* do not lead to very different results. Meanwhile, the error due to the artificial choice of this parameter will decrease with finer mesh applied.

A rolling ball algorithm is developed to determine the location of the boundary layer (as illustrated in Fig. 6). The boundary layer is determined by rolling a rigid ball with a radius of H_b on Γ_{int}^*, and the trace of the center of the ball gives the curve of Γ_h^*. A detailed description of the algorithm is listed as

- Compute the tangent vector t_i and the normal vector n_i at the point i on Γ_{int}^*.
- Compute the corresponding coordinates $X_i^{'}$ on Γ_h^* by

$$X_i^{'} = X_i + H_b n_i, \quad (32)$$

where $X = (X, Z)$ refers to the coordinates of the biofilm–fluid interface.

- Check if the ball and the biofilm–fluid interface Γ_{int}^* overlaps. A distance function is defined as

$$F_i(X) := \left(X - X_i^{'}\right)^2 + \left(Z - Z_i^{'}\right)^2 - H_b^2. \quad (33)$$
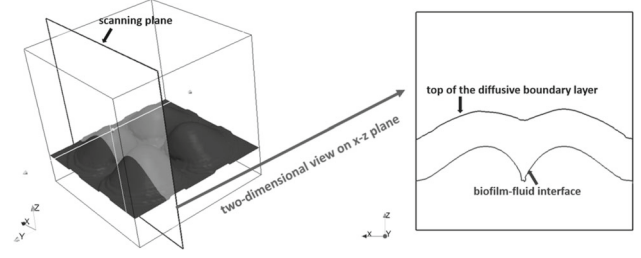

Fig. 7 Illustration of the rolling ball procedures for three-dimensional problem

For an arbitrary point i on Γ_{int}, if $F_i(X) \geq 0$ is always satisfied, point $X_i^{'}$ is marked as a point on Γ_h^*, otherwise, it is not taken as a point on Γ_h^*. An alternative form of $F_i(X)$ that can tolerate a certain amount of discrete error is used in this paper

$$F_i(X) := \left(X - X_i^{'}\right)^2 + \left(Z - Z_i^{'}\right)^2 - (H_b - \delta_i)^2, \quad (34)$$

where δ_i is a parameter related to the grid size. Here we take

$$\delta_i = \frac{1}{4}\min\{d_j\}, \quad \text{for } (j = 1, 2, 3, 4) \quad (35)$$

where d_j refers to the length of the edges of the element which contains point i (on Γ_{int}^*).

For three-dimensional problems we use a flat plane to scan the three-dimensional structure in y direction as shown in Fig. 7 and repeat the procedure in x direction. On each scanning plane a two-dimensional rolling ball procedure is carried out to obtain the line of the top of the diffusive boundary layer. With all these lines a three-dimensional surface which represents the top of the diffusive boundary layer is reconstructed.

3.4 Summary of the numerical techniques

As is shown in Sect. 2, the whole model is described by Eq. (8). Taking the two-dimensional problem as an example, for each time step, Eq. (8a) is linearized and solved with Newton–Raphson method iteratively. Four nodes bi-linear elements are used for spatial discretization. A total number of 4 integration points are used for the Gauss-Legendre integration within each element. With the solution of the oxygen concentration, Eq. (8b) is solved with the standard finite element method using 8-node quadratic elements. The reason of applying higher order elements to solve Eq.(8b) is to ensure the approximation of the gradient of the potential used in Eq. (8c) has the same numerical accuracy as the volume fractions of the biomass. The biofilm growth velocity
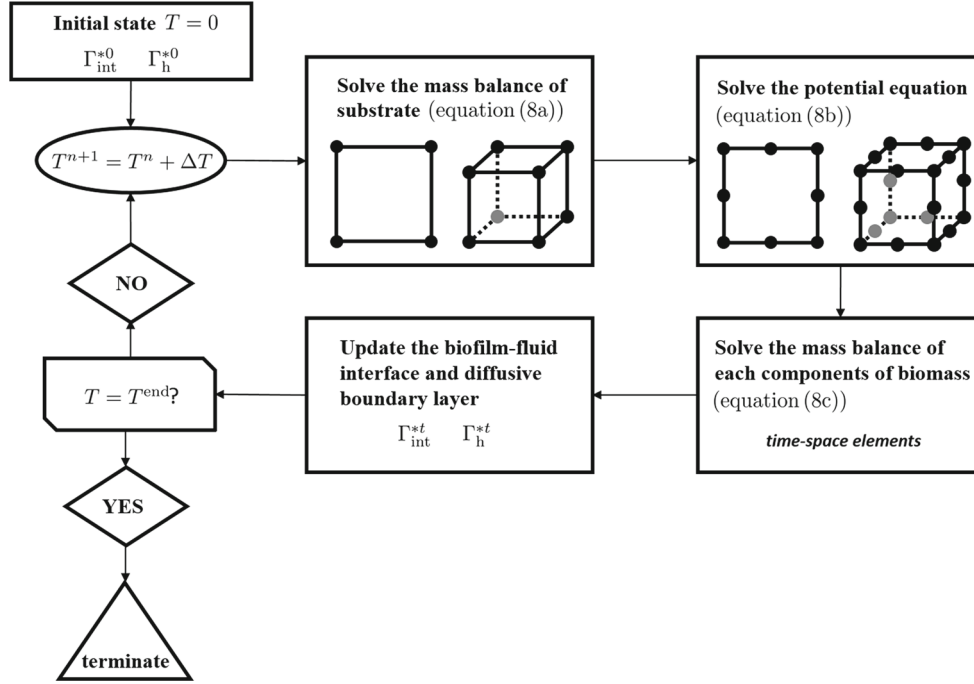
Fig. 8 Flow chart of the numerical scheme for modeling the growth of biofilm

u is evaluated on the Gauss points of the 4-node bi-linear element. With the known biofilm growth velocity **u** and the concentration of the substrate S, the new volume fractions of the active biomass and inert biomass are obtained by solving Eq. (8c). The equations are solved with the spatially stabilized methods by using 4-node bi-linear elements for spatial discretization and different order of the elements for temporal discretization. The elements we used here lead to numerical schemes of 1st, 3rd and 5th orders accuracy on the discontinuous temporal points and 2nd order accuracy in space in ideal situations. However, the introduction of the diffusion for stabilization reduces the spatial accuracy locally at the interface. Both the SUPG and the FIC methods stabilize the problem by introducing stabilization terms into the discretized equation. The SUPG method introduces the term proportional to the residuum of the discretized equation (spatial part) along the streamline. The FIC method used in this paper avoids diffusing the problem more than necessary by introducing a higher order diffusion term. In this sense, the FIC scheme performs as a higher order up-wind scheme. By the end, the new biofilm–fluid interface is captured by Eq. (31) and the rolling ball method is applied to determine the location of the diffusive boundary layer. A flowchart of the computational framework is illustrated in Fig. 8.

For three-dimensional problems, the numerical strategy is similar to the two-dimensional case discussed previously. The only difference is the choice of the spatial discretization elements. For Eqs. (8a) and (8c), 8-node bi-linear three-dimensional elements are applied for spatial discretization, while 20-node quadratic three-dimensional element are used for the discretization of Eq. (8b).

4 Numerical analysis and examples

4.1 Numerical verification: with a one-dimensional single component biofilm growth problem
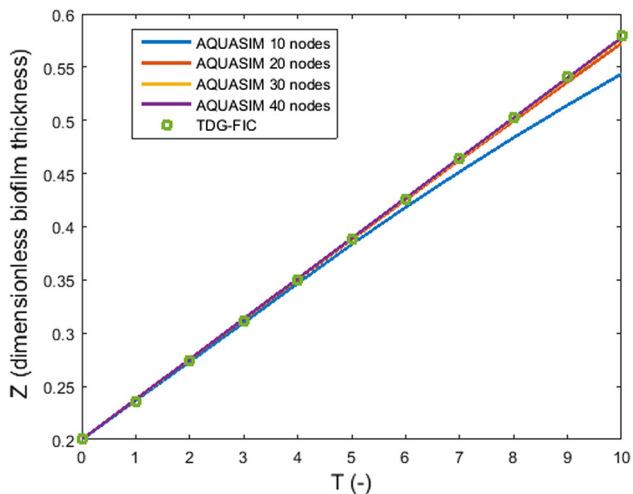
In this part, a one-dimensional biofilm growth problem is considered. We verified the numerical strategy presented in this paper as well as the code by comparing the simulation results from an open source software "AQUASIM". The governing equations are solved in a dimensionless domain $\Omega_d :\rightarrow \{\mathbf{X} = (X, Z) : 0 \leq X \leq 1, \quad 0 \leq Z \leq 1\}$ with a mesh of 200×200 elements for spatial discretization. 3rd order TDG–FIC scheme and a time step of $\Delta T = 0.1$ is applied. The initial biofilm–fluid interface is defined as

$$\Gamma_{\text{int}}^{*0} :\rightarrow Z = 0.2. \tag{36}$$

Because the initial biofilm–fluid interface is flat, this case essentially is a one-dimensional problem. Only active biomass is considered here and the decay effect is omitted. Therefore, both the inactivation rate κ_i and the decay rate κ_d are set to be zero. The modeling parameters are listed in Table 1 [2] and the dimensionless parameters used in the simulation can be calculated from Eq. (7). The same problem is simulated with "AQUASIM" [47,48] with different number of nodes to dis-

Table 1 Parameters used for two-dimensional simulation

Quantity name	Symbol	Value	Unit
Length of the computational domain	W	300	μm
Height of the computational domain	H	300	μm
Thickness of the boundary layer	H_b	45	μm
Oxygen concentration in bulk fluid	$\bar{s}$	4×10^{-3}	kg/m^3
Diffusion coefficient of oxygen	D	2×10^{-9}	m^2/s
Biofilm density	ρ	60	kg/m^3
Biofilm yield	Y	10^{-1}	$[-]$
Maximum growth rate of biofilm	μ	10^{-5}	s^{-1}
Monod half-rate constant	k_{O_2}	10^{-5}	kg/m^3

Fig. 9 One-dimensional biofilm growth simulation results with "AQUASIM" and the numerical strategy (with TDG–FIC method) presented in this paper

cretize the intial biofilm layer. The numerical setup follows the "AQUASIM" user manual [42]. As shown in Fig. 9, the "AQUASIM" results converge to the TDG–FIC result with the increase of the number of the discretization nodes.

4.2 Two-dimensional biofilm growth simulation

The governing equations are solved in a dimensionless domain $\Omega_d :\to \{\mathbf{X} = (X, Z) : 0 \leq X \leq 1, \quad 0 \leq Z \leq 1\}$ with a mesh of 100×100 elements for spatial discretization. The initial biofilm–fluid interface is defined as

$$\Gamma_{int}^{*0} :\to Z = 0.2 + 0.05 \sin\left(4\pi X + \frac{\pi}{2}\right), \tag{37}$$

and the initial volume fraction for each component of the biomass is given as $v_1^0 = 1$ and $v_2^0 = 0$ respectively, which means that only active biomass exists initially. Other parameters used in the simulation are listed in Table 1 [2] and the dimensionless parameters used in the simulation can be cal-

culated from Eq. (7). $v^* = 0.9$ is applied for capturing the biofilm–fluid interface.

Different simulation results are shown in Figs. 10 and 11. We use the 3rd order temporal accuracy TDG–FIC scheme together with a 100×100 spatial mesh for the simulations and the dimensionless time step is taken as $\Delta T = 0.1$. The dimensionless decay parameter is taken as $\xi_1 = 0.1728$ (corresponding to $\kappa_d = 2.0 \times 10^{-6}$ s^{-1}) for both results. However, different dimensionless inactivation parameters of the active biomass are chosen as $\xi_2 = 0.0864$ (corresponding to $\kappa_i = 1.0 \times 10^{-6}$ s^{-1}) and $\xi_2 = 0.5184$ (corresponding to $\kappa_i = 6.0 \times 10^{-6}$ s^{-1}) respectively.

As shown in Fig. 10, the well known finger pattern [2] is reproduced. The white curve in the figure represents the biofilm–fluid interface. Clearly, a rather sharp interface is captured with the numerical scheme presented in this paper. In column (b) of Fig. 10, the curve above the biofilm–fluid interface is the top of the diffusive boundary layer obtained with the rolling ball algorithm. The concentration of oxygen decreases continuously from the top of the boundary layer due to diffusion. But as soon as the oxygen accesses the biofilm, it is consumed by the active biomass. The results in columns (c) and (d) show that the active biomass mainly accumulates at the top of the biofilm due to a higher concentration of oxygen distribution compared to the bottom.

In Fig. 11, a flat pattern is generated when a larger dimensionless inactivation parameter $\xi_2 = 0.5184$ is taken compared with the case shown in Fig. 10. With a higher inactivation rate, less active biomass is accumulated. This leads to a thicker oxygen penetration depth (the thickness of the active layer) in the biofilm. Based on the perturbation analytical study by [27], the oxygen penetration depth plays a vital role on generating different patterns of biofilm. The results demonstrate that the active layer can distinguish the perturbation from the flat biofilm–fluid interface when the oxygen penetration depth is small (corresponding to a thin active layer). The finger pattern is generated in such situations as the thin active layer helps to magnify the perturbed interface. The thickness of the active layer is influenced by the Damköhler numbers in the model. There are two Damköh-
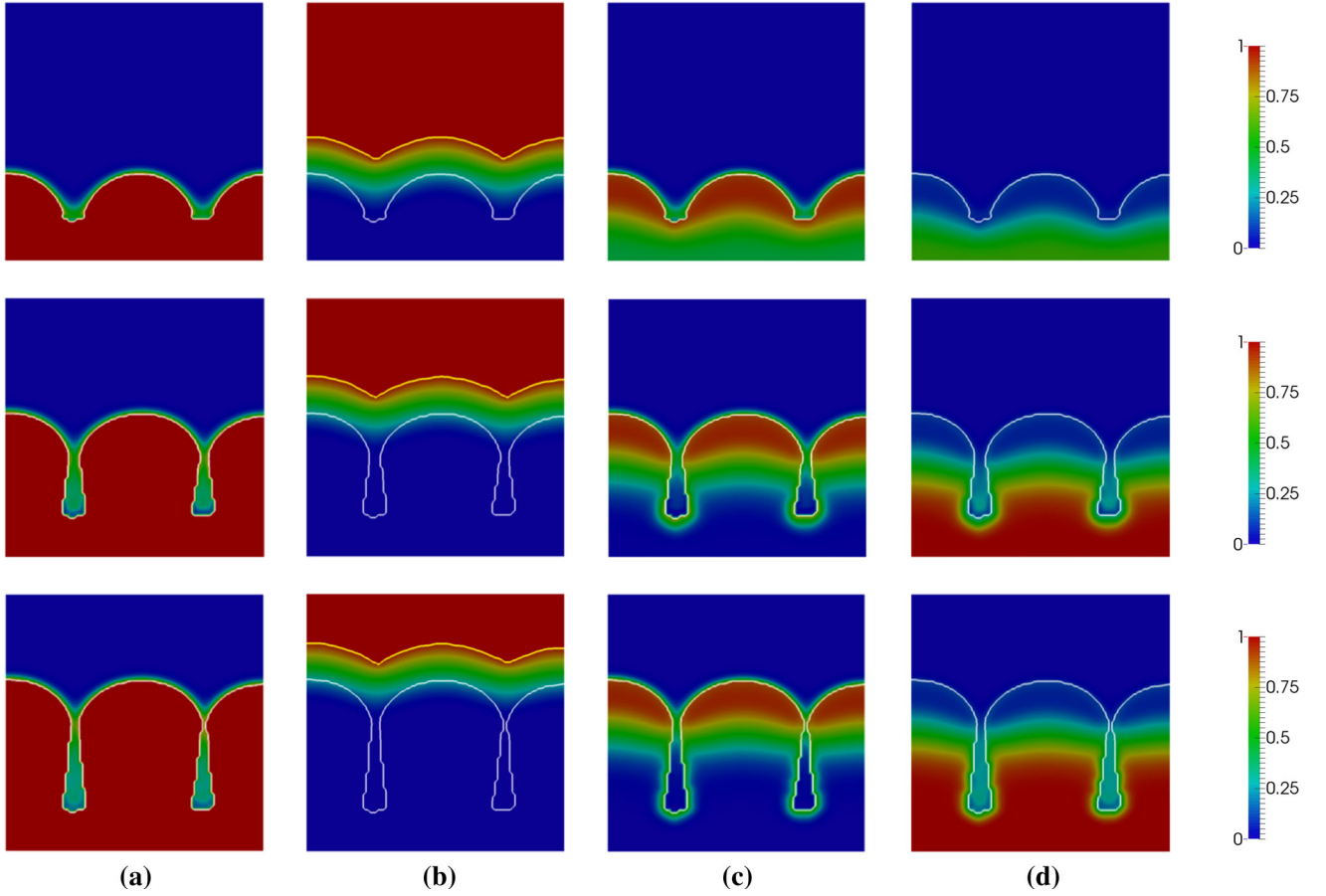
Fig. 10 Biofilm growth simulation results at $T = 5$, $T = 15$ and $T = 20$.($\xi_1 = 0.1728$, $\xi_2 = 0.0864$). **a** Total biomass distribution at different times. **b** Concentration of oxygen distribution at different times. **c** Active biomass distribution at different times. **d** Inert biomass distribution at different times

ler numbers in the A–K model corresponding to two mass transport processes, namely the mass balance of oxygen (Eq. (1)) and active biomass (Eq. (6a)). The Damköhler number of the Eq. (1) can be written as

$$Da_1 = \frac{\text{oxygen diffusion time}}{\text{oxygen consumption time}} = \frac{\frac{H^2}{D}}{\frac{1}{\mu}} \sim \Theta^2, \tag{38}$$

and the Damköhler number of the Eq. (6a) can be written as

$$Da_2 = \frac{\text{biofilm growth time (advection)}}{\text{biomass production time (reaction)}}$$
$$= \frac{t_d}{\frac{1}{\mu} - \frac{1}{\kappa_d} - \frac{1}{\kappa_i}} = \frac{1}{\frac{1}{\Psi} - \frac{1}{\xi_1} - \frac{1}{\xi_2}}. \tag{39}$$

In the simulations, only the variable ξ_2 is different, and a smaller ξ_2 leads to a larger Da_2. This means the time scale of the production of active biomass is smaller with a smaller ξ_2. Therefore, more active biomass is produced in the case

with a smaller ξ_2 for the same time of biofilm growth. Recall the right hand side of Eq. (1), clearly a thinner active layer is produced with smaller ξ_2. Therefore, the finger pattern of the biofilm is represented in the simulation results. As a remark, the magnitude of Da_1 also influences the thickness of the active layer. Similarly, a larger Da_1 leads to a thinner active layer too, and the finger pattern is more likely to be produced.

As the growth velocity of biofilm in the A–K model is described as the gradient of the potential field Φ. Different potential profiles which result in different growth velocity fields would naturally lead to different biofilm patterns. The growth velocity and the potential fields corresponding to the flat and finger patterns discussed above are shown in Fig. 12. The arrows refer to the growth velocity vectors at different points of the biofilm and the black lines denote the contours of the potential Φ. The results demonstrate that the largest magnitude of the growth velocity is at the top of the biofilm surface . This is obvious, as the oxygen concentration is larger there. Especially, in the case of the finger pattern,
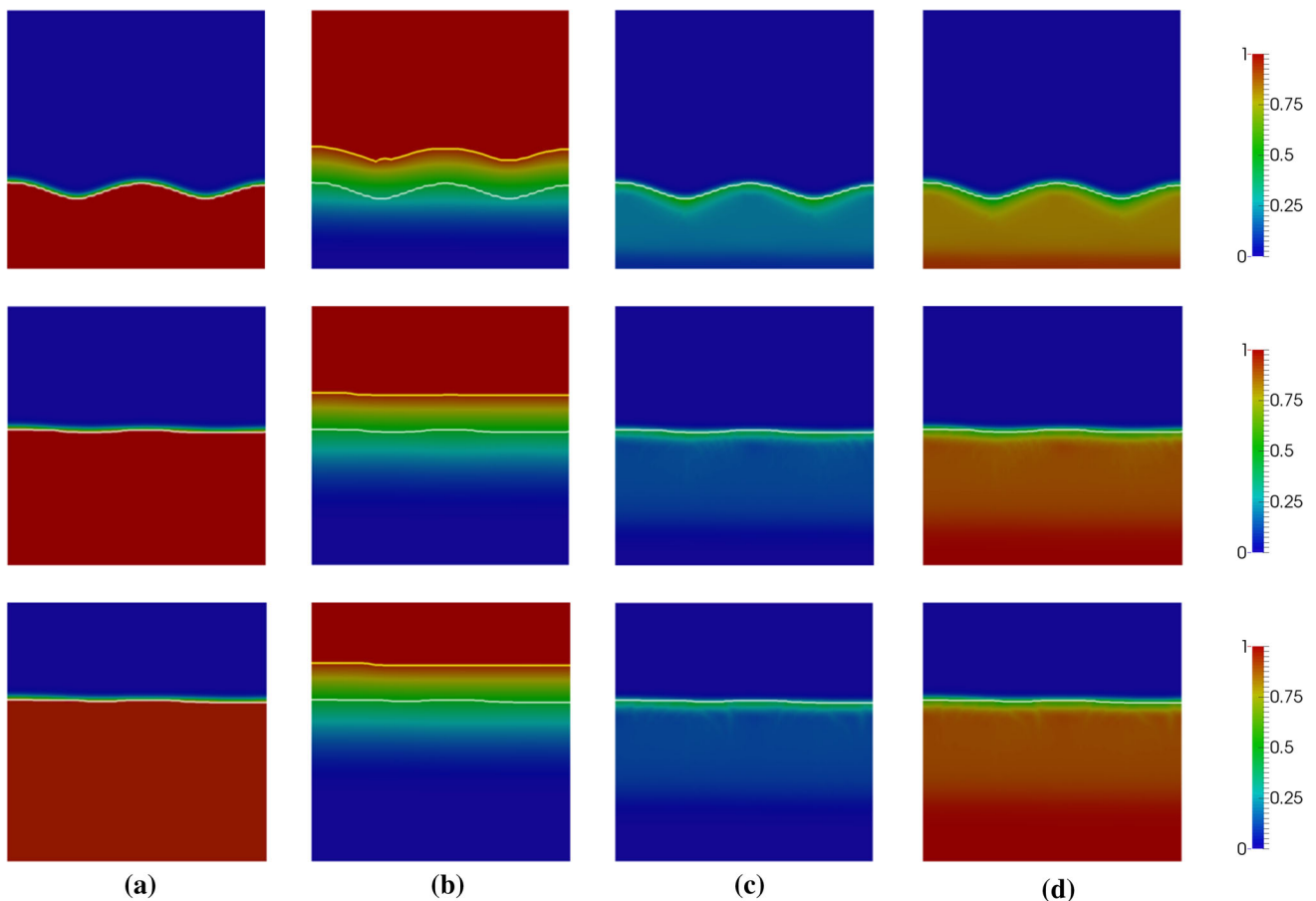
Fig. 11 Biofilm growth simulation results at $T = 5$, $T = 15$ and $T = 20.(\xi_1 = 0.1728, \xi_2 = 0.5184)$. **a** Total biomass distribution at different times. **b** Concentration of oxygen distribution at different times. **c** Active biomass distribution at different times. **d** Inert biomass distribution at different times

both positive and negative potential regions are observed and the patterns are the same as presented in [27].

4.3 Numerical dissipation and dispersion

Whenever a numerical scheme is applied to solve a PDE, numerical truncation errors by the scheme are introduced. Generally, the numerical truncation errors can be categorized as numerical dissipation, which introduces an amplitude error to the solution and numerical dispersion, which generates a phase error [20]. Numerical dissipation smears the solution and the numerical dispersion leads to a non-monotonic solution of the PDE. In this section, the behavior of the numerical dissipation as well as the numerical dispersion in the biofilm modeling strategy are studied. The modeling parameters used are the same as the ones used in the case shown in Fig. 10, as numerical dissipation and in particular dispersion have a larger impact on these types of solutions than on the flat patterns. The measures of numerical dissipation as well as the numerical dispersion are introduced in Sect. 4.3.1 and a comparative study of different stabilization strategies, namely the

SUPG and the FIC methods, is presented in Sect. 4.3.2. In Sect. 4.3.3, more results on the influence of the time–space element size on the properties of numerical dissipation and numerical dispersion are presented.

4.3.1 Measures for numerical dissipation and dispersion

Due to the lack of an analytical solution of the non-linear hyperbolic PDE with free surface involved, it is very difficult (even impossible) to carry out stability analysis in general for such PDEs system. Even though some heuristic stability analyses for some particular one-dimensional nonlinear problems are reported, it is still not solved in general for multi-dimensional non-linear PDEs [25]. A rigorous error analysis is thus not possible. For this reason, we use heuristic (ad hoc) measures for the numerical errors. The true solution of the biomass volume fractions outside of the biofilm is obviously zero. Numerical dissipation leads to a smearing out of sharp fronts. We use the value of the numerical solution outside of the biofilm as a measure for the global numerical dissipation, as these values are due to the smearing of the
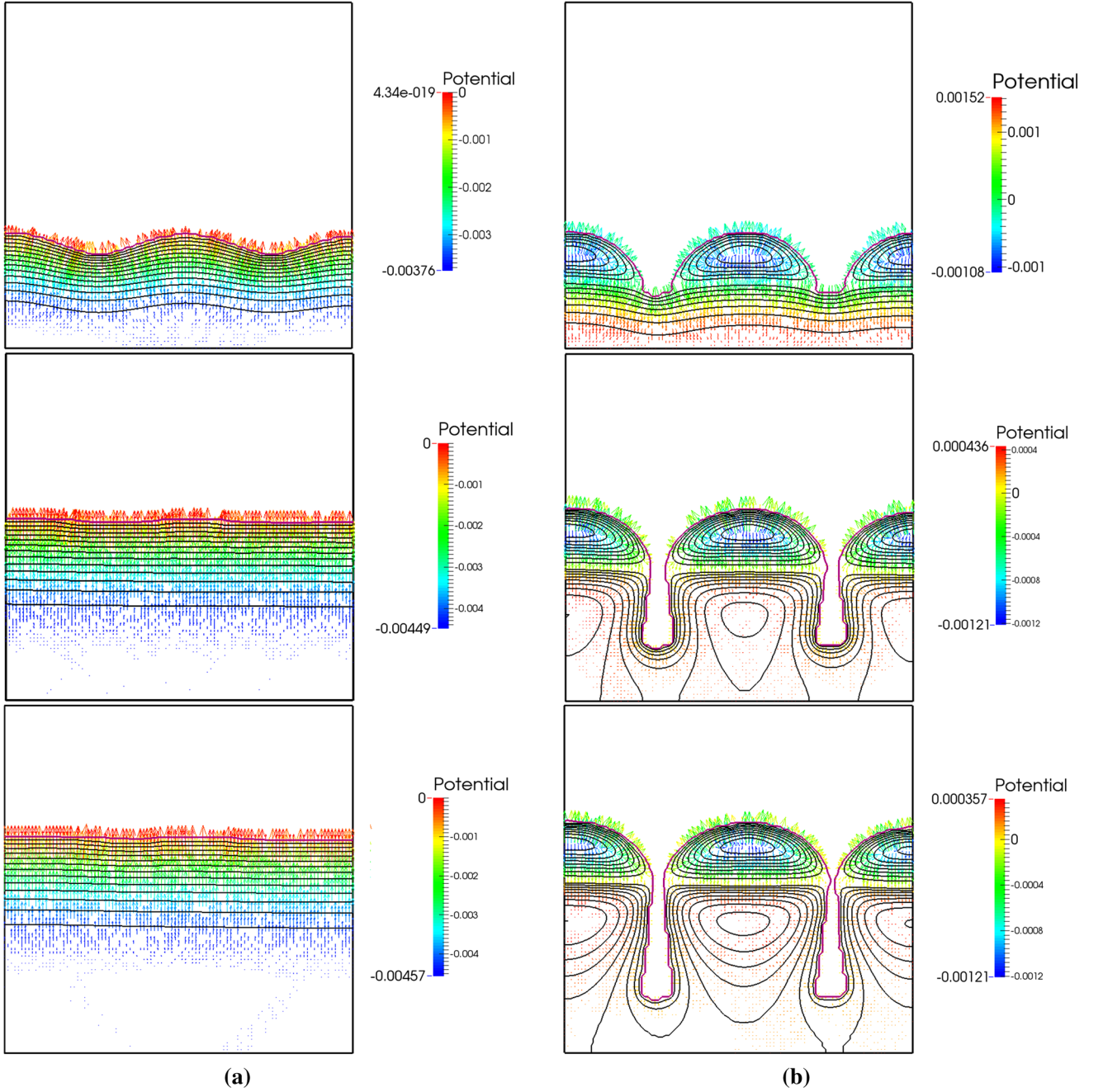
Fig. 12 Biofilm growth simulation results at $T = 5$, $T = 15$ and $T = 20$ [$\xi_1 = 0.1728$, $\xi_2 = 0.5184$ (**a**), $\xi_2 = 0.0864$ (**b**)]. **a** Potential fields and the growth velocities of the flat pattern case. **b** Potential fields and the growth velocities of the finger pattern case

sharp interface. We can calculate the global volume fraction of total biomass outside the biofilm by

$$M_{dissipation} = \sum_{i=1}^{N_n} \bar{v}_i A_i w_i^A \times 100\%, \qquad (40)$$

where N_n is the total number of nodes and A_i refers to the area surrounding node i. In this paper $A_i = A$ which is the area of a single element. w_i^A is an area weight of each

node, which is $w_i^A = 1$ inside of the computational domain, $w_i^A = 0.5$ on the edges of the domain and $w_i^A = 0.25$ on the four corner nodes. $\bar{v}_i$ refers to the total biomass volume fraction outside the biofilm of node i and $\bar{v}$ is defined as

$$\bar{v} :\to \{(\bar{v} = v_1 + v_2) \mid 0 < \bar{v} < v^*, \bar{v} \in \mathbb{R}\}. \qquad (41)$$

In order to assess the global numerical dispersion, the concept of total variation (TV) [21] which is widely used for
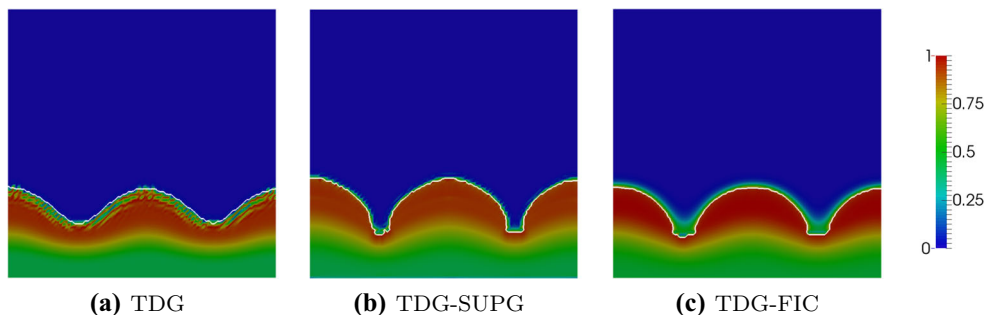
Fig. 13 Active biomass simulation results at $T = 5$ with different stabilization strategies. **a** TDG. **b** TDG–SUPG. **c** TDG–FIC

designing high resolution non-oscillatory Finite Difference schemes is adapted as measure of numerical dispersion. The volume fraction of the active biomass is chosen to study the behavior of the numerical dispersion. The TV value of the global active biomass volume fraction reads

$$TV_{active} = \sum_{i=1}^{N_n} | (v_1)_i - (v_1)_{i*} |, \tag{42}$$

where i^* refers to the neighbor nodes of the node i with a coordination number of 4. The global dispersion is measured as

$$M_{dispersion} = \frac{TV_{active}}{A_{B_t}}, \tag{43}$$

where A_{B_t} denotes the area of the biofilm in the dimensionless domain. It should be noted that TV is mostly used in order to construct numerical solutions that do not increase TV for problems where TV of the real solution is known to not increase. For general nonlinear systems, the behavior of TV of the true solution is not necessarily known, so that TV of the numerical solution does not quantify an error (as outlined in [32], pp 344–345). Because both of $M_{dissipation}$ and $M_{dispersion}$ depend on the biofilm patterns and the real solution is actually TV non-diminishing (due to the growth of finger patterns), the values $M_{dissipation}$ and $M_{dispersion}$ are only useful as qualitative measures for comparative studies (see discussion in Sect. 4.3.2). Also, both $M_{dissipation}$ and $M_{dispersion}$ presented above only dependent on spatial distributions, which means that these measures are time-varying.

4.3.2 SUPG versus FIC

The simulation results at $T = 5$ by using only the TDG method, the combined TDG–SUPG method and the TDG–FIC method are shown in Figs. 13 and 14. A quadratic temporal approximation which leads to a 5th order accuracy in time is applied together with a 100×100 spatial mesh. Non-monotonic behavior of the solution due to numerical

dispersion is observed as shown in Fig. 13a. It is also observed that the wrinkles propagate from the biofilm–fluid interface to the whole biofilm over time if only the TDG method is used. As shown in Fig. 14a, b, numerical dispersion is well controlled by using both the SUPG and FIC methods and the wrinkles are suppressed. However, the SUPG method fails to control the numerical dispersion well around the biofilm–fluid interface where essentially is a sharp interface (as shown in Fig. 14a). On the other hand, numerical dispersion is well controlled both around the biofilm–fluid interface and the domain inside of the biofilm by using the FIC method at a cost of introducing more numerical dissipation compared to the SUPG method. As the biofilm–fluid interface is defined by the total biomass volume fraction in the numerical strategy presented, numerical dispersion around the biofilm–fluid interface can lead to an over-estimation of the biofilm height in each time step. For this reason, the TDG–FIC method is more suitable for simulating the biofilm problem within our numerical framework.

The measures for numerical dissipation $M_{dissipation}$ and dispersion $M_{dispersion}$ over time are shown in Fig. 15. The TDG–FIC method introduces much more numerical dissipation than the TDG–SUPG method (as shown in Fig. 15a). The amount of dissipation for FIC is higher, but of the same order as the dissipation for the SUPG stabilization (less than 2.5% of the total biomass). It also shows that for both TDG–SUPG and TDG–FIC, $M_{dissipation}$ increases over time. This is due to the change of the biofilm pattern. An increasing length of the biofilm–fluid interface leads to an increasing value of $M_{dissipation}$. In other words, the measure for numerical dissipation presented in this paper works under the pre-condition that the biofilm patterns are comparable at the same time with different stabilization strategies. As shown in Fig. 15a, the $M_{dissipation}$ value of the TDG case is even larger than the value of the TDG–SUPG case at around $T = 18$. This is because the system of TDG becomes unstable and the biofilm pattern is not comparable with the ones of the TDG–SUPG and the TDG–FIC solutions.

We can evaluate the stability of the system from the $M_{dispersion}$ plot. It is observed that there is a big jump of

Fig. 14 Total biomass simulation results at $T = 5$ with different stabilization strategies. **a** TDG–SUPG. **b** TDG–FIC

(a) TDG-SUPG
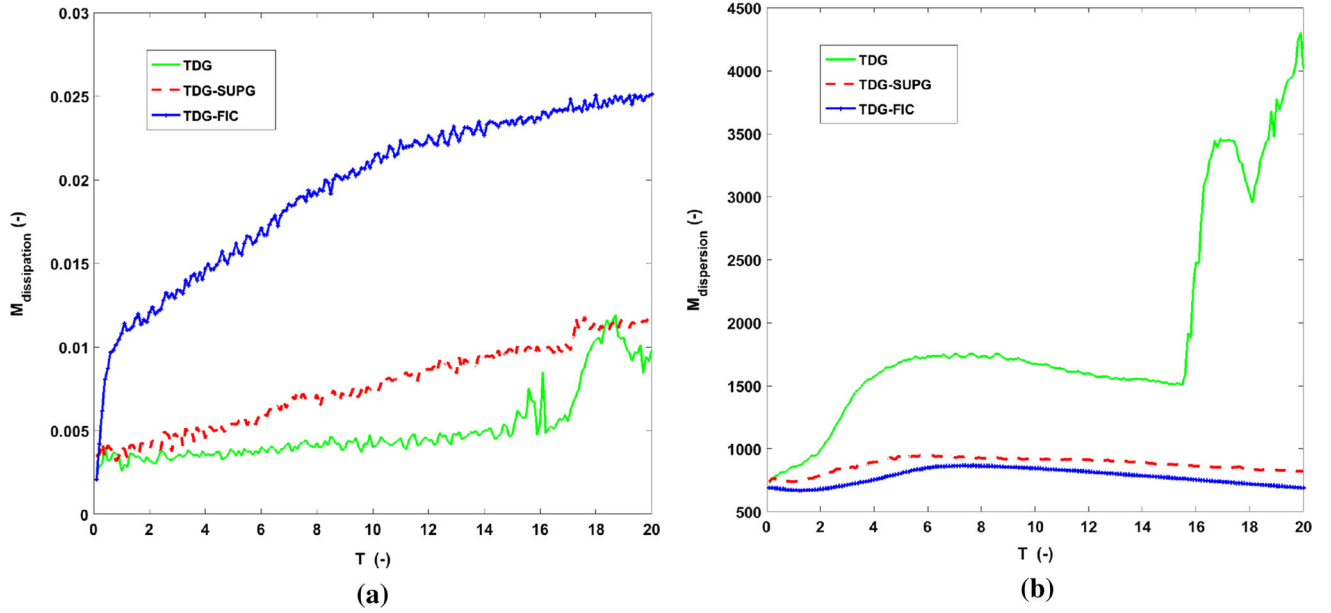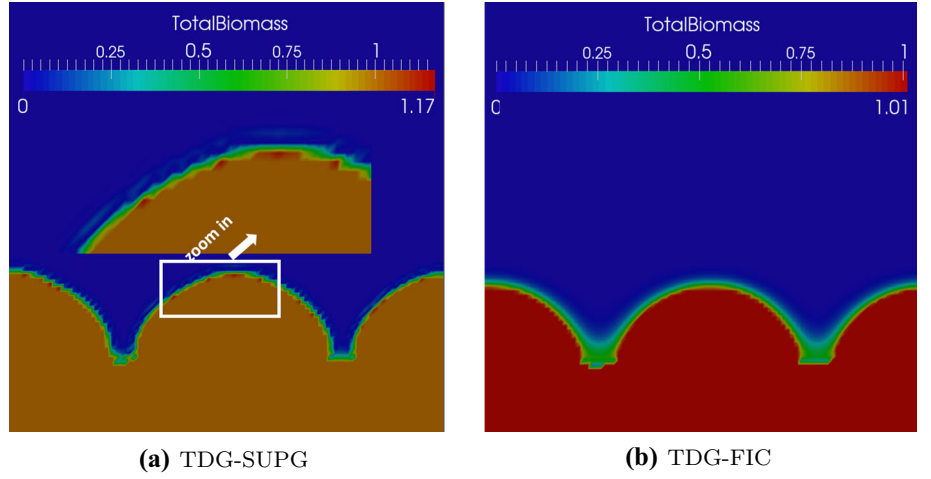
(b) TDG-FIC

(a)

(b)

Fig. 15 Measurement of the numerical dissipation and dispersion of different stabilization strategies. **a** Measurement of the numerical dissipation. **b** Measurement of the numerical dispersion

the value of the TDG curve in Fig. 15b. This is a sign of an instability of the system and the simulation result is not reliable anymore when this happens. On the contrary, the TDG–SUPG and the TDG–FIC curves are smooth which indicate that the systems are stable all the time. Moreover, it demonstrates that the numerical dispersion measured for TDG–SUPG is larger than for TDG–FIC. We can simply use the maximum value of $M_{dispersion}$ in time as a rough measure of the dispersion property of the solution

$$\bar{M}_{dispersion} = \max_{T=0}^{T^{end}} \{M_{dispersion}\}. \tag{44}$$

The values of $\bar{M}_{dispersion}$ calculated by using different stabilization strategies and temporal accuracy schemes are

listed in Table 2. The bold cells in the table denote the cases in which unstable solutions were obtained. Even the 1st order temporal accuracy TDG–FIC method is stable while with the TDG–SUPG scheme only higher time order schemes are stable (here it refers to the fifth order time scheme).

Table 2 $\bar{M}_{dispersion}$ values of the cases with different stabilization strategies and temporal accuracy TDG–FIC schemes

	TDG	TDG–SUPG	TDG–FIC
$\mathcal{O}(\Delta t)$	**2754.0**	**2208.0**	864.1
$\mathcal{O}((\Delta t)^3)$	**3322.0**	**1645.0**	863.7
$\mathcal{O}((\Delta t)^5)$	**4303.0**	947.5	864.1

Table 3 Convergence properties of TDG–SUPG method and TDG–FIC method with different orders of temporal accuracy

Iteration Index	Error $\|\Delta v\|$					
	TDG–SUPG			TDG–FIC		
	$\mathcal{O}\left(\Delta t\right)$	$\mathcal{O}\left((\Delta t)^3\right)$	$\mathcal{O}\left((\Delta t)^5\right)$	$\mathcal{O}\left(\Delta t\right)$	$\mathcal{O}\left((\Delta t)^3\right)$	$\mathcal{O}\left((\Delta t)^5\right)$
1	2.6695	3.8281	4.7879	1.2063	1.7646	2.1527
2	0.1470	0.1711	0.0487	0.0212	0.0219	0.0246
3	0.0120	0.0151	7.4154E^{-4}	2.2624E^{-4}	1.7690E^{-4}	1.9302E^{-4}
4	0.0013	0.0016	1.2062E^{-5}	3.7996E^{-6}	2.0447E^{-6}	2.2049E^{-6}
5	1.6165E^{-4}	2.0025E^{-4}	1.9821E^{-7}	6.7568E^{-8}	2.5968E^{-8}	2.8037E^{-8}
6	2.1815E^{-5}	2.5681E^{-5}	3.5861E^{-9}	1.2210E^{-9}	3.3926E^{-10}	3.6796E^{-10}
7	3.0509E^{-6}	3.3691E^{-6}	7.4782E^{-11}	2.2241E^{-11}	4.4912E^{-12}	4.8948E^{-12}
8	4.3381E^{-7}	4.4647E^{-7}				
9	6.2238E^{-8}	5.9471E^{-8}				
10	8.9808E^{-9}	7.9470E^{-9}				
11	1.3014E^{-9}	1.0644E^{-9}				
12	1.8922E^{-10}	1.4282E^{-10}				
13	2.7586E^{-11}	1.9191E^{-11}				

Furthermore, the TDG–FIC scheme shows better convergence properties than the TDG–SUPG scheme. As shown in Table 3, the Newton–Raphson iteration of Eq. (8c) converges within 7 steps by even using the 1st order temporal accuracy TDG–FIC scheme. For the TDG–SUPG scheme, only the 5th order temporal accuracy scheme converges after the 7th iteration and the lower order schemes converge after the 13th iteration. All these aspects shown above indicate that the TDG–FIC scheme is more suitable than the TDG–SUPG scheme for the hyperbolic equations in the A–K model. As a remark, it seems that the TDG–FIC does not benefit from the higher order time scheme as shown in Table 2. However, by increasing the temporal accuracy, the solution obtained from the TDG–FIC scheme is also improved, especially when large time steps or coarse meshes are used (as shown in Table 6). Of course, as a cost of increasing the temporal accuracy more computational effort is required.

4.3.3 Influence of the spatial and temporal mesh size

Another important factor that influences the properties of numerical dissipation and dispersion is the size of the time-space elements used in the simulation. As shown in the previous section, the TDG–FIC method is better suited than the TDG–SUPG method for this application. Here we use the TDG–FIC method to study the influence of the time step size and spatial element size for the biofilm simulation. The simulation results by using different spatial mesh sizes are shown in Fig. 16. The 5th order temporal accuracy TDG–FIC scheme with a dimensionless time step of $\Delta T = 0.1$ has been applied for these cases. Clearly, with finer mesh less numeri-

cal dissipation is observed. Similarly to Eq. (44) we can also measure the dissipation profile by taking the maximum value of $M_{dissipation}$ in time, i.e.

$$\bar{M}_{dissipation} = \max_{T=0}^{T^{\mathrm{end}}}\{M_{dissipation}\}. \tag{45}$$

The values of $\bar{M}_{dissipation}$ shown in Table 4 are in agreement with the results shown in Fig. 16. Obviously, more numerical dissipation will be introduced if a coarser mesh is used. Too much numerical dissipation may even ruin the simulation results (as shown in the case of 25×25 mesh in Fig. 16). It should be noted that the size of the element in space and time also influences the property of the numerical dispersion.

We compare the numerical results of different time steps used in Fig. 17. The 100×100 spatial mesh is used here in order to reduce the influence of the numerical dissipation introduced by using coarse meshes.

As shown in Fig. 17, all the results by using time steps of $\Delta T = 0.05$ and $\Delta T = 0.1$ show smooth and sharp biofilm–fluid interfaces and the solutions converge when $\Delta T \leq 0.1$. By using a larger time step of $\Delta T = 0.2$, only the higher order time schemes ($\mathcal{O}\left((\Delta t)^3\right)$ and $\mathcal{O}\left((\Delta t)^5\right)$) provide good patterns. However, local "over-shoot" features due to the numerical dispersion are observed in those cases. The values $\bar{M}_{dispersion}$ and $\bar{M}_{dissipation}$ observed by using different time steps and temporal accuracy schemes are presented in Table 5 and Table 6. There is no extreme large value of $\bar{M}_{dispersion}$ in Table 5, which again demonstrates that the TDG–FIC method can control the numerical dispersion well. For the cases of $\Delta T = 0.05$ and $\Delta T = 0.1$
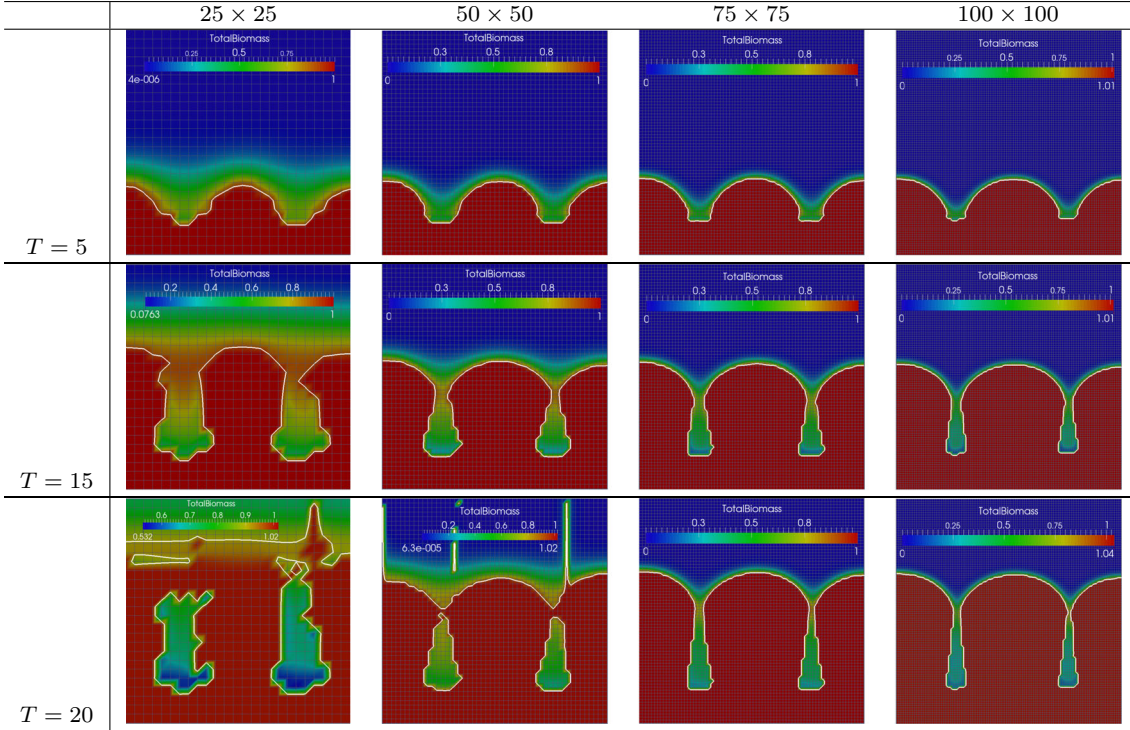
Fig. 16 Biofilm growth simulation results at $T = 5$, $T = 15$ and $T = 20$ by using different spatial mesh size and TDG–FIC schemes (with different time accuracy)

Table 4 $\bar{M}_{dissipation}$ values of the cases with different stabilization stretagies and temporal accuracy TDG–FIC schemes

	25×25 (%)	50×50 (%)	75×75 (%)	100×100 (%)
$\mathcal{O}(\Delta t)$	32.81	10.21	3.98	2.51
$\mathcal{O}\big((\Delta t)^3\big)$	29.79	10.21	4.00	2.52
$\mathcal{O}\big((\Delta t)^5\big)$	29.79	10.21	3.99	2.51

(converged results), more numerical dispersion results when a smaller time step is applied. This is because the numerical dissipation is smaller with smaller time steps and the interface is sharper as a result. This leads to a larger value of TV of the solution itself and hence to a larger measure of the dispersion is obtained. Meanwhile, slightly more numerical dissipation is measured in the case $\Delta T = 0.05$ than in the case $\Delta T = 0.1$ as shown in Table 6. With a closer look at the simulation results shown in Fig. 17 one can find the gaps between two fingers of each case is different, the gap is bigger in the case of $\Delta T = 0.05$ than in the case $\Delta T = 0.1$ while the fingers are even connected in some other larger time step cases. A slightly larger gap could lead to a slightly larger measure of dissipation naturally, but as the patterns are still comparable, the difference is not too big.

However, for the cases with large time steps i.e. $\Delta T = 0.2$ and $\Delta T = 0.5$, more numerical dissipation (as shown in the bold cells in Table 6) is observed. The biofilm–fluid interface can hardly be captured accurately while too much numerical dissipation is introduced to the system and this error could accumulate over time which results in an unstable solution. As a remark, it is always subjective to pick critical values of $\bar{M}_{dispersion}$ and $\bar{M}_{dissipation}$ to judge if the simulation result is acceptable or not. To the authors' opinion, good predictions of the biofilm patterns as well as the amount of each component of biomass are the most important issues.

4.4 Convergence analysis

A Cauchy Convergence analysis of the simulation results with the TDG–FIC method of different time order schemes is carried out in this section. For the convergence analysis over different time step sizes, the grid size is taken as $\Delta X = 0.01$ and the error norms of the solution are calculate by using the solution of the total biomass vectors as [34]

$$\epsilon_{total}^{(l)} = \|\frac{\boldsymbol{v}^{(l)} - \boldsymbol{v}^{(l-1)}}{n_{\upsilon}}\|, \quad (l = 1, 2, 3). \tag{46}$$
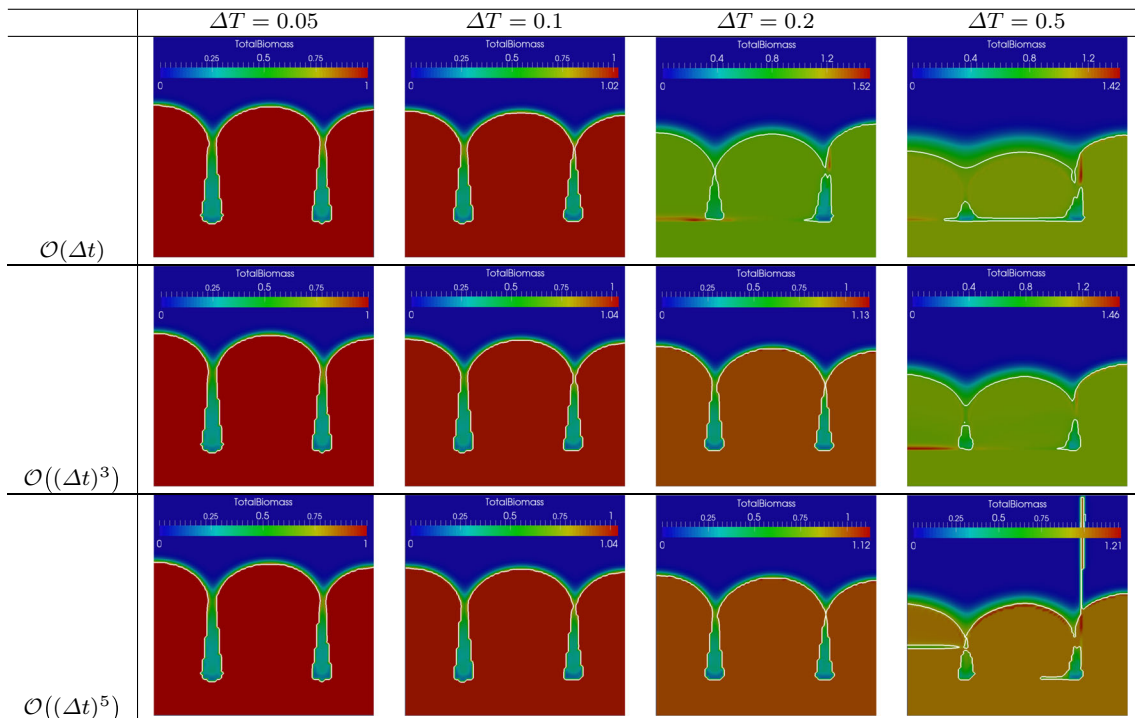
Fig. 17 Biofilm growth simulation results at $T = 20$ by using different time steps and TDG–FIC schemes (with different time accuracy)

Table 5 $\bar{M}_{dispersion}$ values of the cases with different time steps and temporal accuracy TDG–FIC schemes

	$\Delta T = 0.05$	$\Delta T = 0.1$	$\Delta T = 0.2$	$\Delta T = 0.5$
$\mathcal{O}(\Delta t)$	879.6	864,1	862.7	893.2
$\mathcal{O}((\Delta t)^3)$	880.7	863.7	873.8	907.2
$\mathcal{O}((\Delta t)^5)$	880.7	864.1	874.6	930.2

where $\boldsymbol{v}^{(l)}$ is the lth total biomass solution of the data set and $n_v = 10201$ is the length of the solution vector. $\boldsymbol{v}^{(0)}$ refers to the total biomass solution computed by using a time step of $\Delta T^{(0)} = 0.025$ and $\Delta T^{(l)} = 2\Delta T^{(l-1)}$. A loglog plot of $\epsilon_{total}^{(l)}$ over $\Delta T^{(l-1)}$ is shown as in Fig. 18 and the slope of the plot indicates the order of convergence over time discretization.

Due to the sizes of the solution vectors change over different spatial mesh sizes and the problem is non-linear, it is not straightforward to use Eq. (46) to calculate the errors [29]. We use the total biomass in the system $\bar{\upsilon}_\Omega$, which is a scalar,

for the convergence analysis with different spatial mesh sizes ΔX. The errors, similar to Eq. (46), are calculated as

$$\bar{\epsilon}_{total}^{(l)} = |\bar{\upsilon}_\Omega^{(l)} - \bar{\upsilon}_\Omega^{(l-1)}|, \quad (l = 1, 2, 3). \tag{47}$$

where $\bar{\upsilon}_\Omega^{(l)}$ is the lth total biomass volume fraction in the domain of the data set. $\bar{\upsilon}_\Omega^{(0)}$ refers to the total biomass computed by using a 200×200 mesh ($\Delta X^{(0)} = 0.005$)and $\Delta X^{(l)} = 2\Delta X^{(l-1)}$. The time step is chosen as $\Delta T = 0.05$ and the grid convergence profile is shown in Fig. 19.

The results demonstrate that all the tests converge by decreasing the time step size and mesh size, which indicates that the numerical strategy presented in the paper provides converging results. The results also demonstrate that the convergence property is not improved by using more than 3rd order time scheme. As a remark, the measures for numerical dissipation as well as for numerical dispersion presented in previous sections also indicate similar converging properties (see Tables 4, 5, 6).

Table 6 $\bar{M}_{dissipation}$ values of the cases with different time steps and temporal accuracy TDG–FIC schemes

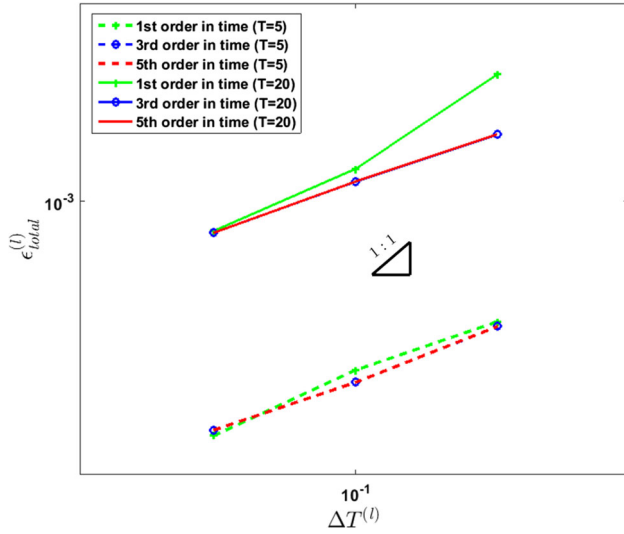|	$\Delta T = 0.05$ (%)	$\Delta T = 0.1$ (%)	$\Delta T = 0.2$ (%)	$\Delta T = 0.5$ (%)
$\mathcal{O}(\Delta t)$	2.73	2.51	**4.35**	**6.44**
$\mathcal{O}((\Delta t)^3)$	2.71	2.52	2.38	**4.03**
$\mathcal{O}((\Delta t)^5)$	2.71	2.51	2.37	**4.17**

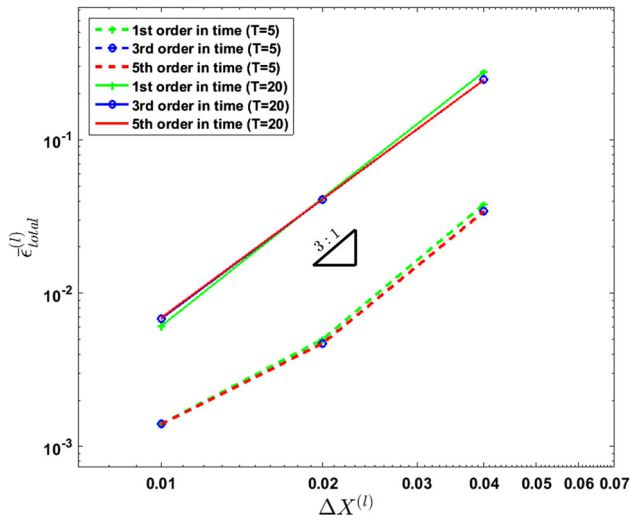**Fig. 18** Convergence of total biomass over different time steps at $T = 5$ and $T = 20$

Fig. 19 Convergence of total biomass over different mesh sizes at $T = 5$ and $T = 20$

4.5 Three-dimensional biofilm growth simulation

The numerical strategy presented in this paper can be extended to model three-dimensional biofilm growth. The governing equations are also solved in the dimensionless form within a dimensionless computation domain $\Omega_d :\rightarrow$ $\{\mathbf{X} = (X, Y, Z) : 0 \leq X \leq 1, \ 0 \leq Y \leq 1, \ 0 \leq Z \leq 1\}$. A mesh with $70 \times 70 \times 70$ cube elements is used for the spatial discretization. The third order time accuracy TDG–FIC scheme is applied here. The parameters used for the three-dimensional simulation are listed in Table 7. As an example, we take the initial biofilm–fluid interface (Fig. 20) as

$$\Gamma_{\text{int}}^0 :\rightarrow Z = 0.2 + | 4X - 2 || 4Y - 2 | e^{-(4X-2)^2-(4Y-2)^2}. \tag{48}$$

Table 7 Parameters used for three-dimensional simulation

| Quantity name	Symbol	Value	Unit
Lengths of the computational domain	$W1, W2$	300	μm
Height of the computational domain	H	300	μm
Thickness of the boundary layer	H_b	45	μm
Oxygen concentration in bulk fluid	$\bar{s}$	4×10^{-3}	kg/m^3
Diffusion coefficient of oxygen	D	2×10^{-9}	m^2/s
Biofilm density	ρ	60	kg/m^3
Biofilm yield	Y	10^{-1}	$[-]$
Maximum growth rate of biofilm	μ	10^{-5}	s^{-1}
Monod half-rate constant	k_{O_2}	10^{-5}	kg/m^3
Biomass inactivation rate	κ_i	10^{-1}	s^{-1}
Biomass decay rate	κ_d	2×10^{-1}	s^{-1}
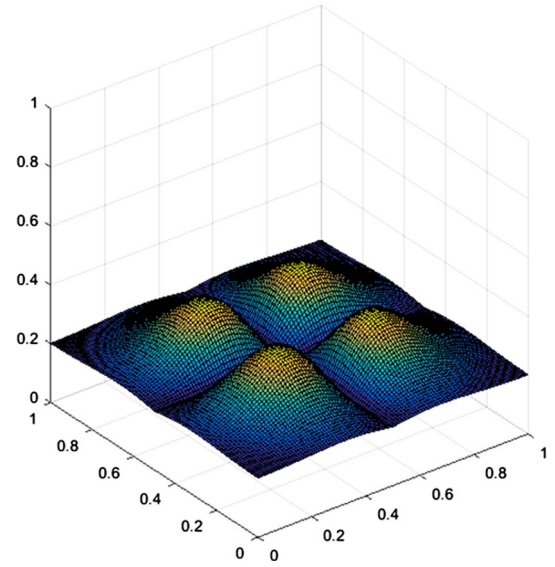
Fig. 20 Initial three-dimensional biofilm–fluid interface

The simulation result of a three-dimensional biofilm after growing for 1.9 days is shown in Fig. 21. The blue surface above the biofilm is the top surface of the diffusive boundary layer which is captured and reconstructed by the rolling ball algorithm presented in Sect. 3.3, and the red surface represents the biofilm–fluid interface. Figure 22 shows the simulation results of the three-dimensional biofilm–fluid interface at different times. The color maps in the figures represent the concentration of the oxygen. It is shown that the peak of the biofilm accesses the most oxygen.
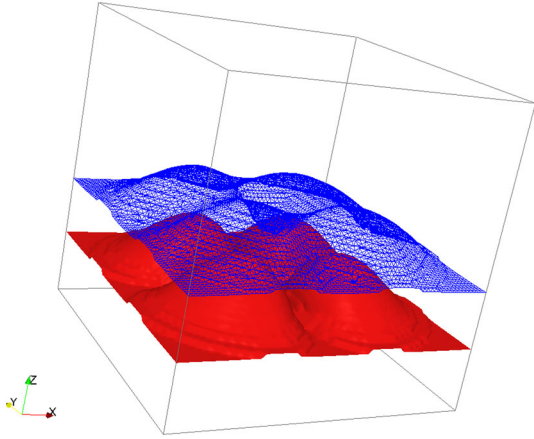
Fig. 21 Three-dimensional simulation result of the biofilm growth at $T = 1.9$

5 Summary and conclusion

A new numerical strategy for the simulation of multi-dimensional (in 2D and 3D) multi-species biofilm growth using the A–K model [2] is presented. The numerical framework is fully finite element method based. The standard finite element method is applied for solving the substrate mass balance equation as well as the potential equation by using 1st and 2nd order elements respectively. A rolling ball algorithm is applied to determine the substrate diffusive boundary layer. Stabilized TDG methods (TDG–SUPG and TDG–FIC) are applied to solve the hyperbolic transport PDE of each component of biomass in order to obtain accurate and stable numerical solutions. Measures for global numerical dispersion as well as numerical dissipation are applied to compare the performance of these two stabilized TDG methods. It demonstrates that both the TDG–SUPG and TDG–FIC methods lead to stable results. However, the TDG–FIC method is more suitable for solving the A–K model with the numerical strategy presented in this paper as the numerical dispersion is better controlled.

With the numerical technique presented in this paper, further studies on the influence of the reaction parameters, different biofilm patterns as well as the biomass distribu-
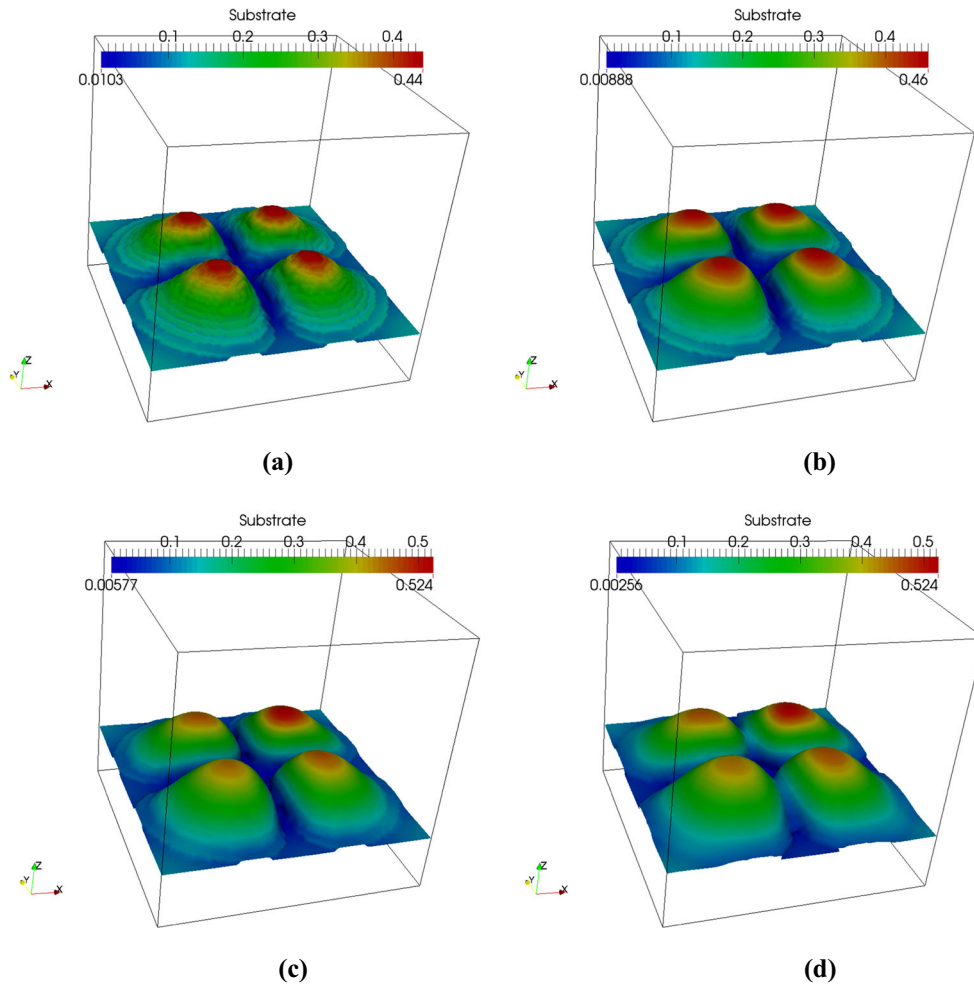

(a)

(b)

(c)

(d)

Fig. 22 Three-dimensional biofilm growth simulation of different time. **a** $T = 0.9$. **b** $T = 1.9$ (c) $T = 2.9$

tion and evolution over time can be studied. Naturally, the numerical strategy could be further improved. For instance using the XFEM [11] or finite cell method [39] could be promoting approaches to solve the Poisson equations with irregular boundary in the model. The *h-p* adaptive finite element framework [41] could also improve the numerical accuracy, especially for better resolution of the biofilm–fluid interface. It might also be beneficial to use DG in space to control the fluxes in order to improve the stability of the solution.

References

1. Albero AB, Ehret AE, Böl M (2014) A new approach to the simulation of microbial biofilms by a theory of fluid-like pressure-restricted finite growth. Comput Methods Appl Mech Eng 272:271–289
2. Alpkvist E, Klapper I (2007) A multidimensional multispecies continuum model for heterogeneous biofilm development. Bull Math Biol 69(2):765–789
3. Brooks AN, Hughes TJ (1982) Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. Comput Methods Appl Mech Eng 32(1–3):199–259
4. Chen S, Merriman B, Osher S, Smereka P (1997) A simple level set method for solving stefan problems. J Comput Phys 135(1):8–29
5. Clarelli F, Di Russo C, Natalini R, Ribot M (2013) A fluid dynamics model of the growth of phototrophic biofilms. J Math Biol 66(7):1387–1408
6. Codina R (1998) Comparison of some finite element methods for solving the diffusion–convection–reaction equation. Comput Methods Appl Mech Eng 156(1):185–210
7. Cogan N, Keener JP (2004) The role of the biofilm matrix in structural development. Math Med Biol 21(2):147–166
8. Costerton JW, Stewart PS, Greenberg E (1999) Bacterial biofilms: a common cause of persistent infections. Science 284(5418):1318–1322
9. D'Acunto B, Frunzo L, Klapper I, Mattei MR (2015) Modeling multispecies biofilms including new bacterial species invasion. Math Biosci 259:20–26
10. Donea J, Huerta A (2003) Finite element methods for flow problems. Wiley, New Jersey
11. Duddu R, Bordas S, Chopp D, Moran B (2008) A combined extended finite element and level set method for biofilm growth. Int J Numer Meth Eng 74(5):848–870
12. Duddu R, Chopp DL, Voorhees P, Moran B (2011) Diffusional evolution of precipitates in elastic media using the extended finite element and the level set methods. J Comput Phys 230(4):1249–1264
13. Eberl HJ, Sudarsan R (2008) Exposure of biofilms to slow flow fields: the convective contribution to growth and disinfection. J Theor Biol 253(4):788–807
14. Eberl HJ, Picioreanu C, Heijnen J, Van Loosdrecht MC (2000) A three-dimensional numerical study on the correlation of spatial structure, hydrodynamic conditions, and mass transfer and conversion in biofilms. Chem Eng Sci 55(24):6209–6222
15. Eberl HJ, Parker DF, Van Loosdrecht MC (2001) A new deterministic spatio-temporal continuum model for biofilm development. Comput Math Methods Med 3(3):161–175
16. Esser DS, Leveau JHJ, Meyer KM (2015) Modeling microbial growth and dynamics. Appl Microbiol Biotechnol 99(21):8831–8846
17. Gebara F (1999) Activated sludge biofilm wastewater treatment system. Water Res 33(1):230–238
18. Gibou F, Fedkiw RP, Cheng LT, Kang M (2002) A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. J Comput Phys 176(1):205–227
19. Gottlieb S, Shu C (1998) Total variation diminishing Runge–Kutta schemes. Math Comput Am Math Soc 67(221):73–85
20. Gray WG, Pinder GF (1976) An analysis of the numerical solution of the transport equation. Water Resour Res 12(3):547–555
21. Harten A (1983) High resolution schemes for hyperbolic conservation laws. J Comput Phys 49(3):357–393
22. Horn H, Lackner S (2014) Modeling of biofilm systems: a review. Adv Biochem Eng/Biotechnol 146:53–76
23. Hughes TJ, Hulbert GM (1988) Space-time finite element methods for elastodynamics: formulations and error estimates. Comput Methods Appl Mech Eng 66(3):339–363
24. Hulbert GM (1992) Time finite element methods for structural dynamics. Int J Numer Meth Eng 33(2):307–331
25. Hundsdorfer W, Verwer JG (2003) Numerical solution of time-dependent advection–diffusion–reaction equations, vol 33. Springer, New York
26. Jiang G, Peng D (2000) Weighted ENO schemes for Hamilton–Jacobi equations. SIAM J Sci Comput 21(6):2126–2143
27. Klapper I, Dockery J (2002) Finger formation in biofilm layers. SIAM J Appl Math 62(3):853–869
28. Klapper I, Dockery J (2010) Mathematical description of microbial biofilms. SIAM Rev 52(2):221–265
29. Krause R, Mücke R, Rank E (1995) hp-version finite elements for geometrically non-linear problems. Commun Numer Methods Eng 11(11):887–897
30. Kreft JU, Picioreanu C, Wimpenny JW, Van Loosdrecht MC (2001) Individual-based modelling of biofilms. Microbiology 147(11):2897–2912
31. Laspidou CS, Rittmann BE (2002) A unified theory for extracellular polymeric substances, soluble microbial products, and active and inert biomass. Water Res 36(11):2711–2720
32. LeVeque RJ (2002) Finite volume methods for hyperbolic problems, vol 31. Cambridge University Press, Cambridge
33. Mabrouk N, Deffuant G, Tolker-Nielsen T, Lobry C (2010) Bacteria can form interconnected microcolonies when a self-excreted product reduces their surface motility: evidence from individual-based model simulations. Theory Biosci 129(1):1–13
34. de Miranda S, Mancuso M, Ubertini F (2010) Time discontinuous galerkin methods with energy decaying correction for non-linear elastodynamics. Int J Numer Meth Eng 83(3):323–346
35. Murray JD (2001) Mathematical biology II: spatial models and biomedical applications. Springer, New York
36. Nackenhorst U, Ziefle M, Suwannachit A (2010) Finite element techniques for rolling rubber wheels. In: Elastomere friction. Springer, pp 123–163
37. Oñate E, Miquel J, Zárate F (2007) Stabilized solution of the multidimensional advection–diffusion–absorption equation using linear finite elements. Comput Fluids 36(1):92–112
38. Osher S, Fedkiw R (2006) Level set methods and dynamic implicit surfaces, vol 153. Springer, New York

39. Parvizian J, Düster A, Rank E (2007) Finite cell method. Comput Mech 41(1):121–133

40. Picioreanu C, Van Loosdrecht MC, Heijnen JJ (2000) Effect of diffusive and convective substrate transport on biofilm structure formation: a two-dimensional modeling study. Biotechnol Bioeng 69(5):504–515

41. Rachowicz W, Oden J, Demkowicz L (1989) Toward a universal h–p adaptive finite element strategy part 3. Design of h–p meshes. Comput Methods Appl Mech Eng 77(1):181–212

42. Reichert P (1998) AQUASIM 2.0: computer program for the identification and simulation of aquatic systems, PEAK, vol Basiskurs B7/98. Swiss Federal Institute for Environmental Science and Technology (EAWAG), Dübendorf

43. Sapotnick A, Nackenhorst U (2012) A combined FIC–TDG finite element approach for the numerical solution of coupled advection-diffusion-reaction equations with application to a bioregulatory model for bone fracture healing. Int J Numer Meth Eng 92(3):301–317

44. Sherratt JA, Dallon JC (2002) Theoretical models of wound healing: past successes and future challenges. Comptes Rendus Biol 325(5):557–564

45. Soleimani M, Wriggers P, Rath H, Stiesch M (2016) Numerical simulation and experimental validation of biofilm in a multiphysics framework using an SPH based method. Comput Mech 58(4):619–633

46. Wang Q, Zhang T (2010) Review of mathematical models for biofilms. Solid State Commun 150(21–22):1009–1022

47. Wanner O, Gujer W (1986) A multispecies biofilm model. Biotechnol Bioeng 28(3):314–328

48. Wanner O, Reichert P (1996) Mathematical modeling of mixed-culture biofilms. Biotechnol Bioeng 49(2):172–184

49. Wimpenny JW, Colasanti R (1997) A unifying hypothesis for the structure of microbial biofilms based on cellular automaton models. FEMS Microbiol Ecol 22(1):1–16

50. Wise SM, Lowengrub JS, Frieboes HB, Cristini V (2008) Three-dimensional multispecies nonlinear tumor growthi: model and numerical method. J Theor Biol 253(3):524–543

51. Xu Z, Meakin P, Tartakovsky A, Scheibe TD (2011) Dissipative-particle-dynamics model of biofilm growth. Phys Rev E 83(6):066,702

52. Zhang T, Cogan N (2008a) Phase field models for biofilms. II. 2-d numerical simulations of biofilm–flow interaction. Commun Comput Phys 4(1):72–101

53. Zhang T, Cogan NG, Wang Q (2008b) Phase field models for biofilms. I. Theory and one-dimensional simulations. SIAM J Appl Math 69(3):641–669

54. Zienkiewicz OC, Taylor RL (2000) The finite element method: the basis, vol 1. Butterworth-Heinemann, Oxford