Numerical simulators for modeling one- and multi-dimensional multiphase transport and biodegradation

CRS4
SAI/WRM

C. Gallo

June 2001

Contents

1 Introduction

This chapter is dedicated to presenting the mathematical and numerical formulations used in this work for studying coupling phenomena between multiphase flow and biodegradation. Referring to the modeling review presented in Chapter 3 for detailed presentation of the mathematical formulation, here the numerical solution and code implementation of the equations are documented. The codes contain the implementation for the one- and multi-dimensional case. As documented in chapter 2, a number of models and numerical simulators have been described in the literature. Given that a large effort was made already in implementing multi-dimensional multiphase flow simulators, such as STOMP (*White and Oostrom* 1999) and TOUGH2 (*K. Pruess* 1991), there was no point to developing a brand new code. However, the research topic of this thesis, i.e., studying the coupling of multiphase flow and biodegradation, did not fit perfectly with the model already available, in the sense that not all the features needed were already implemented. Thus, the following choices have been made. On the one hand, an easy-to-use and easy-to-modify one-dimensional simulator for two-phase flow plus biodegradation was developed for preliminary and basic studies. On the other hand, an available multiphase flow simulator was selected and modified for introducing the same biodegradation module developed for the one-dimensional code.

The one dimensional simulator, 2FB (two-phase flow and biodegradation) implements a two-phase (NAPL and aqueous) flow model plus multispecies transport in the aqueous phase coupled with biodegradation in a porous medium. The model is constituted by different parts, namely, two-phase conservation equations, advection-dispersion transport in the aqueous phase, and biodegradation kinetics. Different numerical techniques, namely mixed finite elements and finite volumes, were applied for solving the different groups of the equations. Two-phase is solved using a standard *IMPES* (implicit-pressure explicit-saturation) discretization approach (*Aziz and Settari* 1979).

The STOMP (subsurface transport over multiple phases, *White and Oostrom* 1999) code was selected for the multidimensional case. It includes features such as fractured media, multiple-phase systems, non-wetting fluid entrapment, nonaqueous phase liquids, first-order chemical reactions, radioactive decay, solute transport, and nonequilibrium dissolution. The STOMP simulator solves the governing-conservation equations and constitutive functions using numerical techniques for nonlinear systems. These equations are discretized using the integrated-finite-difference method of *Patankar* (1980). Discretization in time is performed using a fully implicit backward Euler scheme, while the global coupling for solving nonlinearities is performed by applying the Newton-Raphson linearization technique.

Both simulators are integrated with the biomodule, which is constituted by a set of routines which include a number of different biodegradation formulations chosen from those presented in Chapter 3. The biomodule is integrated in time using implicit time-stepping and the Newton-Raphson iterative procedure for solving nonlinearities present in the kinetic equations.

The outline of the chapter is as follows. Section 2 is dedicated to the one-dimensional simulator 2FB. In section 2.1 the equations describing the process included in the 2FB simulator are listed. In section 2.2 the numerical solution of 2FB's equations is be presented.

Section 3 is dedicated to describing the modified version of the STOMP numerical simulator. Section 3.1 summarizes the equations describing the process included in the 2FB simulator will be described. In section 3.2 the numerical solution applied in the STOMP simulator will be presented. Then the coupling strategy for solving nonlinearities located in the different parts of the model are described and schematized through the use of a flowchart diagram.

Section 4 is dedicated to present biomodule's features. In section4.1 equations describing three biodegradation modeling approaches as presented, while in section 4.2 the numerical solutions are described. A description of biomodule's features is also included.

Finally, in section 5 the coupling between the biomodule and the computational cores of the two simulators are discussed. Comments on the computational costs, memory requirement, and convergence issues are made as well as on the applicability of those simulators. Tips and indications regarding the use of the simulators are also given. The application of these model through numerical simulations will be described in Chapter 6 and Chapter 7.

2 The one-dimensional model

2.1 The mathematical model

The one-dimensional model was conceived for dealing with a limited number of processes, such as two-phase (NAPL and aqueous) flow, NAPL dissolution in the aqueous phase, multispecies transport in the aqueous phase, and biodegradation. "Limited" is intended with respect to the case of the multi-phase, multi-component, and multi-dimensional STOMP model. This choice is motivated by the need of having a flexible and fast tool for analyzing fastly and accurately a restricted set of phenomena. In this section, equations governing two-phase flow, dissolution kinetics, and constitutive relationships are listed. Next, the numerical solution of these equations and their implementation in a one-dimensional model are described.

2.1.1 Two-phase flow

Two-phase flow in porous media is described by mass conservation equations and Darcy's law for each phase (*Hassanizadeh and Gray* 1997):

$$\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla\cdot(\rho_\alpha \mathbf{v}_\alpha) = \phi E_\alpha \quad \alpha = l, n \tag{1}$$

where ϕ [-] is porosity, S [-] is saturation, and ρ [M/L^3] is density. The source term E_α [M/L^3] is total mass transferred to phase α from other phases. We note that in our case, $E_l = -E_n$. Subscripts l and n stand for aqueous and NAPL phases, respectively. $\mathbf{v}_\alpha$ [L/M] is the superficial velocity of phase α and is given by the extended Darcy's law for multiphase systems (*Hassanizadeh and Gray* 1997):

$$\mathbf{v}_\alpha = -\left[\frac{k_{r\alpha}}{\mu_\alpha}\mathbf{K}\cdot(\nabla P_\alpha - \rho_\alpha \mathbf{g})\right] \quad \alpha = l, n \tag{2}$$

where $\mathbf{K}$ [L^2] is soil permeability tensor, $k_{r\alpha}$ [-] is relative permeability, μ_α [M/LT] is viscosity, P [M/LT^2] is pressure, $\mathbf{g}$ [ML/T^2] is the gravity force. These equations must be supplemented with relationships for relative permeability and capillary pressure in terms of saturation.

2.1.2 Multispecies transport

Under the assumption that no compositional effects are present (*Hassanizadeh and Gray* 1997), transport of dissolved species in the aqueous phase can be described using the well-known advection-dispersion-reaction equation. Since biodegradation occurs only in the aqueous phase, only transport in this phase is accounted for and the subscript l for concentrations dropped, namely $C_l^i \rightarrow C^i$:

$$\frac{\partial(\phi S_l\, C^i)}{\partial t} + \nabla \cdot \left(C^i \mathbf{v}_l - \phi S_l \mathbf{D_h^i} \cdot \nabla \mathbf{C^i} \right) = \phi S_l E_l^i - \phi S_l B^i \quad i = 1, N_{ds} \tag{3}$$

where C^i [M/L^3] is the concentration of species i, $\mathbf{D_h^i}$ [M/L^{2}T] is the diffusion-dispersion tensor, E_l^i [M/L^{3}T] is the rate of dissolution of species i into water, B^i [M/L^{3}T] is the biodegradation rate of species i and/or mass transfer to microcolonies or biofilms, and N_{ds} is the total number of dissolved species; these could be electron acceptors (such as oxygen), nutrients, dissolved NAPL components, and transported bacteria.

2.1.3 Interphase mass transfer

NAPL dissolution in the aqueous phase is modeled using a kinetic mass-transfer equation (see Chapter 3 for a detailed discussion on this topic):

$$E_l^i = k_{do} \left(C_l^{i,eq} - C^i \right) \tag{4}$$

where $C_l^{i,eq}$ [M/L^3] is the equilibrium dissolution concentration of the species i in the aqueous phase. We note that $E_l = \sum_{i=1}^{N_{ds}} E_l^i$. The mass-transfer rate constant k_{do} [1/T] is generally dependent on many parameters, but in the current implementation it is assumed to be a given constant.

2.1.4 Capillary and relative permeability curves

These formulas incorporate the dependency of the relative permeability and capillary pressure from saturation. By assuming water as the wetting fluid, capillary pressure is defined as $P_c = P_n - P_l$. Fluid saturations are considered as unique functions of the capillary pressures, namely, no hysteretical behaviours included. The Brooks-Corey function (*Brooks and Corey* 1964) relating saturation to capillary head is considered:

$$P_c = P_d * S_{le}^{-1/\lambda} \tag{5}$$

where P_d [M/LT^2] is the entry pressure and λ [-] is fitting parameter related to the pore-size distribution. S_{le} is the effective aqueous saturation, defined as $S_{le} = (S_l - S_{lr})/(1 - S_{lr} - S_{nr})$, and S_{lr} and S_{nr} are the wetting- and nonwetting-phase residual saturations, respectively. Relative permeabilities for fluid phases take the following form

$$k_{rl} = (S_{le})^m \qquad k_{rn} = (1 - S_{le})^m \tag{6}$$

where m [-] is determined experimentally.

2.2 The numerical solution

The numerical solution of the 2FB model is quite complex, since a number of different techniques are applied. Phase pressures and saturations are discretized by an *IMPES*-like approach (*Aziz and Settari* 1979), which first solves for a total velocity $\mathbf{v}_T$ and the aqueous pressure P_l, and then advances the saturations to the next timestep by an explicit Runge-Kutta finite volume method. Mixed-hybrid finite elements (*Brezzi et al.* 1991) are used to compute pressure and pressure gradients. Saturation equation is discretized in space using a 2^{nd}-order in space TVD FV (finite volume) discretization (*Harten* 1986,1987) and using a 2^{nd} order in time RK (Runge-Kutta). An analogous discretization is implemented for the transport equation. Biodegradation of dissolved NAPL is advanced in time using a OP (operator splitting) technique for decoupling the advection from dispersion and reaction contributions.

2.2.1 The split formulation

Let us rewrite the system of equations for phase pressures and saturations as

$$\left\{ \begin{array}{lcl} \dfrac{\partial}{\partial t}(\phi \rho_l S_l) & + \ \nabla \cdot \rho_l \mathbf{v}_l & = \ \phi E_l \\ \dfrac{\partial}{\partial t}(\phi \rho_n S_n) & + \ \nabla \cdot \rho_n \mathbf{v}_n & = \ \phi E_n \end{array} \right. . \tag{7}$$

Phase velocities $\mathbf{v}_l$, $\mathbf{v}_n$ are given by the Darcy's-like relation

$$\left\{ \begin{array}{lcl} \mathbf{v}_l & = & -\lambda_l \left(\nabla P_l - \rho_l \mathbf{g}\right), \\ & & \\ \mathbf{v}_n & = & -\lambda_n \left(\nabla P_n - \rho_n \mathbf{g}\right), \end{array} \right. \tag{8}$$

with $\lambda_l = \dfrac{k_{rl}}{\mu_l}\mathbf{K}$ and $\lambda_n = \dfrac{k_{rn}}{\mu_n}\mathbf{K}$. By introducing the total velocity $\mathbf{v}_T = \mathbf{v}_l + \mathbf{v}_n$ and the total "fractional ratio" $\lambda_T = \lambda_l + \lambda_n$, we can write $\mathbf{v}_T$ as

$$\mathbf{v}_T = -\left(\lambda_l + \lambda_n\right) \nabla P_l + q_S \tag{9}$$

where the source term q_S is defined as

$$q_S = \left(\lambda_l \rho_l + \lambda_n \rho_n\right) \mathbf{g} - \lambda_n P_c' \nabla S_l. \tag{10}$$

Introducing $Q_T = \frac{E_l}{\rho_l} + \frac{E_n}{\rho_n}$, one obtains

$$\nabla \cdot \mathbf{v}_T = Q_T \tag{11}$$

Thus, the final system is obtained:

$$\begin{cases} \nabla \cdot \mathbf{v}_T &=& Q_T \,, \\ \mathbf{v}_T &=& -\left(\lambda_l + \lambda_n\right) \nabla P_l + q_S \,. \end{cases} \tag{12}$$

This set of equation is discretized by a mixed-hybrid finite element approach. The method of approximation gives an estimate of the normal flux of $\mathbf{v}_T$, which is necessary to advance in time the equation (7) for S_l. The following piece of metacode summarizes the iterative split approach which advances in time the initial solution:

General initialization
Read Input
t=0 **while** $t < t_{max}$
  loop [Global] **until** convergence
    loop [P-S] **until** convergence
      estimate $S^{t+\Delta t,k}, \nabla S^{t+\Delta t,k}$
      compute $P^{t+\Delta t,k+1}$ & $\mathbf{v}_T^{t+\Delta t,k+1}$
      upwind $S^{t+\Delta t,k} \rightarrow \mathbf{v}_l^{t+\Delta t,k+1}, \mathbf{v}_n^{t+\Delta t,k+1}$
      compute $S^{t+\Delta t,k+1}$
      check: $||S^{t+\Delta t,k+1} - S^{t+\Delta t,k}||_2 < \epsilon_{sat}$
    end loop
    loop [C-bio] **until** convergence
      compute $\tilde{C}^{i,t+\Delta t,m+1}$ (adv/disp/mass-transf.)
      call Biomodule (biodegradation)
      check: $||C^{i,t+\Delta t,m+1} - C^{i,t+\Delta t,m}||_2 < \epsilon_{conc} \quad i = N_{d.s.}$
    end loop
    check: Global convergence
  end loop
  if Biofilm-model then
    compute $\phi^{t+\Delta t}$ (porosity)
    compute $K^{t+\Delta t}$ (permeability)
  end if
  $t = t + \Delta t$
end while
print final results & statistics
Stop

where k and m are two iteration indexes, and ϵ_{sat} and ϵ_{conc} are user-specified tolerances. Basically, the algorithm consists in the time loop `(Global)`, which is split in two subloops, respectively called `(P-S)` and `(C-bio)`. At each timestep the former subloop updates the phase pressures P_l, P_n and the saturations S_l, S_n. The latter one updates concentrations C^i and the microbial distribution C^X.

At each step, system (12) is solved by approximating the normal component of the total velocity field at cell interfaces, $\mathbf{v}_T \cdot \mathbf{n}$ say, by the discontinuous lowest order mixed-hybrid

Raviart-Thomas finite elements and the aqueous phase pressure P_l by piecewise constant elements over the computational cells and at cell interfaces (*Brezzi* 1991). Let us denote these approximate fields by respectively $\mathbf{q}_h$, ψ_h and λ_h. The symbol h denotes the maximum diameter of all the cells K in a given mesh $\mathcal{T}_h$, $|K|$ is the cell volume and ∂K the cell boundary. The variational formulation reads as

$$\begin{cases} \int_\Omega \frac{\mathbf{K}^{-1}}{(\lambda_l+\lambda_n)}\mathbf{q}_h\cdot\mathbf{w}_h - \sum_K \int_K \psi_h \nabla\cdot\mathbf{w}_h + \sum_K \int_{\partial K} \lambda_h\,\mathbf{w}_h\cdot\mathbf{n}_K + \int_\Omega q_S \frac{\mathbf{K}^{-1}}{(\lambda_l+\lambda_n)}\mathbf{q}_h\cdot\mathbf{w}_h = 0\,, \\ \sum_K \int_K v_h \nabla\cdot\mathbf{q}_h = \int_\Omega Q_T v_h\,, \\ \sum_K \int_{\partial K} \mathbf{q}_h\cdot\mathbf{n}_K \mu_h = 0\,, \end{cases} \tag{13}$$

where the test functions $\mathbf{w}_h$, v_h and μ_h are taken in the same functional spaces of the corresponding unknowns. Standard algebraic manipulations yields a linear problem which is solved for the set of Lagrangian multipliers λ_h by using the static condensation technique. Backward substitution closes the solution algorithm (*Brezzi* 1991). This non-conforming finite element method guarantees that a zero-divergence constraint is well satisfied by the total velocity field. This issue is required to fairly simulate the non-linear degradation kinetics, as pointed out in (*Ewing* 1998).

The semi-discrete finite volume formulation that approximates equations (1) is obtained by integrating separately in a cell-wise fashion each phase saturation equation, applying the divergence theorem, approximating the interface integrals with the midpoint rule, and finally computing the advective fluxes by an upwind estimation of the phase velocities $\mathbf{v}_l$ and $\mathbf{v}_n$. This method takes the form

$$|K|\frac{dS_K}{dt} + \sum_{k\in\sigma(K)} l_k H(S_k^{int}, S_k^{ext}, \mathbf{n_k}) + \sum_{k'\in\sigma'(K)} l_{k'} H_{k'}^{(bc)}(S_k^{int}) = |K| Q(S_K), \qquad \forall K \in \mathcal{T}_h,$$

where S_K indicates the K-th cell-averaged value for both S_l and S_n, $\sigma(K)$ is the set of cells adjacent to K, $\sigma'(K)$ is the set of boundary edges on ∂K, $H(S_k^{int}, S_k^{ext}, \mathbf{n_k})$ and $H_{k'}^{(bc)}(S_k^{int})$ are the numerical fluxes at internal and boundary edges, $Q(S_K)$ stands for the source terms in (1), and l_k, $l_{k'}$ are the edge lengths. The numerical fluxes depend on S_k^{int} and S_k^{ext}, which are the traces at each cell interface of the linearly reconstructed saturation. The boundary flux $H_{k'}^{(bc)}(S_k^{int})$ may also depend on a set of suitable *external* data. A special linear reconstruction procedure interpolates the cell-averaged saturation values $\{S_K\}_{K\in\mathcal{T}_h}$ and ensures a TDV stability condition by limiting the reconstructed slopes. This avoids order one oscillations to appear in sharp gradient solution regions (*Ewing* 1998). An explicit two-stage 2^{nd}-order Runge-Kutta scheme advances in time the approximate solutions S_K. The method is formally second order accurate (*Gallo and Manzini* 1996). A predictor-corrector strategy is deserved to treat the non-linear dependence on conventrations C in the sink source term B^i. As for the phase saturations, the concentration equation is discretized by using a second order finite volume scheme plus TVD reconstruction. Further details are given in (*Gallo and Manzini* 1998a).

The transport of dissolved species, equation (3) includes also a diffusion-dispersion component. As well known from the literature this 2^{nd}-order component should be computed using an implicit scheme. Therefore, a standard 3-points finite difference scheme in space and a fully implicit Euler discretization in time were selected. Starting from:

$$\frac{\partial C}{\partial t}|_{disp} = \nabla \cdot \phi S_l(\mathbf{D_h} \cdot \nabla \mathbf{C}) \tag{14}$$

the discretization in space and in time (using fully implicit Euler scheme) is performed. Discretization For the i-th cell of the domain between the times t^k and t^{k+1}, it is:

$$\frac{C_i^{k+1} - C_i^k}{t^{k+1} - t^k} = \frac{1}{(\Delta x)^2} \left[\phi_{i+1/2}^k S_{l,i+1/2}^{k+1} \left(C_{i+1}^{k+1} - C_i^{k+1} \right) - \phi_{i-1/2}^k S_{l,i-1/2}^{k+1} \left(C_i^{k+1} - C_{i-1}^{k+1} \right) \right] \tag{15}$$

where the subscripts $i \pm 1/2$ indicate the interfaces between cell i and cells $i \pm 1$, respectively, where some ϕ and S_l are evaluated. It should be noted that ϕ^k and S_l^{k+1} were used, since S_l at $t + \Delta t$ is already available from the $P - S$ loop, while ϕ is updated only at the end of each time step. Equation (15) leads to a symmmetric three-band system that is solved using a standard iterative solver, such as conjugate gradient.

3 The multi-dimensional model

3.1 Mathematical formulation

As mentioned in the introduction, multi-dimensional multiphase flow is modeled using the STOMP code (*White and Oostrom* 1999). This simulator relies on a number of mass conservation equations plus a set of constitutive relationships which make explicit the reciprocal interactions between fluid and solid phases (see for example *Hassanizadeh and Gray* 1998). Conservation relations are written for each phase and for each component. A "phase" refers to the sum of all components comprising a fluid, such as gaseous, aqueous, and NAPL.

3.1.1 Phase conservation

Phase conservation equations is written as (*Hassanizadeh and Gray* 1997):

$$\frac{\partial(\phi \rho_\alpha S_\alpha)}{\partial t} + \nabla \cdot (\rho_\alpha \mathbf{v}_\alpha) = \phi \sum_\beta E_{\beta,\alpha} \quad \alpha = g, l, n \tag{16}$$

where ϕ [-] is porosity, S_α [-] is phase saturation, ρ [M/L^3] is phase density, $\mathbf{v}_\alpha$ [L/T] is phase velocity, $E_{\beta,\alpha}$ [M/L^{3}T] accounts for interphase mass exchange from phase β to phase α (this term is discussed later). Subscripts g, l, n stand for gaseous, aqueous, and NAPL phase, respectively.

3.1.2 Phase velocity

The phase velocity $\mathbf{v}_\alpha$ is given by the extended Darcy's law for multiphase systems (*Hassanizadeh and Gray 1997*):

$$\mathbf{v}_\alpha = -\left[\frac{k_{r\alpha}\mathbf{K}}{\mu_\alpha} \cdot (\nabla P_\alpha - \rho_\alpha \mathbf{g})\right] \quad \alpha = g, l, n \tag{17}$$

where P_α is the pressure [M/LT^2], $\mathbf{K}$ [L^2] is the permeability tensor, $k_{r\alpha}$ is the relative permeability function, μ_α [M/LT] is phase dynamic viscosity, and $\mathbf{g}$ [L/T^2] is gravity.

3.1.3 Component conservation

Generally, each component is assumed to be present in every phase and phase properties may be affected by its composition. For each component of a phase, conservation equation is given as:

$$\frac{\partial(\phi\rho_\alpha S_\alpha \omega_\alpha^i)}{\partial t} + \nabla \cdot \left(\rho_\alpha \omega_\alpha^i \mathbf{v}_\alpha - \mathbf{J}_\alpha^\mathbf{i}\right) = \phi \sum_\beta E_{\beta,\alpha}^i - \phi S_\alpha B_\alpha^i - (1-\phi) A_{ads}^i \quad \alpha = g, l, n \quad i = o, w, a \tag{18}$$

where ω_α^i [-] is the mass fraction of component i in phase α, B_α^i [M/L^{3}T] is the biodegradation term, and $E_{\beta,\alpha}^i$ [M/L^{3}T] accounts for interphase mass exchange of component i from phase β to phase α, and A_{ads}^i for adsorption on the soil matrix. $\mathbf{J}_\alpha^\mathbf{i}$ [M/L^{2}T] is the diffusion-dispersion component based on Fick's law.

3.1.4 Transport of diluted species in the aqueous phase

Dissolved oil is not the only component that is dissolved in the aqueous phase. Other species, such as oxygen, nitrates, nutrients, etc. can be transported by the bulk motion of the aqueous phase. In this case, however, their concentrations are quite small and possible variations in concentration of these components do not affect aqueous phase properties and the flow pattern. Therefore, their transport is modeled using the classical advection-dispersion-reaction equation:

$$\begin{aligned} \frac{\partial(\phi S_l\; C_l^i)}{\partial t} \quad &+ \quad \nabla \cdot \left(C_l^i \mathbf{v}_l - \phi S_l \mathbf{D}_{\mathbf{h,l}}^{\mathbf{i}} \nabla \mathbf{C}_\mathbf{l}^\mathbf{i}\right) = \phi \sum_\beta E_{l,\beta}^i - \phi S_l B_l^i \\ &-(1-\phi) A_{ads}^i \quad \beta = g, n \quad i = 1, N_{d.s.} \end{aligned} \tag{19}$$

where C_α^i [M/L^3] is the concentration of species j in phase α, $D_{h,\alpha}^i$ [M/L^{2}T] is the diffusion-dispersion flow term based on Fick's law, B_α^i [M/L^{3}T] is the biodegradation term of species i in phase l, and A_{ads}^i accounts for adsorption. The subscript $d.s.$ stands for dissolved species in phase l.

The equation set presented is quite typical for presenting multiphase flow and transport in porous media. In the case of the STOMP code, the equations are slightly modified. These are equivalent to that shown in equations (16)-(19), although some simplifying assumptions are applied. Dissolution of water and air in the NAPL phase is neglected; partitioning of oil, water, and air among the phases is computed at equilibrium; only oil can adsorb on the soil matrix and this adsorption is instantaneous. Let us consider equation (18) and sum over all the phases. The results is a conservation equation for component i over all phases.

3.1.5 Water conservation

:

$$\frac{\partial}{\partial t}\left[\sum_{\alpha=g,l}(\phi\rho_\alpha S_\alpha \omega_\alpha^w)\right]+\sum_{\alpha=g,l}(\nabla\cdot\omega_\alpha^w\rho_\alpha\mathbf{v}_\alpha-\mathbf{J}_\alpha^{\mathbf{w}})=q^w \tag{20}$$

where $\mathbf{J}_\alpha^{\mathbf{w}}$ is defined as:

$$\mathbf{J}_\alpha^{\mathbf{w}}=-\tau\phi\rho_\alpha\frac{\mathbf{M}^{\mathbf{w}}}{\mathbf{M}_\alpha}\mathbf{D}_\alpha^{\mathbf{w}}\cdot\nabla\chi_\alpha^{\mathbf{w}}\quad \alpha=\mathbf{g},\mathbf{l} \tag{21}$$

where τ [-] is tortuosity, M^w [-], and M_α [-] are the molecular weights of water and of phase α, respectively, and χ_α^w [-] is the molar fraction of water in phase α. q^w [M/L^{3}T] is a source/sink term accounted for explicitely.

3.1.6 Air conservation

:

$$\frac{\partial}{\partial t}\left[\sum_{\alpha=g,l}(\phi\rho_\alpha S_\alpha \omega_\alpha^a)\right]+\sum_{\alpha=g,l}(\nabla\cdot\omega_\alpha^a\rho_\alpha\mathbf{v}_\alpha-\mathbf{J}_\alpha^{\mathbf{a}})=q^a \tag{22}$$

where $\mathbf{J}_\alpha^{\mathbf{a}}$ is defined as:

$$\mathbf{J}_\alpha^{\mathbf{a}}=-\tau\phi\rho_\alpha\frac{\mathbf{M}^{\mathbf{a}}}{\mathbf{M}_\alpha}\mathbf{D}_\alpha^{\mathbf{a}}\cdot\nabla\chi_\alpha^{\mathbf{a}}\quad \alpha=\mathbf{g},\mathbf{l} \tag{23}$$

M^a is the molecular weights of air, and q^a [M/L^{3}T] is a source/sink term accounted for explicitely.

3.1.7 Oil conservation

:

$$\frac{\partial}{\partial t}\left[\sum_{\alpha=g,l,n}(\phi\rho_\alpha S_\alpha \omega_\alpha^o)+(1-\phi)\omega_s^o\rho_s\right]+\sum_{\alpha=g,l,n}(\nabla\cdot\omega_\alpha^o\rho_\alpha\mathbf{v}_\alpha-\mathbf{J}_\alpha^{\mathbf{o}})=q^o \tag{24}$$

where $\mathbf{J}_\alpha^{\mathbf{o}}$ is defined as:

$$\mathbf{J}_\alpha^{\mathbf{o}}=-\tau\phi\rho_\alpha\frac{\mathbf{M}^{\mathbf{o}}}{\mathbf{M}_\alpha}\mathbf{D}_\alpha^{\mathbf{o}}\cdot\nabla\chi_\alpha^{\mathbf{o}}\quad \alpha=\mathbf{g},\mathbf{l},\mathbf{n} \tag{25}$$

where M^o [-] is the molecular weight of oil and q^o [M/L^{3}T] is a source/sink term accounted for explicitely.

3.1.8 Transport of solutes

The solutes mass concentration equation assumes that the solute mass is partitioned among the fluid and the solid phases assuming thermodynamic and geochemical equilibrium. Solute transport is assumed to take place in every phase by advection and diffusion-dispersion. The following expression is obtained by extending equation (19) for every phase, namely n, l, and g, and summing over all phases. Thus solute transport equations reads as follows:

$$\frac{\partial C^i}{\partial t} = - \sum_{\alpha=l,g,n} \left(\nabla \cdot \left[C^i_\alpha \mathbf{v}_\alpha \right] \right) + \sum_{\alpha=l,g,n} \left(\nabla \left[\tau_\alpha S_\alpha \phi D^{C^i}_\alpha + \phi S_\alpha \mathbf{D}_{\mathbf{h},\alpha} \right] C^i_\alpha \right) + \sum_{\alpha=l,g,n} \phi S_\alpha B^i_\alpha \quad (26)$$

where

$$C = \sum_{\alpha=l,g,n} \phi S_\alpha C^i_\alpha + \ [1 - \phi] \, C^i_s \quad (27)$$

and C_s is concentration in the solid phase.

3.1.9 Constitutive relationships

A number of constitutive relationships are implemented in the STOMP simulator. Different choices are available for relative permeability curves, capillary curves, density, tortuosity, etc. Including these equations would be cumbersome and not important for the purpose of this work. Thus, the interested reader can refer to the STOMP theory guide (*White and Oostrom* 1999).

3.2 The numerical solution

3.2.1 General features

The numerical discretization in space and in time of the balance equations of the components air, water, and oil, and the solute transport is briefly described (refer to the STOMP theory guide (*White and Oostrom* 1996 for further detail).

The STOMP simulator solves the governing conservation equations and constitutive functions using numerical techniques for nonlinear systems. These equations are discretized using the integrated-finite-difference method of *Patankar* (1980). This method is locally and globally conserving and requires the physical domain be spatially discretized into an orthogonal computational domain which comprises nonoverlapping volumes (nodes). Each volume can have a maximum of two neighbouring nodes for each dimensional direction. Intrinsic properties are assumed to be uniform over the volume domain and are defined for a node point at the geometric center of the volume. Fluxes are defined at the geometric center of the surfaces between node volumes and along a direction parallel to the surface normal. Fluxes across node surfaces between nodes adjacent to the domain boundary are controlled through boundary conditions.

The system of algebraic equations that include the discretized governing conservation equations and the constitutive functions is nonlinear. Nonlinearities in the soil moisture retention functions, relative permeability functions, and physical properties near phase transitions are the primary contributors. Nonlinearities are solved by applying the Newton-Raphson linearization technique, which yields quadratic convergence of the residuals, given a sufficiently close estimates of the primary unknowns. At this point it is useful to schematize through a piece of metacode how the STOMP simulator works.

Start Execution
em do initialization
set chemical-physical properties
while $t < t_{max}$ *solve* FLOW problem
loop [Newton] **until** convergence
    set Boundary Properties and Fluxes, Source terms;
    build Jacobian matrix
    solve Linear System
    compute Convergence **if** convergence [Newton] = true **Then**
        Exit loop [Newton]
    else
        if not iteration limit [Newton] **then**
            perform another iteration
        else
            set Convergence [Newton] = False
            exit loop [Newton]
        end if
    end if
if convergence [Newton] = true **then**
    compute transport solution
    call Biomodule
    print results
    update Primary Variables & Properties
    set $t = t + \Delta t$
else
    if Time Reduction limit not reached **Then**
        reduce time step
        return to solve flow
    else
        STOP simulation
        print diagnostics
    end loop
print final results & statistics
STOP Execution

3.2.2 Spatial and temporal discretization

3.2.2.1 Accumulation/source terms The mass conservation equations (20), (22), and (24) are discretized by assuming a piecewise profile to express the variation in primary variables between node points and integrating over the node volume. The accumulation terms are integrated over the node volume. Intrinsic properties for the node volume are represented by properties at the node centroid. Source terms are similarly integrated over the node volume, as follows:

$$\frac{\partial}{\partial t}\left[\overline{M}^w V^e\right] = \frac{\partial}{\partial t}\left[V^e \sum_{\alpha=l,g} (\phi \omega_\alpha^w \rho_\alpha S_\alpha)\right] \tag{28}$$

$$\frac{\partial}{\partial t}\left[\overline{M}^a V^e\right] = \frac{\partial}{\partial t}\left[V^e \sum_{\alpha=l,g} (\phi \omega_\alpha^a \rho_\alpha S_\alpha)\right] \tag{29}$$

$$\frac{\partial}{\partial t}\left[\overline{M}^o V^e\right] = \frac{\partial}{\partial t}\left[V^e \sum_{\alpha=l,g,n} (\phi \omega_\alpha^o \rho_\alpha S_\alpha) + (1-\phi)\omega_s^o \rho_s\right] \tag{30}$$

where

$$\frac{\partial}{\partial t}\left[\int_{V^e} \overline{M}^j dV\right] = \frac{\partial}{\partial t}\left[\overline{M}^j V^e\right] \quad \text{for} \quad j = w, a, o \tag{31}$$

and

$$\int_{V^e} q^j dV = q^j V^e \quad \text{for} \quad j = w, a, o \tag{32}$$

3.2.2.2 Flux terms Flux terms, advection and diffusion/dispersion, are evaluated on the node surfaces. Integration of the flux terms over the node volume is performed by applying Green's theorem to the volumetric integrals to obtain the integrals over the cell boundaries. Control volumes are transformed into nodal control surfaces. Flux directions parallel to the surfaces are transformed in summations over the six faces. This transformation strictly requires an orthogonal grid system for the flux directions to be aligned with the surface normals, in order to avoid mass balance errors.

Darcy fluxes are discretized in the six coordinated directions (Top,Bottom, North, South, East, and West – T, B, N, S, E, W, respectively), using (by default) upwind interfacial averaging for the component mass fraction, phase density, and relative permeability, while harmonic averaging is used for the intrinsic permeability and phase viscosity.

$$\int_{V^e}\left[\sum_{\alpha=l,g,n}\left(\nabla\omega_\alpha^j\rho_\alpha\mathbf{v}_\alpha + \nabla J_\alpha^j\right)\right] dV = \int_{\Gamma^e}\left[\sum_{\alpha=l,g,n}\left(\nabla\omega_\alpha^j\rho_\alpha\mathbf{v}_\alpha + \nabla J_\alpha^j\right)\right]\cdot \mathbf{n} d\Gamma \quad \text{for} \quad j = w, a, o \tag{33}$$

where F_α^j and J_α^j are defined in equations (20), (22), (24), for $j = w, a, o$, respectively. By transforming the integral (33) into the summation over the six surfaces of the node, we get:

$$\int_{\Gamma^e}\left[\sum_{\alpha=l,g,n}\left(\nabla\omega_\alpha^j\rho_\alpha\mathbf{v}_\alpha + J_\alpha^j\right)\right]\cdot \mathbf{n} d\Gamma = \sum_{\varsigma=W,E,S,N,T,B}\left[\sum_{\alpha=l,g,n}\left(\nabla(\omega_\alpha^j\rho_\alpha\mathbf{v}_\alpha)_\varsigma^j + J_{\alpha_\varsigma}^j\right)\right] A_\varsigma \qquad j = w, a, o \tag{34}$$

where A_ς is the area of the ς face, and where we specify the following:

$$(\omega_\alpha^j\rho_\alpha\mathbf{v}_\alpha)_\varsigma^j = \frac{\langle\ \omega_\alpha^j\rho_\alpha k_{r,\alpha}\ \rangle_\varsigma^{uw}\ \langle\mathbf{K}\rangle_\varsigma^{\mathbf{h}}}{\langle\mu_\alpha\rangle_\varsigma^h}\left(\frac{\left(\ P_{\alpha_{\varsigma+}} - P_{\alpha_{\varsigma-}}\ \right)}{\Delta x_\varsigma} + \langle\rho_\alpha g\rangle_\varsigma^a \mathbf{z}_{g_\varsigma}\right)$$

$$
\begin{aligned}
&\text{for} \quad \alpha = l, g, n; \quad j = w, a, o; \quad \varsigma = E, W, S, N, B, T && (35)\\
J^j_{\alpha_\varsigma} &= -\left\langle \tau_\alpha \phi \rho_\alpha S_\alpha \frac{M^j}{M_\alpha} D^j_\alpha \right\rangle^h_\varsigma \frac{\left(\chi^j_{\alpha_{\varsigma+}} - \chi^j_{\alpha_{\varsigma-}}\right)}{\Delta x_\varsigma}\\
&\text{for} \quad \alpha = l, g, n; \quad j = w, a, o; \quad \varsigma = E, W, S, N, B, T && (36)
\end{aligned}
$$

Subscripts uw, h, and a stand for upwind, harmonic, and average respectively, and refer to how parameters are evaluated at cell interfaces.

3.2.2.3 Time discretization The mass conservation equations are discretized in time using a fully implicit scheme. The primary unknowns for the mass conservation equations are intrinsic properties at node volume centroids (node grid point) for time level $t + \Delta t$. In the residual form, the final discretized equation is written as

$$
\begin{aligned}
\left\{R^j\right\}^{t+\Delta t} &= V^e \left[\frac{\{M^j\}^{t+\Delta t} - \{M^j\}^t}{\Delta t}\right] - V^e \left\{q^j\right\}^{t+\frac{\Delta t}{2}}\\
&+ \sum_{\varsigma=W,E,S,N,T,B} \left[\sum_{\alpha=l,g,n} \left(\left\{\left(\omega^j_\alpha \rho_\alpha v_\alpha\right)^j_\varsigma\right\}^{t+\Delta t} + \left\{\mathbf{J}^{\mathbf{j}}_{\alpha_\varsigma}\right\}^{t+\Delta t}\right)\right] A_\varsigma \quad j = w, a, o \qquad (37)
\end{aligned}
$$

3.2.2.4 Global coupling The discretized governing equations for component mass conservation form a nonlinear set of algebraic equations. Nonlinearities arise from the dependence of secondary variables on the primary unknowns. A set of assumptions is made when solving the problem. The Newton-Raphson scheme is used for solving the nonlinear system of the discretized equation. It is assumed that concentrations of diluted species do not affect the flow pattern, thus they are excluded from the global coupling through the Newton Raphson linearization algorithm. This can be summarized as follows. Given a vectorial function $\mathbf{F}(\mathbf{x})$ of the vector variable $\mathbf{x}$, it is supposed that for $\mathbf{x} = \bar{\mathbf{x}}$ it is $\mathbf{F}(\bar{\mathbf{x}}) = \mathbf{0}$, while for a generic $\mathbf{x}$, $\mathbf{F}(\mathbf{x}) \neq \mathbf{0}$. Thus, if the correct solution $\bar{\mathbf{x}}$ is sufficiently close to $\mathbf{x}$, it can be written as a Taylor expansion around the $\mathbf{x}$, as follows (stopped at the first order derivative):

$$
\mathbf{F}(\bar{\mathbf{x}}) = \mathbf{F}(\mathbf{x} + \mathbf{\Delta x}) \approx \mathbf{F}(\mathbf{x}) + \mathbf{F}'(\mathbf{x})\,\mathbf{\Delta x} \qquad (38)
$$

where $\Delta\mathbf{x} = \bar{\mathbf{x}} - \mathbf{x}$ and $\mathbf{F}'$ is the Jacobian matrix of $\mathbf{F}$. The iteration scheme is the following:

$$
\begin{aligned}
\Delta \mathbf{x}^{\mathbf{m+1}} &= -\frac{\mathbf{F}(\mathbf{x}^{\mathbf{m}})}{\mathbf{F}'(\mathbf{x}^{\mathbf{m}})} && (39)\\
\mathbf{x}^{\mathbf{m+1}} &= \mathbf{x}^{\mathbf{m}} + \mathbf{\Delta x}^{\mathbf{m+1}} && (40)\\
\text{until} \quad \Delta \mathbf{x}^{\mathbf{m+1}} &< \epsilon && (41)
\end{aligned}
$$

where ϵ is a pre-specified tolerance.

The application of this scheme implies two major computational issues. One is related to the evaluation of the Jacobian matrix and the other is related to the solution of the linear system. In the STOMP simulator, the following choices have been made. The derivatives of $\mathbf{F}$ in the components of $\mathbf{x}$ are computed numerically as incremental ratios instead of using analytic derivatives due the numerical stability reasons. Thus, the numerical derivative of the i-th components of $\mathbf{F}$ in the j-th component of $\mathbf{x}$ is expressed as:

$$\frac{\partial F_i}{\partial x_j} \approx \frac{F_i|_{x_j+\Delta x_j} - F_i|_{x_j}}{\Delta x_j} \tag{42}$$

The Newton-Raphson procedure starts at any new time step, convergence failure, or time step reduction. Finally, the solution of the linear system of equation is solved using either a direct solver or an iterative solver (refer to STOMP user guide for further details).

3.2.2.5 Solute transport The solute mass conservation equation (19) is discretized by assuming a piecewise profile for the solute concentration between node points and integrating over the node volume. The advection and diffusion/dispersion transport terms of the solute mass conservation are combined following the power-law scheme of *Patankar* (1980) . Integration of the accumulation term for solute mass over the node volume proceeds as follows

$$\int_{V^e} \left(\frac{\partial C}{\partial t}\right) dV = \frac{\partial C}{\partial t} V^e \tag{43}$$

Solute transport fluxes are computed between node points and comprise advective and diffusive-dispersive components. The diffusive-dispersive term is computed using a user-defined interfacial average for the effective diffusion-dispersion coefficient (harmonic averaging by default). Thus, we write:

$$\begin{aligned}
&\int_{V^e} \left[- \sum_{\alpha=l,g,n} (\nabla C_\alpha \mathbf{v}_\alpha) + \sum_{l,g,n} \left(\nabla \left[\left(\tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \right) \nabla C_\alpha \right] \right) \right] dV = \\
&\int_{\Gamma} \left[- \sum_{\alpha=l,g,n} (C_\alpha \mathbf{v}_\alpha) + \sum_{l,g,n} \left(\left(\tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \right) \nabla C_\alpha \right) \right] \cdot \mathbf{n} d\Gamma = \\
&\sum_{\varsigma=W,E,S,N,T,B} \left[- \sum_{\alpha=l,g,n} (\nabla C_\alpha v_\alpha)_\varsigma + \sum_{l,g,n} \left(\left(\tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \right) \nabla C_\alpha \right)_\varsigma \right] A_\varsigma
\end{aligned} \tag{44}$$

where we have:

$$\begin{aligned}
v_{\alpha_\varsigma} &= \frac{\langle k_{r,\alpha} \rangle_\varsigma^{uw} \langle \mathbf{K} \rangle_\varsigma^{\mathbf{h}}}{\langle \mu_\alpha \rangle_\varsigma^h} \left(\frac{\left(P_{\alpha_{\varsigma+}} - P_{\alpha_{\varsigma-}} \right)}{\Delta x_\varsigma} + \langle \rho_\alpha g \rangle_\varsigma^a \mathbf{z}_{g_\varsigma} \right) \\
&\text{for} \quad \alpha = l, g, n; \quad \varsigma = E, W, S, N, B, T
\end{aligned} \tag{45}$$

and

$$\begin{aligned}
&\left(\left(\tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \right) \nabla C_\alpha \right)_\varsigma = \\
&\left\langle \tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \right\rangle_\varsigma^h \frac{\left(C_{\alpha_{\varsigma+}} - C_{\alpha_{\varsigma-}} \right)}{\Delta x_\varsigma} \\
&\text{for} \quad \alpha = l, g, n; \quad \varsigma = E, W, S, N, B, T
\end{aligned} \tag{46}$$

The solution of the solute transport equation depends on the local Peclet number. The power law scheme is based on the solute concentration profile for steady conditions with no source nor decay. The power-law scheme closely approximates the exact solution for steady conditions without excessive computational expense. Solute flux from combined advective and diffusive-dispersive transport can be expressed using the power-law scheme as follows:

$$
\begin{aligned}
Pe_{\alpha_\varsigma} &= \frac{V_{\alpha_\varsigma} \Delta x_\varsigma}{\langle \tau_\alpha S_\alpha \phi D_\alpha^C + S_\alpha \phi \mathbf{D}_{h,\alpha} \rangle_\varsigma^h} \qquad (47) \\
G_{\alpha_\varsigma}^C &= -C_{\alpha_{\varsigma+}} \left[\max\left(-V_{\alpha_\varsigma}, 0\right) + \left(\mathbf{D}_{\alpha_e}^C\right)_\varsigma \max\left(\left(1 - \frac{0.1 \ |V_{\alpha_\varsigma}|}{\mathbf{D}_{\alpha_e}^C}\right)^5, 0 \right) \right] \\
&+ \ C_{\alpha_{\varsigma-}} \left[\max\left(V_{\alpha_\varsigma}, 0\right) + \left(\mathbf{D}_{\alpha_e}^C\right)_\varsigma \max\left(\left(1 - \frac{0.1 \ |V_{\alpha_\varsigma}|}{\mathbf{D}_{\alpha_e \ \varsigma}^C}\right)^5, 0 \right) \right] \\
&\text{for} \quad \alpha = l, g, n \quad \varsigma = W, E, S, N, B, T \qquad (48)
\end{aligned}
$$

where

$$\left(\mathbf{D}_{\alpha_e}^C\right)_\varsigma = \frac{\left\langle \tau_\alpha S_\alpha \phi D_\alpha^C \ + S_\alpha \phi \mathbf{D}_{h,\alpha} \ \right\rangle_\varsigma^h}{\Delta x_\varsigma} \qquad (49)$$

According to the previous definitions, we can write the solute transport equation in the spatial discretized form:

$$\sum_{l,g,n} \left(\sum_\varsigma G_{\alpha_\varsigma}^C \right) = \frac{\partial C}{\partial t} V^e \qquad (50)$$

Equation (50) is to be advanced in time. The time discretization is made using a fully implicit scheme, which reads as follows:

$$\sum_{l,g,n} \left(\sum_\varsigma G_{\alpha_\varsigma}^C \right)^{t+\Delta t} = \frac{C^{t+\Delta t} - C^t}{\Delta t} V^e \qquad (51)$$

4 Biodegradation

In chapter 3 biodegradation modeling in the literature was reviewed. There three major modeling approaches differing in some underlying assumptions were identified. For each of these approaches a number of alternatives proposed were mentioned. In this section, the focus is on selecting a single modeling formulation for each of the three approaches. The complete set of equation is here re-written in detail and the numerical solution described. All these equations are incorporated in the bio-module which is plugged in both the 2FB and the STOMP simulators.

4.1 Mathematical model

In the following paragraphs the equations describing biodegradation of free-bacteria (FB), microcolony (MC), and biofilm (BF) models are briefly listed. Basic hypotheses and assumptions for each of the three models are described in detail in Chapter 3.

4.1.1 Free-bacteria equations

The first approach considered is the "free-bacteria" model (FB). In this model, the same concentration value of each chemical species is used in the transport equation (bulk flow) and in the equation describing the kinetics of degradation. The biodegradation rate (term B^i in equation 19) is given by:

$$B^i = C^X \mu_0^i \sum_{l=1}^{N_{EA}} Y_{i,l} \frac{C^i}{K^{i,l}+C^i} \frac{C^l}{K^l+C^l} \prod_{j=1}^{N_{nut}} \frac{C^j}{K^{j,l}+C^j} \quad i=1,N_S \tag{52}$$

$$B^l = C^X \frac{C^l}{K^l+C^l} \sum_{l=1}^{N_S} \mu_0^i Y_{i,l} \frac{C^i}{K^{i,l}+C^i} \prod_{j=1}^{N_{nut}} \frac{C^j}{K^{j,l}+C^j} \quad l=1,N_{EA} \tag{53}$$

$$B^j = C^X \sum_{l=1}^{N_S} Y_{i,j} \mu_0^i \sum_{l=1}^{N_{EA}} \frac{C^i}{K^{i,l}+C^i} \frac{C^l}{K^l+C^l} \prod_{j=1}^{N_{nut}} \frac{C^j}{K^{j,l}+C^j} \quad i=1,N_{nut} \tag{54}$$

where $K^{j,l}$ and K^l [M/L^3] are half-saturation constants, $Y^{j,l}$ yield coefficients, μ_0^i [1/T] is the maximum degradation rate constant. The kinetics of growth/decay of biomass is :

$$B^X = \frac{1}{C^X}\frac{\partial C^X}{\partial t} = \sum_{i=1}^{N^S} \mu_0^i \sum_{l=1}^{N_{EA}} \frac{C^i}{K^{i,l}+C^i} \frac{C^l}{K^l+C^l} \prod_{j=1}^{N_{nut}} \frac{C^j}{K^{j,l}+C^j} - k_d \tag{55}$$

where C^X [M/L^3] is the bacterial population density and k_d [1/T] is the bacterial decay constant.

4.1.2 Microcolony-based equations

The second approach is the "microcolonies" based model (MC). Growth/decay of increase/decrease in the number of occurs within the microcolonies only. All concentrations within microcolonies. The B-terms equation (19) reads as:

$$B^i = -N_c \kappa^i A_c \frac{(C^i - c_m^i)}{\delta}, \quad i=1,N_{d.s.} \tag{56}$$

where c_m^i [M/L^3] is the concentration of species i within the microcolonies, κ^i [L^2/T] is the mass-exchange coefficient between the bulk and microcolonies, m_c [M] is the mass of a microcolony, A_c [L^2] is the contact area of one microcolony for the mass diffusion process, δ [L] is the boundary layer between bulk flow and microcolonies.

$$\frac{dc^i}{dt} = -\mu_0^i Y_i \sum_{l=1}^{N_{EA}} \mu_0^i \frac{c^i}{K^{l,i}+c^i} \cdot \frac{c^l}{K^l+c^l} \cdot \prod_{j=1}^{N_{nut}} \frac{c^j}{K^{l,j}+c^j} + \kappa^i A_c \frac{(C^i-c^i)}{\delta} \quad i=1,N_s \tag{57}$$

$$\frac{dc^l}{dt} = -\frac{c^l}{K^l+c^l} \cdot \sum_{i=1}^{N_s} \mu_0^i Y_i \frac{c^i}{K^{l,i}+c^i} \cdot \prod_{j=1}^{N_{nut}} \frac{c^j}{K^{l,j}+c^j} + \kappa^l A_c \frac{(C^l-c^l)}{\delta} \quad l=1,N_{EA} \tag{58}$$

$$\frac{dc_j}{dt} = -\sum_{i=1}^{N_s} \mu_0^i Y_i \cdot \sum_{l=1}^{N_{EA}} \frac{c^i}{K^{l,i}+c^i} \cdot \frac{c^l}{K^l+c^l} \cdot \prod_{j=1}^{N_{nut}} \frac{c^j}{K^{l,j}+c^j} + \kappa^j A_c \frac{(C^j-c^j)}{\delta} \quad j=1,N_{nut} \tag{59}$$

Microcolony growth/decay kinetics reads as:

$$\frac{1}{N_c}\frac{\partial N_c}{\partial t} = \sum_{i=1}^{N^S} \mu_0^i \sum_{l=1}^{N_{EA}} \frac{c^i}{K^{i,l}+c^i}\frac{c^l}{K^l+c^l}\prod_{j=1}^{N_{nut}^i}\frac{c^j}{K^{j,l}+c^j} - k_d \tag{60}$$

4.1.3 The biofilm model equations

The third approach is the biofilm approach (BF). The key processes are biodegradation in the bulk phase, mass exchange between bulk flow and biofilm, and biodegradation within the biofilm. Moreover, porosity changes caused by biofilm growth/decay are also included. In this case the B-terms are written as follows:

$$B^S = C^X Y_S r_{bio} + E_{l,b}^S \tag{61}$$
$$B^O = C^X Y_O r_{bio} + E_{l,b}^O \tag{62}$$
$$B^X = -C^X r_{bio} + E_{l,b}^X + k_d C^X \tag{63}$$

where $E_{l,b}^i$ is the transfer of mass from the aqueous phase to the biofilm and r_{bio} is the rate of biodegradation within the aqueous phase, given by equation (64)

$$r_{bio} = \mu_0 \frac{C^S}{K^S+C^S}\frac{C^O}{K^O+C^O} \tag{64}$$

An equation for $E_{l,b}^i$ is needed. This reads as follows:

$$E_{l,b}^S = \kappa_b\left[K_b C^S - c_b^S\right] \tag{65}$$
$$E_{l,b}^O = \kappa_b\left[K_b C^O - c_b^O\right] \tag{66}$$
$$E_{l,b}^X = \kappa_b\left[K_b C^X - c_b^X\right] \tag{67}$$

where c_b^i is the concentration within the biofilm (mass of species per unit volume of the biofilm), κ_b [1/T] and K_b [-] are the mass transfer coefficient and the macroscale partition coefficient, respectively.

The equations describing the change of concentrations within the biofilm are (*Hassanizadeh* 1999):

$$\frac{\partial\left(\phi_b c_b^S\right)}{\partial t} = -\phi_b Y_S c_b^X r_{bio,b} + \phi E_{l,b}^S \tag{68}$$
$$\frac{\partial\left(\phi_b c_b^O\right)}{\partial t} = -\phi_b Y_O c_b^X r_{bio,b} + \phi E_{l,b}^O \tag{69}$$

where ϕ_b is biofilm volume fraction, i.e. fraction of total soil volume occupied by the biofilm. The rate of biodegradation within the biofilm, $r_{bio,b}$, is given by

$$r_{bio,b} = \mu_0 \frac{c_b^S}{K^S+c_b^S}\frac{c_b^O}{K^O+c_b^O} \tag{70}$$

An equation similar to (69) gives the rate of growth/decay of biofilms:

$$\frac{\partial\left(\phi_b c_b^X\right)}{\partial t} = \phi_b c_b^X r_{bio,b} - \phi_b k_d c_b^X + \phi E_{l,b}^X \tag{71}$$

The sum of ϕ and ϕ_b is equal to the total porosity of the medium which is assumed to be a constant. Thus, it holds:

$$\frac{\partial \phi}{\partial t} = -\frac{\partial \phi_b}{\partial t} \tag{72}$$

Assuming that the average mass of bacteria per unit volume of the biofilm, c_b^X is a known constant, the combination of equations (71) and (72) yields:

$$\frac{\partial \phi}{\partial t} = -\frac{\phi_b c_b^X r_{bio,b} - \phi_b c_b^X k_d \ + E_{l,b}^X}{c_b^X} \tag{73}$$

Permeability variations are described as follows (*Bird et al.* 1960):

$$\frac{K_{new}}{K_{old}} = \left(\frac{\phi_{new}}{\phi_{old}}\right)^3 \Big/ \left(\frac{1-\phi_{new}}{1-\phi_{old}}\right)^2 \tag{74}$$

4.2 Numerical solution of biodegradation equations

The solution of biodegradation implies the solution of both the transport in the aqueous phase and the biodegradation module. As widely described in the literature (*Yeh and Tripathi* 1989), the solution of multispecies transport with chemical reactions is generally accomplished in two separate steps: after computing advection-dispersion concentrations are updated accounting for to the contribution given by biodegradation. This procedure is chosen due to the prohibitive computational effort of implementing a global Newton-Raphson iterative algorithm. Actually, the operator-splitting solution procedure introduces a weak coupling between the equations: this coupling, however, can be strengthened by performing an iteration between the advection-dispersion and biodegradation blocks until convergence is achieved. Let us consider the following reactive transport equation:

$$\frac{\partial C}{\partial t} = \nabla \cdot (\mathbf{v}C) + \nabla \cdot (D \cdot \nabla C) + B(C) = L(C) + B(C) \tag{75}$$

where the advection/dispersion operator is defined as

$$L(C) = \nabla \cdot (\mathbf{v}C) + \nabla \cdot (\mathbf{D} \cdot \nabla \mathbf{C}) \tag{76}$$

and B accounts for biodegradation. The total variation of C in time within the control volume is split in two different contributions, i.e., the advective-dispersive component and the biodegradation component:

$$\frac{\partial C}{\partial t} = \left(\frac{\partial C}{\partial t}\right)_{adv,disp} + \left(\frac{\partial C}{\partial t}\right)_{bio} \tag{77}$$

Time derivatives are expressed as incremental ratios between t^k and t^{k+1}. This discretization is made in two steps, as follows:

$$
\begin{aligned}
\frac{\partial C}{\partial t} &= \left(\frac{\partial C}{\partial t}\right)_{adv,disp} + \left(\frac{\partial C}{\partial t}\right)_{bio} \\
\left(\frac{\Delta C}{\Delta t}\right) &= \left(\frac{\Delta C}{\Delta t}\right)_{adv,disp} + \left(\frac{\Delta C}{\Delta t}\right)_{bio} \\
\frac{C^{k+1} - C^k}{t^{k+1} - t^k} &= \frac{\tilde{C}^{k+1} - C^k}{t^{k+1} - t^k} + \frac{C^{k+1} - \tilde{C}^{k+1}}{t^{k+1} - t^k} \qquad (78)
\end{aligned}
$$

where $\tilde{C}^{k+1}$ is an intermediate concentration accounting for contribution of the advection-dispersion. Accounting for equations (75), (76), and (77), we can write:

$$
\begin{aligned}
\frac{\tilde{C}^{k+1} - C^k}{t^{k+1} - t^k} &= L(\tilde{C}^{k+1}) \qquad (79) \\
\frac{C^{k+1} - \tilde{C}^{k+1}}{t^{k+1} - t^k} &= B(C^{k+1}) \qquad (80)
\end{aligned}
$$

4.2.1 Free-bacteria model

Equations (52-54) describing the degradation of substrate(s), electron-acceptor(s), and nutrient(s) are discretized in time using an implicit time stepping. By recalling that

$$
B^{k+1} = \left(\frac{\Delta \mathbf{C}}{\Delta t}\right)_{bio} = \frac{\mathbf{C}^{k+1} - \tilde{\mathbf{C}}^{k+1}}{t^{k+1} - t^k} \qquad (81)
$$

the discretization in time is written as follows:

$$
\begin{aligned}
\frac{C^{i,k+1} - \tilde{C}^{i,k+1}}{t^{k+1} - t^k} &= -C^{X,k} \mu_0^i \sum_{l=1}^{N_{EA}} Y_{i,l} \frac{C^{i,k+1}}{K^{i,l} + C^{i,k+1}} \frac{C^{l,k+1}}{K^l + C^{l,k+1}} \prod_{j=1}^{N_{nut}} \frac{C^{j,k+1}}{K^{j,l} + C^{j,k+1}} \\
& \quad i = 1, N_S \qquad (82) \\
\frac{C^{l,k+1} - \tilde{C}^{l,k+1}}{t^{k+1} - t^k} &= -C^{X,k+1} \frac{C^{l,k+1}}{K^l + C^{l,k+1}} \sum_{l=1}^{N_S} \mu_0^i Y_{i,l} \frac{C^{i,k+1}}{K^{i,l} + C^{i,k+1}} \prod_{j=1}^{N_{nut}^i} \frac{C^{j,k+1}}{K^{j,l} + C^{j,k+1}} \\
& \quad l = 1, N_{EA} \qquad (83) \\
\frac{C^{j,k+1} - \tilde{C}^{j,k+1}}{t^{k+1} - t^k} &= -C^{X,k} \sum_{l=1}^{N_S} \mu_0^i \sum_{l=1}^{N_{EA}} Y_{j,i} \frac{C^{i,k+1}}{K^{i,l} + C^{i,k+1}} \frac{C^{l,k+1}}{K^l + C^{l,k+1}} \prod_{j=1}^{N_{nut}} \frac{C^{j,k+1}}{K^{j,l} + C^{j,k+1}} \\
& \quad i = 1, N_{nut} \qquad (84)
\end{aligned}
$$

The solution procedure implies the solution of the nonlinearity arising by the presence of $\mathbf{C}^{k+1}$ in both R.H.S. and L.H.S. of the equation. Therefore, the Newton-Raphson (NR) linearization scheme is applied for solving the nonlinear system, given its convergence properties (*Gallo and Manzini* 1998a,1998b). Let us define $\mathbf{C} = (C^1, ..., C^{N_s+N_{EA}+N_{nut}})$ as the array containing the concentrations of all the species present in the system, and $\mathbf{B} = (B^1, ..., B^{N_s+N_{EA}+N_{nut}})$. Thus, starting from equation:

$$
\mathbf{C}^{k+1} = \tilde{\mathbf{C}}^{k+1} + (t^{k+1} - t^k)\, \mathbf{B}(C^{k+1}) \qquad (85)
$$

equation (85) is re-written in the form of a residual function:

$$\mathbf{G}(\mathbf{C}^{k+1}) = \mathbf{C}^{k+1} - \tilde{\mathbf{C}}^{k+1} - (t^{k+1} - t^k)\mathbf{B}(\mathbf{C}^{k+1}) \tag{86}$$

Assuming that $\mathbf{G}(\mathbf{C}^{k+1} + \mathbf{C}^{k+1}) = 0$, $\mathbf{G}$ can be expanded around $\mathbf{C}^{k+1}$ using Taylor's formula and stopping at the first term, one obtains:

$$\mathbf{G}(\mathbf{C}^{k+1} + \Delta\mathbf{C}^{k+1}) \approx \mathbf{G}(\mathbf{C}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{C}^{k+1})\right)\Delta\mathbf{C}^{k+1} \tag{87}$$

where $\mathbf{J}$ is the Jacobian of the vector function $\mathbf{G}$. Imposing that

$$\mathbf{G}(\mathbf{C}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{C}^{k+1})\right)\Delta\mathbf{C}^{k+1} = \mathbf{0} \tag{88}$$

the iteration scheme appears as follows:

$$\begin{aligned}
\Delta\mathbf{C}^{k+1,m+1} &= -\mathbf{J}^{-1}\left(\mathbf{G}(\mathbf{C}^{k+1,m})\right)\mathbf{G}(\mathbf{C}^{k+1,m}) && (89)\\
\mathbf{C}^{k+1,m+1} &= \mathbf{C}^{k+1,m} + \Delta\mathbf{C}^{k+1,m+1} && (90)\\
\text{until} \quad & \Delta\mathbf{C}^{k+1,m+1} < \epsilon
\end{aligned}$$

where the first iteration for starting the scheme is $\mathbf{C}^{k+1,m} = \tilde{\mathbf{C}}^{k+1}$.

After solving for $\mathbf{C^{k+1}}$, microbial growth (equation 55) is integrated in time analytically at the end of each time iteration of the transport equations as follows:

$$\ln\left(\frac{C^{X,k+1}}{\tilde{C}^{X,k+1}}\right) = \left[\sum_{i=1}^{N^S}\mu_0^i \sum_{l=1}^{N_{EA}} \frac{C^{i,k+1}}{K^{i,l} + C^{i,k+1}} \frac{C^{l,k+1}}{K^l + C^{l,k+1}} \prod_{j=1}^{N_{nut}^i} \frac{C^{j,k+1}}{K^{j,l} + C^{j,k+1}} - k_d\right]\left(t^{k+1} - t^k\right) \tag{91}$$

4.2.2 Microcolony-based model

Equations (57)-(59) governing the MC model are discretized in time using an implicit (backward Euler) time stepping:

$$\begin{aligned}
\frac{c^{i,k+1} - c^{i,k}}{t^{k+1} - t^k} &= -\mu_0^i Y_i \sum_{l=1}^{N_{EA}} \mu_0^i \frac{c^{i,k+1}}{K^{l,i} + c^{i,k+1}} \cdot \frac{c^{l,k+1}}{K^l + c^{l,k+1}} \cdot \prod_{j=1}^{N_{nut}} \frac{c^{j,k+1}}{K^{l,j} + c^{j,k+1}}\\
&\quad + \kappa^i A_c \frac{(\tilde{C}^{i,k+1} - c^{i,k+1})}{\delta} \qquad i = 1, N_s && (92)\\
\frac{c^{l,k+1} - c^{l,k}}{t^{k+1} - t^k} &= -\frac{c^{l,k+1}}{K^l + c^{l,k+1}} \cdot \sum_{i=1}^{N_s} \mu_0^i Y_i \frac{c^{i,k+1}}{K^{l,i} + c^{i,k+1}} \cdot \prod_{j=1}^{N_{nut}} \frac{c^{j,k+1}}{K^{l,j} + c^{j,k+1}}\\
&\quad + \kappa^l A_c \frac{(\tilde{C}^{l,k+1} - c^{l,k+1})}{\delta} \qquad l = 1, N_{EA} && (93)\\
\frac{c^{j,k+1} - c^{j,k}}{t^{k+1} - t^k} &= -\sum_{i=1}^{N_s} \mu_0^i Y_i \cdot \sum_{l=1}^{N_{EA}} \frac{c^{i,k+1}}{K^{l,i} + c^{i,k+1}} \cdot \frac{c^{l,k+1}}{K^l + c^{l,k+1}} \cdot \prod_{j=1}^{N_{nut}} \frac{c^{j,k+1}}{K^{l,j} + c^{j,k+1}}\\
&\quad + \kappa^j A_c \frac{(\tilde{C}^{j,k+1} - c^{j,k+1})}{\delta} \qquad j = 1, N_{nut} && (94)
\end{aligned}$$

Equations (92)-(94) forms a nonlinear system. The Newton-Raphson linearization scheme is chosen for computing the solution. The iterative scheme can be obtained in a way similar to that presented in the previous section. Let us define $\mathbf{C} = (C^1, ..., C^{N_s+N_{EA}+N_{nut}})$ and $\mathbf{c} = (c^1, ..., c^{N_s+N_{EA}+N_{nut}})$, and be $\mathbf{\Lambda} = (\mathbf{\Lambda^1}, ..., \mathbf{\Lambda^{N_s+N_{EA}+N_{nut}}})$ the array containing the R.H.S. of equations (92)-(94). This system of equations is re-written in a residual form in a compact fashion as follows:

$$\mathbf{G}(\mathbf{c}^{k+1}) = \mathbf{c}^{k+1} - \tilde{\mathbf{c}}^k - (t^{k+1} - t^k)\mathbf{\Lambda_b}(\mathbf{c^{k+1}}) \tag{95}$$

Assuming that $\mathbf{G}(\mathbf{c}^{k+1} + \mathbf{c}^{k+1}) = 0$, $\mathbf{G}$ can be expanded around $\mathbf{c}^{k+1}$ and we stop at the first term:

$$\mathbf{G}(\mathbf{c}^{k+1} + \Delta\mathbf{c}^{k+1}) \approx \mathbf{G}(\mathbf{c}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{c}^{k+1})\right)\Delta\mathbf{c}^{k+1} \tag{96}$$

and we impose that

$$\mathbf{G}(\mathbf{c}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{c}^{k+1})\right)\Delta\mathbf{c}^{k+1} = \mathbf{0} \tag{97}$$

Equation (97) is revealing of the iteration scheme:

$$\begin{aligned}
\Delta\mathbf{c}^{k+1,m+1} &= -\mathbf{J}^{-1}\left(\mathbf{G}(\mathbf{c}^{k+1,m})\right)\mathbf{G}(\mathbf{c}^{k+1,m}) && (98)\\
\mathbf{c}^{k+1,m+1} &= \mathbf{c}^{k+1,m} + \Delta\mathbf{c}^{k+1,m+1} && (99)\\
\text{until} \quad & \Delta\mathbf{c}^{k+1,m+1} < \epsilon
\end{aligned}$$

where the first iteration for starting the scheme is $\mathbf{c}^{k+1,m} = \tilde{\mathbf{c}}^k$.

Recalling equation (56) and discretizing in time

$$B^i = \frac{\mathbf{C}^{k+1} - \tilde{\mathbf{C}}^{k+1}}{t^{k+1} - t^k} = -N_c\kappa A_c\frac{(\tilde{\mathbf{C}}^{k+1} - \mathbf{c}^{k+1})}{\delta} \tag{100}$$

$\mathbf{C}^{k+1}$ is updated as follows:

$$\mathbf{C}^{k+1} = \tilde{\mathbf{C}}^{k+1} - (t^{k+1} - t^k)N_c\kappa A_c\frac{(\tilde{\mathbf{C}}^{k+1} - \mathbf{c}^{k+1})}{\delta} \tag{101}$$

Microbial growth, equation (60), is integrated in time analytically at the end of each time iteration as:

$$\ln\left(\frac{N_c^{k+1}}{N_c^k}\right) = \left[\sum_{i=1}^{N^S}\mu_0^i\sum_{l=1}^{N_{EA}}\frac{c^{i,k+1}}{K^{i,l} + c^{i,k+1}}\frac{c^{l,k+1}}{K^l + c^{l,k+1}}\prod_{j=1}^{N_{nut}^i}\frac{c^{j,k+1}}{K^{j,l} + c^{j,k+1}} - k_d\right]\left(t^{k+1} - t^k\right) \tag{102}$$

4.2.3 Biofilm model

The numerical solution of the biofilm model is similar to that of FB and MC models. Equations (61), (62), (63), (68), and (69) are discretized in time using backward Euler implicit time stepping scheme:

$$\frac{C^{S,k+1} - \tilde{C}^{S,k+1}}{t^{k+1} - t^k} = -C^{X,k+1}Y_S r_{bio}^{k+1} + E_{l,b}^{S,k+1} \tag{103}$$

$$\frac{C^{O,k+1} - \tilde{C}^{O,k+1}}{t^{k+1} - t^k} = -C^{X,k+1} Y_O r_{bio}^{k+1} + E_{l,b}^{O,k+1} \tag{104}$$

$$\frac{C^{X,k+1} - \tilde{C}^{X,k+1}}{t^{k+1} - t^k} = C^{X,k+1} r_{bio}^{k+1} + E_{l,b}^{X,k+1} + k_d C^{X,k+1} \tag{105}$$

$$\frac{\phi_b^k \left(c_b^{S,k+1} - c_b^{S,k}\right)}{\Delta t} = -\phi_b^k Y_S c_b^X r_{bio,b} + \phi^k E_{l,b}^{S,k+1} \tag{106}$$

$$\frac{\phi_b^k \left(c_b^{O,k+1} - c_b^{O,k}\right)}{\Delta t} = -\phi_b^k Y_O c_b^X r_{bio,b} + \phi^k E_{l,b}^{O,k+1} \tag{107}$$

where, according to equations (65), (66), (67), (64), and (64), the following formulas are considered:

$$E_{l,b}^{S,k+1} = \kappa_b \left[K_b C^{S,k+1} - c_b^{S,k+1}\right] \tag{108}$$

$$E_{l,b}^{O,k+1} = \kappa_b \left[K_b C^{O,k+1} - c_b^{O,k+1}\right] \tag{109}$$

$$E_{l,b}^{X,k+1} = \kappa_b \left[K_b C^{X,k+1} - c_b^{X,k+1}\right] \tag{110}$$

$$r_{bio}^{k+1} = \mu_0 \frac{C^{S,k+1}}{K^S + C^{S,k+1}} \frac{C^{O,k+1}}{K^O + C^{O,k+1}} \tag{111}$$

$$r_{bio,b}^{k+1} = \mu_0 \frac{c_b^{S,k+1}}{K^S + c_b^{S,k+1}} \frac{c_b^{O,k+1}}{K^O + c_b^{O,k+1}} \tag{112}$$

The equations (103) through (107) constitute the nonlinear system that is solved using Newton-Raphson algorithm. Let us define $\mathbf{C} = (C^S, C^O, C^X, c_b^S, c_b^O)$ and be $\mathbf{\Lambda_b} = (\mathbf{\Lambda}^{\mathbf{C^S_b}}, \mathbf{\Lambda}^{\mathbf{C^O_b}}, \mathbf{\Lambda}^{\mathbf{C^X}}, \mathbf{\Lambda}^{\mathbf{c^S_b}}, \mathbf{\Lambda}$ the array containing the R.H.S. of equations (103)—(107). This system of equations can be re-written in a residual form in a compact fashion as follows:

$$\mathbf{G}(\mathbf{C}^{k+1}) = \mathbf{C}^{k+1} - \tilde{\mathbf{C}}^k - (t^{k+1} - t^k)\mathbf{\Lambda_b}(\mathbf{C^{k+1}}) \tag{113}$$

Assuming that $\mathbf{G}(\mathbf{C}^{k+1} + \mathbf{C}^{k+1}) = 0$, $\mathbf{G}$ can be expanded around $\mathbf{C}^{k+1}$ stopping at the first term:

$$\mathbf{G}(\mathbf{C}^{k+1} + \Delta\mathbf{C}^{k+1}) \approx \mathbf{G}(\mathbf{C}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{C}^{k+1})\right)\Delta\mathbf{C}^{k+1} \tag{114}$$

and we impose that

$$\mathbf{G}(\mathbf{C}^{k+1}) + \mathbf{J}\left(\mathbf{G}(\mathbf{C}^{k+1})\right)\Delta\mathbf{C}^{k+1} = \mathbf{0} \tag{115}$$

Equation (115) is revealing of the iteration scheme:

$$\Delta\mathbf{C}^{k+1,m+1} = -\mathbf{J}^{-1}\left(\mathbf{G}(\mathbf{C}^{k+1,m})\right)\mathbf{G}(\mathbf{C}^{k+1,m}) \tag{116}$$

$$\mathbf{C}^{k+1,m+1} = \mathbf{C}^{k+1,m} + \Delta\mathbf{C}^{k+1,m+1} \tag{117}$$

$$\text{until} \quad \Delta\mathbf{C}^{k+1,m+1} < \epsilon$$

where the first iteration for starting the scheme is $\mathbf{C}^{k+1,m} = (\tilde{C}^{S,k+1}, \tilde{C}^{O,k+1}, \tilde{C}^{X,k+1}, c_b^{S,k}, c_b^{O,k})$.

Variation of biofilm volume ratio are then calculated by discretizing in time equation (73) as follows

$$\frac{\phi_b^{k+1} - \phi_b^k}{t^{k+1} - t^k} = \frac{\phi_b^{k+1} c_b^X r_{bio,b}^{k+1} - \phi_b^{k+1} c_b^X k_d \ + \phi E_{l,b}^{X,k+1}}{c_b^X} \tag{118}$$

Porosity is updated to equation (72) which is discretized in time ($t^{k+1} - t^k$ vanished in both terms of equation 119):

$$\phi^{k+1} = \phi^k - (\phi_b^{k+1} - \phi_b^k) \tag{119}$$

Variation of permeability is computed after the completion of the nonlinear iteration and are thus coupled in a weakly fashion with biodegradation:

$$\frac{|\mathbf{K}|^{\mathbf{k+1}}}{|\mathbf{K}|^{\mathbf{k}}} = \left(\frac{\phi^{k+1}}{\phi^k}\right)^3 \Big/ \left(\frac{1-\phi^{k+1}}{1-\phi^k}\right)^2 \tag{120}$$

4.2.4 Metacode

In order to give a visual appearance of how the biomodule works, a piece of metacode is given:

***
*** biomodule is invoked
***
start
foreach node/cell

> *setup* local variables from global ones
> **start** biomodule
> **loop** [local-bio] **until** convergence
>
> > *compute* functional values
> > *build* Jacobian matrix
> > *compute* R.H.S.
> > *solve* linear system (Gauss-Jordan)
> > *check:* $||C^{i,t+\Delta t,m+1} - C^{i,t+\Delta t,m}||_2 < \epsilon_{bio}$
> > **if** $N_{iter} \geq N_{iter,max}$ **then**
> >
> > > *print* "Convergence not achieved"
> > > **stop** code
> >
> > **end if**
>
> **if** model=FB **then**
>
> > *update* $C^{X,k} \rightarrow C^{X,k+1}$
>
> **else if** model=MC **then**
>
> > *update* $\tilde{\mathbf{C}}^{\mathbf{k+1}} \rightarrow \mathbf{C}^{\mathbf{k+1}}$
> > *update* $N_c^k \rightarrow N_c^{k+1}$ (microcolonies)
>
> **else if** model=BF **then**
>
> > *update* $\phi_b^k \rightarrow \phi_b^{k+1}$ (biofilm volume ratio)
>
> **end if**
> *update* global variables from local ones
> **end loop**

5 Comments on the code coupling and interfacing

5.1 Coupling different modules

The 2FB and the modified version of the STOMP code are coupled with the biomodule which solves biodegradation. Two different coupling strategies have been chosen for the two codes. In the case of the 2FB code the coupling between the flow-solver and the biomodule has been implemented in a strong fashion. The biomodule is invoked just after computing the advection/dispersion contributions and is inserted in an iterative loop which strengthens the coupling by means of a block iteration between the various components. The same has not been done for the STOMP modification. In the STOMP simulator the computation the biomodule is invoked after computing advection/dispersion and no iteration is performed between the two components. This is aimed at reducing the computational complexity that for the STOMP code is very high, although an approximate solution is expected.

5.2 Stabilization procedures

Biodegradation computation in some cases needed a stabilization procedure, due to non-convergence of the Newton-Raphson linearization procedure. A number of experiments (not presented) showed that this was related to the magnitude of the timestep that caused the iteration convergence to fail. The solution to this problem was found in solving biodegradation using a fractional time-stepping procedure. This implies that if $\Delta t = N_{step} \times \Delta t_{bio}$, the biomodule is run N_{step} times using a time step Δt_{bio}. This ensure that, on the one hand, the procedure will converge and, on the other hand, that at the end of the biodegradation step, time will be consistent in the computation.

5.3 Interfacing the modules

For both codes, the use of the biomodule is strongly dependent on a couple of "interface" subroutines. These are invoked at the beginning and at the end of the biomodule and are aimed at interfacing local with global variables and viceversa. The biomodule solves biodegradation at each node/cell of the domain, but for only one node/cell at the time. Also, the biomodule works with mass concentration, while, in the case of the STOMP code, dissolved oil is referred in terms of mass-fraction. Further, the possibility of using a different time stepping algorithm (as mentioned in the previous paragraph) relies on a proper interface. These routines also keep track of zeroing variables and to retrieve the values of variables which are not used in the calling module (either 2FB or STOMP), such as concentration within microcolonies, biofilm volume fraction, and concentration within biofilm.

5.4 Units of measure

Another issue that should be regarded carefully is related with the units of measure. The 2FB simulator uses whatever units the user inputs. The only requirement is that they are consistent in the whole. The STOMP code can be input with almost any possible unit and makes automatically checks and conversions in the S.I. system. Due to the complexity of the STOMP source it was chosen not to extend this procedure to the biomodule. Thus, all the input regarding the biomodule **must be made in S.I.** since STOMP makes all the computations using the S.I. system and interface routines are conceived to preserve it.

5.5 On the computational costs

The computational cost of biodegradation is quite different for the 2FB and the modified STOMP simulator. In the case of 2FB, the biomodule has a large impact on the global computation both because it is in a iteration block and can affect convergence, and because the convergence is generally quite fast in the case of one-dimensional two-phase flow. A number of numerical experiments (see *Gallo and Manzini* 2000) was performed and confirmed this fact. The situation is quite different in the case of the STOMP simulator. Given the complexity of the phenomena accounted for, such as the presence of three phases, the use of different saturation and permeability curves, and, above all, the presence of multiple dimensions, the convergence of the global Newton-Raphson iteration (see section 3.2) is more difficult and time demanding. In this case, biodegradation is a minor part of the computation and does really affect the global CPU time in terms of computation.

With regards to memory requirements, inserting the biomodule in an available simulator does not require a particular effort. Apart of the array variables containing the concentrations in microcolonies and/or biofilm no other relevant memory allocations are needed.

Acknowledgements The financial support of this work was provided by the Italian Ministry of the University, Project ISR8, C11-B.

References

[1] K. Aziz and A. Settari. *"Petroleum Reservoir Simulation"*. Applied Sciences Publishers, London, 1979.

[2] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport phenomena.* Jon Wiley and Sons Inc., New York, 1960.

[3] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer Verlag, Berlin, 1992.

[4] R. E. Ewing and S. Weekes. Numerical methods for contaminant transport in porous media. *Computational Mathematics*, 202:75–95, 1998.

[5] V.S. Tripathy G. T. Yeh. A critical evaluation of recent developments in hydrogeological transport models of reactive multichemical components. *Water Resources Research*, 25(1):93–108, 1989.

[6] C. Gallo and G. Manzini. A mixed finite element/finite volume approach for solving biodegradation transport in groundwater. *International Journal of Numerical Methods in Fluids*, 26:533–556, 1998.

[7] C. Gallo and G. Manzini. 2-d numerical modeling of bioremediation in heterogeneous saturated soils. *Transport in Porous Media*, 31:67–88, 1998b.

[8] C. Gallo and G. Manzini. Numerical experiments on two-phase flow with contaminant transport and biodegradation. *Submitted to: "Communication of Numerical Methods in Engineering*, 2000.

[9] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high-order accurate essentially nonoscillatory schemes, iii. *Journal of Computational Physics*, 71:231–303, 1987.

[10] A. Harten, S. Osher, B. Engquist, and S. R. Chakravarthy. Some results on uniformly high-order accurate essentially nonoscillatory schemes. *Applied Numerical Mathematics 2*, pages 347–376, 1986.

[11] S. M. Hassanizadeh. Upscaling equations of solute transport and biodegradation in soils. Technical report, Dept. of Civil Engineering and Geosciences, TUDelft, The Netherlands, 1999.

[12] S. M. Hassanizadeh and W. G. Gray. Basic equations of flow and transport in porous media. In S. K. Sikdar and R. Irvine, editors, *BIOREMEDIATION: Principles and Practice. Vol. 1: Fundamentals and Applications*, pages 19–57. Technomic Publishing Co., Inc, Lancaster, Basel, 1998.

[13] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow.* Hemisphere Publishing Corporation, Washington D.C., 1980.

[14] K. Pruess. Tough2, a general-purpose numerical simulator for multiphase fluid and heat flow. Technical Report LBL-29400, Lawrence Berkeley Laboratory, Berkeley, CA, May 1991, 1991.

[15] Brooks R.H. and A.T. Corey. Hydraulic properties of porous media. hydrology paper 3. Technical report, Colorado State University, Fort Collins, Colorado, 1964.

[16] M.D. White and M. Oostrom. Stomp subsurface transport over multiple phase: Theory guide. Technical Report PNNL-11216 (UC-2010), Pacific Northwest National Laboratory, Richland, Washington, 1996a.