Accounting robustly for instantaneous chemical equilibriums in reactive transport: A numerical method and its application to liquid-liquid extraction modeling

Ludovic Métivier, Hervé Roussel

Accounting robustly for instantaneous chemical equilibriums in reactive transport: A numerical method and its application to liquid-liquid extraction modeling.

Ludovic Métivier[1,*], Hervé Roussel[2]

CEA Saclay, Gif-sur-Yvette, France

Abstract

Reactive transport equations are used in numerous application fields: CO_2 or nuclear waste storage monitoring, separation process in chemical engineering. We present a general method to account robustly for instantaneous equilibriums in reactive transport. This method is adapted to all kind of hydraulic transport models including 1D to 3D convection-diffusion equations. This leads to the resolution of a bound constrained system of Differential Algebraic Equations (DAE). The algebraic constraints come from the adjunction of mass action laws related to the equilibriums, whereas the bounds account for the positivity of the computed quantities. In order to solve the numerical system associated with our method, we use an adaptation of the DASSL solver, CDASSL, that can handle the resolution of bound constrained DAE systems. We present an application of this method to liquid-liquid extraction modeling. Numerical experiments demonstrate the interest of using the CDASSL solver to ensure the bound constraints are satisfied.

Keywords: reactive transport, invariant method, liquid-liquid extraction, mass transfer, instantaneous chemical equilibriums, method of lines, DAE, bound constraints

[*] Corresponding author

Email address: `ludovic.metivier@ujf-grenoble.fr` (Ludovic Métivier)

[1] CEA, DEN, DM2S/SFME/LSET, F-91191 Gif-sur-Yvette, France.

[2] CEA, DEN, DRCP/SCPS/LMPR, F-30207 Bagnols-sur-Céze, France.

1. Introduction

Reactive transport is a physical phenomenon occurring in various industrial fields such as the modeling of CO_2 or nuclear waste storage, or industrial separation process such as distillation, absorption, gas-liquid-extraction, and liquid-liquid extraction to name a few. Modeling this phenomenon involves the description of the transport of a multiphasic flow coupled to monophasic chemical kinetic reactions and equilibriums, and mass transfer between each phase.

Given the properties of the flow, the problem amounts to computing the concentrations of the transported chemical species. Retroaction of the chemical composition of the flow on the transport properties (such as the velocity of the phases or the diffusion operator) can be accounted for.

A common feature of reactive transport is the discrepancy between the characteristic times of the physical transport and the chemical reactions, or between the characteristic times of the different chemical reactions. This discrepancy results in numerical "stiffness" of the systems studied, that make it difficult to compute accurate solutions to the reactive transport equation. The problem of interest here is the additional possibility of accounting for "instantaneous" chemical equilibriums.

Here, *instantaneous* has to be carefully defined. We obviously do not consider that the velocity of the exchanges of atoms between molecules described by this kind of chemical phenomenons is infinite. However, we consider that the characteristic time of this exchange is shorter than the characteristic times of all other physical or chemical phenomena involved in the process, by several order of magnitude. A classical example of this type of phenomena are the acide-base equilibriums. To simplify the following discussion, the phrase "instantaneous chemical equilibriums" is shortened to "chemical equilibriums" or "equilibriums".

Accounting for equilibriums is not a simple task. Consider a monophasic volume containing three species AH, H^+, A^- in solution, related by a simple acide-base equilibrium, described by

$$AH \leftrightharpoons A^- + H^+ \tag{1}$$

and the mass action law

$$\frac{[H^+][A^-]}{[AH]} = K_a, \quad K_a \in \mathbb{R}^+. \tag{2}$$

Here, as in the following, [.] denotes the concentrations of species.

A first possibility for computing the concentrations of A^-, AH and H^+ from an initial condition of the system at time t_0 is to use the following three equations

- the mass action law;
- the conservation of $[A]$: $[A]^- + [AH] = [A]^-(t_0) + [AH](t_0)$;
- the conservation of $[H]$: $[AH] + [H^+] = [AH](t_0) + [H^+](t_0)$.

If instead of three species the chemical system is defined with hundreds of species and involves tens of chemical equilibriums, this method becomes difficult to apply without a proper generalization. In addition, if these equilibriums are defined not in a monophasic volume, but in a complex reactive transport scheme, the question of how to integrate these equations into the system remains open.

Another way to account for this equilibrium is to describe it by two opposite reactions

$$A^- + H^+ \rightarrow AH, \qquad AH \rightarrow A^- + H^+, \tag{3}$$

with infinite reaction rates. The benefit of this method is that it associates every equilibriums with two reactions that can be incorporated directly into the reactive transport equations. Thus it is simple to use, and simple to implement. However, it tremendously increases the stiffness of the problem. Indeed, in order to mimic infinite reaction rates, the rates associated to the two opposite reaction have to be multiplied by an extremely large constant.

The method we propose is a generalization of the first method, which we call the invariants method. By considering the algebraic properties of the matrix that defines the chemical equilibriums, we deduce an automatic way of computing invariant quantities such as the total quantity of A or the total quantity of H in the previous example. The reactive transport equations are based on these quantities, and the resulting system is completed with the mass actions laws associated with each equilibrium. Thus the method does not degrade the condition number of the system by incorporating artificial infinite reaction rates.

In order to adapt the method to all kind of transport equations (from the simplest convective case to nonlinear diffusion-convection operators), it is developed in the context of global resolution of the reactive transport equations.

Global resolution uses the method of lines [Schiesser (1991)], which, in the context of Partial Differential Equation (PDE), amounts to semi-discretize the continuous equations in space. The resulting Ordinary Differential Equations (ODE) system is then solved using implicit or semi-implicit methods in time, able to deal with stiff ODE systems. This global strategy has been shown to be more accurate and efficient than explicit coupling between transport and chemical reaction [De Dieuleveut et al. (2009)].

Here, the law mass actions introduced in the system by the invariant method take the form of algebraic constraints. Therefore, the system of equations obtained after semi-discretization in space is a Differential Algebraic Equations (DAE) system rather than an ODE system. The numerical resolution of these systems has been studied since the end of the 1970s, together with the numerical solution of stiff ODE systems. The main idea is to discretize the time derivatives using a fully implicit scheme. This yields a global nonlinear system, including the discretized ODE *and* the algebraic constraints. This nonlinear system is solved using Newton iterations. This strategy was initially proposed by Gear [Gear et al. (1984)]. Numerous solvers have been proposed, based on different implicit discretization methods: Backward Differentiation Methods (BDF) [Hairer et al. (1996)] and Modified Extended BDF (MEBDF) [Cash (2003)], or Runge Kutta methods [Hairer et al. (1996)]. Corresponding solvers are DASSL [Brenan et al. (1995)], MEBDF-DAE [Psihoyios (2007)], or RADAU5 [Hairer et al. (1996)].

In addition, the system raised by the invariant method involves bound constraints. Indeed, the quantities that are computed are the number of moles and/or concentrations, which must remain positive. This aspect of the resulting numerical problem is often eluded. In DASSL for instance, an option is provided to enforce non-negativity of the computed solution, but the method used is clearly non-robust, as it is presented in Section 5.3. We would like to emphasize here the importance of satisfying robustly bound constraints: if not, the physical laws involved in the process such as reaction rates or mass action law *could be impossible to evaluate*, which would lead to a critical failure in the numerical resolution of the equations

Classical DAE solvers do not offer the possibility to robustly handle bound constraints. As a consequence, we use in this study a new DAE solver accounting for bound constraints. This solver, named CDASSL [Métivier et al. (2011)], is based on the time discretization algorithm defined in the DASSL method [Brenan et al. (1995)], coupled with the interior point Newton method CODOSOL, [Bellavia et al. (2011)].

In order to illustrate the efficiency of our method, we apply the invariant method to the modeling of a separation process known as liquid-liquid extraction. Based on this application, a numerical test case is defined, which is solved using alternatively DASSL and CDASSL. The comparison of the results provided by these two solvers demonstrate the importance of accounting for bound constraints in the numerical resolution of the equations associated with the liquid-liquid extraction model.

In Section 2, we give the description of the invariant method for general reactive transport equations. In Section 3, we give a brief description of CDASSL, the bound constrained DAE solver designed for the numerical resolution of bound constrained DAE systems. In Section 4, we present an application of the invariant method to the modeling of liquid-liquid extraction. In Section 5, numerical results on this specific case are presented, that emphasize the interest of using the bound constrained DAE solver CDASSL. A conclusion is given in Section 6.

2. The invariant method for general reactive transport equations

This section starts with the definition of the general reactive transport equations, in a continuous framework, then in the semi-discretized in space domain. Algebraic notations are introduced in order to define the kinetic reactions. The possibility of accounting for the equilibriums is then discussed. First, the method that consists in converting each equilibrium into two reactions is presented in details. The link between the reaction rates and the mass action law defining the equilibrium is given. In order to overcome the numerical difficulties induced by this method, the proposition of the invariant method is made. The resulting model is a DAE system.

2.1. Continuous model

We consider the 3D general reactive transport model. The vector of spatial coordinates is denoted by $\mathbf{x} \in \mathbb{R}^3$. The time coordinate is denoted by $t \in \mathbb{R}$.

We consider a chemical system composed of $Q \in \mathbb{N}$ phases. For $q = 1, \ldots, Q$, denote

- N^q the number of chemical species in phase q;

- $m^q(\mathbf{x},t)$ the vector valued function corresponding to the number of moles of the different species in phase q:

$$\begin{array}{rccc} m^q: & (\mathbf{x},t) & \longrightarrow & m^q(\mathbf{x},t) \\ & \mathbb{R}^3 \times \mathbb{R} & \longrightarrow & \mathbb{R}^{N^q}. \end{array} \tag{4}$$

The general reactive equations transport can be written as a system composed of q subsystems of N^q equations

$$\partial_t m^q = T^q(\partial_{x_j}, m^q) + \chi^q(m^q) + \sum_{\substack{i=1\\ i\neq q}}^{Q} M^{qi}(m^q, m^i), \quad q = 1\ldots, Q \;\; j = 1, \ldots, 3 \tag{5}$$

where, for $q = 1, \ldots, Q$

- T^q denotes the transport operator for phase q;
- χ^q denotes the monophasic chemical operator related to phase q;
- M^{qi} denotes the mass transfer operator between phase q and phase i[1].

The number of equations of this system is denoted by $\mathcal{N}$

$$\mathcal{N} = \sum_{q=1}^{Q} N^q. \tag{6}$$

The only non local operators are the transport operators T^q. The chemical operators do not involve spatial derivatives, and can be rather considered as source terms. A classical expression for the transport operator is the well-known diffusion-convection operator

$$\nabla(-D\nabla m + vm) \tag{7}$$

where D is the diffusion operator and v the velocity of the fluid. However, here, we do not make any assumption about the form of this operator. Instead, we propose to solve the problem using the method of lines [Schiesser (1991)]. This amounts to semi-discretize the equations (5) in space, which comes back to the definition of an order one in time ordinary differential system. This yields the possibility of considering all types of transport operator, possibly nonlinear, and all kind of discretization method for these operators.

[1] Note that in principle $M^{qi} \neq M^{iq}$.

2.2. Semi-discretization in space

Denote by P the total number of elementary volumes defined by the discretization in space. The system of equations (5) is rewritten as a $P\mathcal{N}$ equations system, decomposed into QP subsystems of dimension N^q

$$\partial_t m_p^q = \mathcal{T}_p^q(m_1^q, \dots, m_P^q) + \chi^q(m_p^q) + \sum_{\substack{i=1\\ i\neq q}}^{Q} M^{qi}(m_p^q, m_p^i), \;\; q = 1, \dots, Q. \quad (8)$$

Indeed, the discretization implies one subsystem of N^q equations by phase q and volume p.

In equation (8), $\mathcal{T}_p^q$, $q = 0, \dots, Q$ denote the discretized in space transport operators. Note that these operators depend on the volume p where they are defined: the mass contribution from the other volumes to the mass balance equation p for the phase q depend on the considered volume p. Note also that these operators depend not only on the continuous transport operator chosen to describe the chemical process, but also on the spatial discretization selected.

2.3. Monophasic chemical reactions

We first consider that no equilibrium is defined. The chemical operators χ^q thus only stand for the monophasic kinetic reactions.

We introduce algebraic notations for kinetic reactions. Each phase $q = 1, \dots, Q$ contains $N^q \in \mathbb{N}$ species denoted by $A^q = (A_1^q, \dots, A_{N_q}^q)^T$. A set of $R_{kin}^q \in \mathbb{N}$ kinetic reactions is defined for each phase:

$$\sum_{j=1}^{N^q} (\alpha_{kin})_j^r A_j^q \to \sum_{j=1}^{N^q} (\beta_{kin})_j^r A_j^q, \;\; r = 1, \dots, R_{kin}^q. \quad (9)$$

These sets of reactions are rewritten under the algebraic form

$$S_{kin}^q A^q = 0, \text{ where} \quad (10)$$

$$S_{kin}^q \in \mathbb{M}_{R_{kin}^q, N^q}(\mathbb{R}), \;\; (S_{kin}^q)_{rj} = (\beta_{kin})_j^r - (\alpha_{kin})_j^r, \quad r = 1, \dots, R_{kin}^q, \quad (11)$$

To all these equations corresponds a reaction rate. These reaction rates are nonlinear functions of the species concentrations. Let $[A_i^q]$ denote the concentration of species A_i^q in phase q, and

$$[A]^q = ([A_1^q], \dots, [A_{N^q}^q])^T \quad (12)$$

The ith reaction rate is denoted by

$$v_i^q\left([A]^q]\right). \tag{13}$$

Let v^q denote the vector of reaction rates for the phase q:

$$v^q([A^q]) = (v_1([A^q]), \ldots, v_{R_{kin}}([A^q]))^T \tag{14}$$

The chemical operators χ^q are then rewritten

$$\chi^q(x^q) = \left(S_{kin}^q\right)^T v_{kin}^q([A^q]) \tag{15}$$

This yields the following system for reactive transport without chemical equilibriums:

$$\partial_t m_p^q = \mathcal{T}_p^q(m_1^q, \ldots, m_P^q) + (S_{kin}^q)^T v_{kin}^q([A]_p^q) + \sum_{\substack{i=1\\ i\neq q}}^{Q} M^{qi}(m_p^q, m_p^i) \tag{16}$$

This first system is general, in the sense that any transport operator and any discretization of this operator can be handled, and any mass transfer operator can be accounted for. In the next subsection, we introduce the invariant method to account robustly for equilibriums inside each phase.

2.4. Accounting robustly for chemical equilibriums

2.4.1. First possibility: defining two opposite kinetic reactions

As mentioned in the introduction, a simple method to account for equilibriums consists in converting them into couples of opposite reactions. Let us begin with the following example.

Example 1. *Consider* 4 *species* A, B, C, D *related by the simple equilibrium*

$$A + B \leftrightharpoons C + D \tag{17}$$

The corresponding classical first order mass action law is

$$\frac{[C][D]}{[A][B]} - K = 0 \tag{18}$$

where K *is the equilibrium constant.*

As for the kinetic reactions, a set of $R^q_{eq} \in \mathbb{N}$ equilibriums is defined for each phase, $q = 0, \dots, Q$

$$\sum_{j=1}^{N^q} (\alpha_{eq})^r_j A^q_j \rightleftharpoons \sum_{j=1}^{N^q} (\beta_{eq})^r_j A^q_j, \quad r = 1, \dots, R^q_{kin}. \tag{19}$$

These sets of equilibriums are rewritten in algebraic form

$$S^q_{eq} A^q = 0, \text{ where} \tag{20}$$

$$S^q_{eq} \in \mathbb{M}_{R^q_{eq}, N^q}(\mathbb{R}), \quad (S^q_{kin})_{rj} = (\beta_{eq})^r_j - (\alpha_{eq})^r_j, \quad r = 1, \dots, R^q_{eq}, \tag{21}$$

For each equilibrium a mass action law is defined

$$g^q_r([A]^q), \quad r = 1, \dots, R_{eq} \tag{22}$$

Any equilibrium can be reduced to two opposite reactions.

Example 2. *The equilibrium in example 1 is equivalent to*

$$\left\{ \begin{array}{lcll} A + B & \longrightarrow & C + D & (1) \\ C + D & \longrightarrow & A + B & (2) \end{array} \right. \tag{23}$$

Using the general form of the equilibriums, this amounts to introducing $2R^q_{eq}$ kinetic reactions for each phase q, associated with the stoichiometric matrix S^q_{eq} and $-S^q_{eq}$ and the reaction rates ν^q_1 and $\nu^q_2 \in \mathbb{R}^{N^q_{eq}} \times \mathbb{R}^{N^q_{eq}}$. At equilibrium, $\nu^q_1 = \nu^q_2$ whereas, when the equilibrium is perturbed, $\nu^q_1 \neq \nu^q_2$. Define $\nu^q = \nu^q_1 - \nu^q_2$. Choosing ν^q_1 and ν^q_2 such that

$$\nu^q = (g^q_1([A]^q), \dots, g^q_{R^q_{eq}}([A]^q))^T \tag{24}$$

yields a first model accounting for the equilibriums.

$$\begin{array}{rcccccc} \partial_t m^q_p & = & \mathcal{T}^q_p(m^q_1, \dots, m^q_P) & + & (S^q_{kin})^T v^q_{kin}([A]^q_p) & + & \\ & & (S^q_{eq})^T \nu^q([A]^q_p) & + & \displaystyle\sum_{\substack{i=1 \\ i \neq q}}^{Q} M^{qi}(m^q_p, m^i_p) & & \end{array} \tag{25}$$

The equilibriums are assumed to be instantaneous. When the equilibrium is perturbed, it should be brought back to its steady state instantaneously. To mimic this notion of infinite speed, the "pseudo-reaction rates" ν^q can be multiplied by a large constant κ. This is nonetheless unsatisfactory:

- it introduces non necessary stiffness in the system, possibly resulting in numerical difficulties;
- even for large values of κ the transient mode is perturbed by the introduction of artificial kinetic reactions.

As a consequence, we propose another method that we have named the "invariant method", in order to account more robustly for equilibriums.

2.4.2. The invariant method

Consider the stoichiometric coefficient matrix $(S_{eq}^q)^T \in \mathbb{M}_{N^q,R_{eq}^q}$. Assume that the equilibriums defined for the phase q are linearly independent[1]. Then,

$$R_{eq}^q < N^q, \text{ and } \operatorname{rank}(S_{eq}^q)^T = R_{eq}^q \tag{26}$$

Hence, there exists a matrix $L^q \in \mathbb{M}_{N^q,N^q}$ such that $L^q(S_{eq}^q)^T$ is a row echelon matrix, with R_{eq}^q non zero lines and $N^q - R_{eq}^q$ zero lines:

$$L^q(S_{eq}^q)^T = \begin{pmatrix} 1 & * & & & & & \\ 0 & \dots & 0 & 1 & * & & \\ \vdots & & \vdots & \vdots & & & \\ 0 & \dots & 0 & 0 & \dots & 1 & * \\ 0 & \dots & \dots & \dots & \dots & 0 & \\ \vdots & & & & & \vdots & \\ 0 & \dots & \dots & \dots & \dots & 0 & \end{pmatrix}, \tag{27}$$

The matrix L^q can be obtained through the Gaussian elimination algorithm [Gollub et al. (1996)]. Define

$$N_{inv}^q = N^q - R_{eq}^q, \quad U^q = \begin{bmatrix} 0_{N_{inv}^q,R_{eq}^q} & I_{N_{inv}^q} \end{bmatrix} \in \mathbb{M}_{N_{inv}^q,N^q} \tag{28}$$

[1] This is a reasonable assumption. If it is not the case, the chemical system is ill-posed.

where $I_{N^q_{inv}}$ is the identity matrix of dimension N^q_{inv}. Define

$$\widetilde{L}^q = U^q L^q \in \mathbb{M}_{N^q_{inv},N^q} \tag{29}$$

The matrix $\widetilde{L}$ give the composition of N^q_{inv} chemical invariants inside phase q.

Example 3. *Consider again the equilibrium*

$$A + B \leftrightharpoons C + D \tag{30}$$

in a given phase, where A, B, C, D are the only chemical species. The matrix S_{eq} associated with the equilibrium (30) is

$$S_{eq} = \begin{pmatrix} -1 & -1 & 1 & 1 \end{pmatrix} \tag{31}$$

assuming the species are ordered alphabetically.

Then, matrix L is

$$L = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 \\ -1 & 0 & -1 & 0 \\ -1 & 0 & 0 & -1 \end{pmatrix}, \quad LS_{eq}^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \textit{and} \quad \widetilde{L} = \begin{pmatrix} 0 & -1 & 0 & -1 \\ -1 & 0 & -1 & 0 \\ -1 & 0 & 0 & -1 \end{pmatrix}, \tag{32}$$

This yields 3 *invariants quantities*

$$u_1 = -[B] - [D], \ \ u_2 = -[A] - [C], \ \ u_3 = -[A] - [D] \tag{33}$$

Multiplying the equations of system (25) by the matrix $\widetilde{L}^q$, we obtain the system of equations

$$\widetilde{L}^q \partial_t m^q_p = \widetilde{L}^q \mathcal{T}^q_p(m^q_1, \dots, m^q_P) + \widetilde{L}^q (S^q_{kin})^T v^q_{kin}([A]^q_p) + \widetilde{L}^q \sum_{\substack{i=1 \\ i \neq q}}^{Q} \mathcal{M}^{qi}(m^q_p, m^i_p) \tag{34}$$

The terms related to the equilibriums are cancelled by muliplication of the equations by $\widetilde{L}^q$. This yields a system of size

$$P \times \left(\sum_{q=1}^{Q} N^q_{inv} \right) \ \text{ equations for } \ P \times \mathcal{N} \ \text{ unknowns} \tag{35}$$

The $P \times \left(\sum_{q=1}^{Q} N_{eq}^q \right)$ missing equations are given by the mass action laws related to each equilibrium. We finally obtain the complete system

$$\left\{ \begin{array}{lcl} \widetilde{L}^q \partial_t m_p^q & = & \widetilde{L}^q \mathcal{T}_p^q(m_1^q, \ldots, m_P^q) + \widetilde{L}^q (S_{kin}^q)^T v_{kin}^q([A]_p^q) + \widetilde{L}^q \sum_{\substack{i=1 \\ i \neq q}}^{Q} \mathcal{M}^{qi}(m_p^q, m_p^i) \\ 0 & = & g^q([A]_p^q) \end{array} \right. \tag{36}$$

where $g^q = (g_1^q, \ldots, g_{N_{eq}^q})^T$.

This method can be summarized as follows. Rather than defining mass balance equations for each species, we compute linear combinations of species that are kept unchanged by the equilibriums. Assuming linear independency of the equilibriums, we can define at most N_{inv}^q invariant for each phase q. Therefore, we have to complete the system by adding N_{eq}^q equations for each phase q. These equations are given by the mass action laws defined for each equilibriums. We end up with a complete system that does not involve fictitious infinitely fast reaction rates to account for equilibriums. Instead, it includes algebraic constraints corresponding to the mass action laws. This algebraic method, (the invariant method) is thus very efficient, provided we can solve bound constrained DAE systems. We give in the next section the description of the CDASSL solver, designed to solve this kind of problems.

3. CDASSL: a bound constrained DAE solver

As a DAE system, (36) can be rewritten in a general form

$$G(t, y, y') = 0 \tag{37}$$

where

$$\left\{ \begin{array}{ccc} y : t & \longrightarrow & y(t) \\ \mathbb{R} & \longrightarrow & \mathbb{R}^{P\mathcal{N}} \end{array} \right. \tag{38}$$

stands for the number of moles m_p^q. The function $y'(t)$ denotes the time derivatives of $y(t)$.

The solver CDASSL is designed for the resolution of the following problem

$$\left\{ \begin{array}{l} G(t, y, y') = 0, \ t \in [0; T], T \in \mathbb{R}_+, \\ y(0) = y_0 \\ \forall t \in [0 \ T], \ l \leq y(t) \leq u, \ (l, u) \in \mathbb{R}^{P\mathcal{N}} \times \mathbb{R}^{P\mathcal{N}} \end{array} \right. \tag{39}$$

This solver is based on the DASSL solver [Brenan et al. (1995)], designed to solve index 1 DAE systems without bound constraints. DASSL uses an implicit Backward Differentiation Formula (BDF) discretization method [Hairer et al. (1996)], under its variable stepsize and order formulation, together with a very efficient stepsize and order selection algorithm. This is kept unchanged in CDASSL. However, while DASSL uses a simple modified Newton algorithm to solve the nonlinear system that arises at each iteration, CDASSL implements a more sophisticated Newton method CODOSOL [Bellavia et al. (2011)], based on a trust-region globalization method, and accounting for bound constraints. This yields an efficient bound constrained DAE solver, which is presented briefly in the two following subsections. The reader is referred to [Métivier et al. (2011)] for a complete description of the algorithm, where the strategy used in CDASSL is compared with other possibilities for enforcing non-negativiy constraints. CDASSL has the advantage of implementing general bound constraints, and is therefore more general, even if in the liquid-liquid extraction application we present in section 4 and 5, only non-negativity constraints are used.

3.1. General principle

The BDF discretization method is based on a prediction/correction implementation. Let t_n denote the time reached at iteration n and h_{n+1} the current time step such that

$$h_{n+1} = t_{n+1} - t_n \tag{40}$$

At time t_n, the method aims at computing an approximation of $y(t_{n+1})$, the solution at time t_{n+1}. This approximation is denoted by y_{n+1}.

Let $k \in \mathbb{N}$ denote the order chosen for computing this approximation[1]. A prediction of y_{n+1} is first computed with a predictor polynomial $w^P(t)$. This polynomial is given by the interpolation of the $k+1$ previous available approximations $y_{n-i},\ i = 0, \dots, k$,

$$w^P(t_{n-i}) = y_{n-i}, \ \ i = 0, \dots, k. \tag{41}$$

Let $y_{n+1}^{(0)}$ and ${y'}_{n+1}^{(0)}$ denote the prediction of the solution and its time derivatives. They are defined by

$$y_{n+1}^{(0)} = w^P(t_{n+1}), \ \ {y'}_{n+1}^{(0)} = {w'}^P(t_{n+1}). \tag{42}$$

[1] The stability region of the BDF method implies that k must be chosen between 1 and 5 [Hairer et al. (1996)].

The use of the BDF method, under its fixed leading order coefficient form, amounts to defining a discretization of the time derivatives such that

$$y^{'}_{n+1} = y^{'(0)}_{n+1} - \frac{\alpha}{h_{n+1}}\left(y_{n+1} - y^{(0)}_{n+1}\right), \tag{43}$$

where α is the fixed leading coefficient.

Replacing $y^{'}_{n+1}$ by this expression in (39) yields the nonlinear system

$$\begin{cases} G\left(t_{n+1}, y_{n+1}, y^{'(0)}_{n+1} - \dfrac{\alpha}{h_{n+1}}\left(y_{n+1} - y^{(0)}_{n+1}\right)\right) = 0, \\ y_0 = 0 \\ l \leq y_{n+1} \leq u, \ (l,u) \in \mathbb{R}^2 \end{cases} \tag{44}$$

Using the time discretization provided by the BDF method, the resolution of the problem (39) is thus reduced to the resolution of a sequence of nonlinear systems

$$\begin{cases} F_{n+1}(y) = 0 \\ l \leq y \leq u \end{cases} \tag{45}$$

where

$$F_{n+1}(y) = G\left(t_{n+1}, y, y^{'(0)}_{n+1} - \frac{\alpha}{h_{n+1}}\left(y - y^{(0)}_{n+1}\right)\right) \tag{46}$$

3.2. Stepsize and order selection algorithm

A crucial point in the design of BDF based DAE solvers is the algorithm used to select the stepsize and the order at each iteration. A full description of this algorithm is beyond the scope of this paper.

The main idea is to first reduce the order k of the method when the solution $y(t)$ encounters fast variations. These fast variations are detected by tests on the truncation error of the BDF method. If the variations of $y(t)$ are smooth, then the truncation error should decrease as the order increases. If not the case, it is likely due to fast variations of $y(t)$. Hence, high order methods become extremely unstable, and require very small stepsize to converge. It is thus preferable to use a lower order method and a larger stepsize. Conversely, when the solution $y(t)$ is smooth, it is more valuable to use high order method, that are stable, and allow large stepsizes. This provides a very efficient strategy that aims at taking as large step as possible in any situation, to ensure fast convergence to the final integration time T.

3.3. An efficient Newton method to solve bound constrained nonlinear systems

The sequence of bound constrained nonlinear systems (45) yielded by the time discretization is solved using an interior point method named CODOSOL (Constrained Dogleg Solver) [Bellavia et al. (2011)]. Again, a full description of this algorithm is beyond the scope of the present paper, we shall only mention its basic features.

First, define

$$\Omega \subset \mathbb{R}^s,\ \Omega = \{y \in \mathbb{R}^s,\ l_i \leq y_i \leq u_i,\ i = 1, \ldots, s\}\,. \tag{47}$$

For a given time step $n+1$, define

$$f(y) = \|F_{n+1}(y)\|^2 \tag{48}$$

where $\|.\|$ is the classical L^2 norm.

The method aims in fact at computing the solution of the problem

$$\min_{y \in \Omega} f(y) \tag{49}$$

Clearly, if (45) has a solution, this is also the solution of (49).

A classical Newton procedure is set up to solve problem (49), A sequence of iterates y^d, converging to the solution $\overline{y}$ of (49), is built from an initial guess y^0, such that

$$y^{d+1} = y^d + p^d, \tag{50}$$

where p^d is the descent step taken at iteration d. Each iteration corresponds to the minimization of a local quadratic approximation of $f(y)$. The descent direction p^d is determined by the following trust-region method.

At iteration d, a trust-region of radius Δ_d is defined. This radius reflects the confidence in the current quadratic approximation. The greater Δ_d, the greater the confidence in the quadratic approximation. The following step selection procedure is then used.

- First, compute the Newton descent direction p_N^d.
- Second, project p_N^d so $y^d + p_N^d$ is in the feasible set Ω (bound constraints satisfactied). Let $\overline{p}_N^d$ denote the projected Newton step.
- If $\|\overline{p}_N^d\| \leq \Delta_d$, then the projected Newton step is selected as the descent direction and the step selection procedure ends here.

- Conversely, if $\|\overline{p}_N^d\| > \Delta_d$, then compute the steepest descent direction p_C^d, and project it as for the Newton step to produce a descent direction that remains in the feasible set Ω. Let $\overline{p}_C^d$ denote its projection.

- Compute the optimal convex combination of $\overline{p}_N^d$ and $\overline{p}_C^d$ that minimizes the current quadratic approximation of f and stay within the trust region.

- Depending on the accuracy of the quadratic approximation, update the trust-region radius: if the accuracy is good, the radius can be increased, otherwise it is decreased.

This method ensures linear global convergence and local quadratic convergence. It also ensures bound constrained satisfaction at each iteration of the procedure. Therefore, the bound constraints are permanently satisfied during the resolution of the DAE system.

3.4. CDASSL: summary of the method

The CDASSL solver is a bound constrained DAE solver implemented in the modular `C++` language. It is based on a BDF discretization scheme, with variable order and stepsize. The algorithm for the selection of the stepsize and the order at each timestep comes directly from the DASSL solver, originally designed in `Fortran 77`.

The use of the discretization scheme yields a sequence of bound constrained nonlinear systems that are solved with an appropriate bound constrained Newton solver named as CODOSOL [Bellavia et al. (2011)]. A `C++` implementation of this algorithm, coupled with the `C++` version of the DASSL discretization method and stepsize and order selection algorithm yields the CDASSL solver.

Compared to the other existing DAE solvers, CDASSL can handle bound constraints robustly. At each step of the DAE integration, the constraints are satisfied. Therefore, in case of physical laws that can not be evaluated outside bounds constraints, the CDASSL solver perform wells, whereas other DAE solvers could encounter severe difficulties. As a consequence, this solver is perfectly adapted to the resolution of the system yielded by the use of the invariant method in reactive transport problems proposed in Section 2.

4. An application of the invariant method to liquid-liquid extraction modeling

In this section, we apply the invariant method presented in Section 2 to the particular case of liquid-liquid extraction modeling. Indeed, this study was motivated in great part by the development of accurate and practicable models for this application. The use of the invariant method for modeling this particular process consists in specifying the type of the mass transfer model, and deriving the corresponding equations. We first describe the physical phenomena involved in liquid-liquid extraction. Then we give the mass transfer model and propose a first system of equations without equilibriums. Finally we introduce the equilibriums through the invariant method of Section 2.

4.1. Liquid-liquid extraction

As an illustration of the application of the invariant method to reactive transport, we consider a liquid-liquid extraction problem. This particular separation process is applied in numerous fields, such as ore processing, oil refining, or nuclear wastes recycling. It is based mainly on the transport of two immiscible liquid phases (generally, an aqueous and an organic phase) in opposite directions inside extractors. Depending on the affinities of species for one or the other phase, separation of components can be efficiently achieved, using one or more cycles of liquid-liquid extraction. An emulsion of the two immiscible phases is generated by the extractor in order to activate mass transfer between them. This emulsion is generally described in terms of the phase that remains continuous (the continuous phase) and the one that is dispersed in droplets in the continuous phase (the dispersed phase).

Here we consider a possibly more complex model, where multiple dispersed phases can be represented. This yields the possibility of accounting for dispersed phases of different natures: gas, liquid, or solid, with various chemical compositions. But this also provides the possibility to more precisely modeling the hydraulic behavior of the phases inside the extractor: for instance, the dispersed phases can be separated into several populations with regard to the mean size of the droplets, and their behavior in the extractor. In the case of a continuous extractor, the largest droplets are transported in the opposite direction of the continuous phase, while the smallest droplets may be carried away by the continuous phase.

4.2. Double layer mass transfer model

Let $Q \in \mathbb{N}$ denote the number of dispersed phases. In the sequel, the superscript 0 refers to the continuous phase, whereas the superscript $q = 1, \dots, Q$ refers to a particular dispersed phase. When no distinction is useful between the continuous and the dispersed phases, a superscript $q = 0, \dots, Q$ is used.

Given the description of the invariant method for general reactive transport equation, the definition of our liquid-liquid extraction model consists only in specifying the form of the mass transfer operators M^{qi} in equation (36).

First, we assume that the mass transfer occurs only between the continuous phase and the dispersed phase. Indeed, mass transfer requires large contact surfaces. Neglecting the contact surfaces between different dispersed phases is a reasonable approximation.

We use a double layer description of mass transfer [Lewis et al. (1924)]. This description assumes the presence of a zone located at the interface between the continuous phase and one dispersed phase (the interfacial zone). The species belonging to each phase diffuse in this zone through a film [Di Miceli Raimondi (2008)]. These two films express the resistance of the species in their passage from one phase to the other. Corresponding diffusion coefficients for each species are defined in diagonal matrices $K^q,\ q = 0, \dots, Q$, depending on the considered phase.

The interfacial zones have an infinitely small thickness. Hence, they are assimilated with surfaces where no accumulation of mass is allowed. This implies that at each time step, the quantity of matter exchanged in the interfacial zone is equal to the quantity that has diffused from the inner phases.

The corresponding surfaces areas are denoted by $\sigma^q,\ q = 1, \dots, Q$. These surfaces depend on the hydraulic properties of the dispersed phases, notably on the mean diameter of the droplets, for instance the Sauter diameter. The amount of mass exchanged between two phases is proportional to this surface area.

For $q = 1, \dots, Q$, we denote by $\hat{z}^q$ (respectively z^q) the interfacial concentration vectors of the continuous phase (respectively the dispersed phases). The mass transfer is defined in the interfacial zone, through the definition of kinetic reactions involving species of the two phases.

Example 4. *Consider a dispersed phase containing species A and a continuous phase containing two species E and AE. The transfer of A from the* *dispersed phase to the continuous phase is described by the kinetic interfacial reaction*

$$A + E \longrightarrow AE \tag{51}$$

This reaction involves species of the two phases. In this particular case, E can be defined as the extracting species. For each dispersed phase $q = 1, \ldots, Q$ a set of R^q_{tr} interfacial reaction is thus defined, under the general form

$$\sum_{j=1}^{N^q} \gamma_j^{r,q} A_j^q + \sum_{j=1}^{N^0} \hat{\gamma}_j^{r,q} A_j^0 \longrightarrow \sum_{j=1}^{N^q} \eta_j^{r,q} A_j^q + \sum_{j=1}^{N^0} \hat{\eta}_j^{r,q} A_j^0 \tag{52}$$

These equations are put into the algebraic form

$$S^q_{tr} A^q + \hat{S}^q_{tr} A^0 = 0, \tag{53}$$

with

$$\left\{ \begin{array}{l} S^q_{tr} \in \mathbb{M}_{R^q_{tr}, N^q}(\mathbb{R}), \quad (S^q_{tr})_{rj} = \gamma_j^{r,q} - \eta_j^{r,q} \\ \hat{S}^q_{tr} \in \mathbb{M}_{R^q_{tr}, N^0}(\mathbb{R}), \quad (\hat{S}^q_{tr})_{rj} = \hat{\gamma}_j^{r,q} - \hat{\eta}_j^{r,q} \end{array} \right. \tag{54}$$

As for monophasic kinetic reactions, reaction rates are defined, denoted by vectors $w^q \in \mathbb{R}^{R^q_{tr}}$ for the dispersed phase and $\hat{w}^q \in \mathbb{R}^{R^q_{tr}}$ for the continuous phase. These vectors are nonlinear function of the interfacial concentrations $\hat{z}^q$ and z^q of the continuous and the dispersed phases.

4.3. A first liquid-liquid extraction model without chemical equilibriums

Consider for now that no chemical equilibriums are defined in the system. The corresponding semi-discretized in space equations, following the notations introduced in section 2 are, for $q = 1, \ldots, Q$

$$\left\{ \begin{array}{rclclcl} \partial_t m_p^0 & = & \mathcal{T}_p^0(m_1^0, \ldots, m_P^0) & + & (S^0_{kin})^T v^0_{kin}([A]^0_p) & + & K^0 \sum_{q=1}^{Q} \left([A]^0_p - \hat{z}^q\right) \\ 0 & = & K^0(\hat{z}^q_p - [A]^0_p) & + & (\hat{S}^q_{tr})^T \sigma^q \hat{w}^q \left(\hat{z}^q_p, z^q_p\right) & & \\ \partial_t m_p^q & = & \mathcal{T}_p^q(m_1^q, \ldots, m_P^q) & + & (S^q_{kin})^T v^q_{kin}([A]^q_p) & + & K^q \left([A]^q_p - z^q_p\right), \\ 0 & = & (z^q_p - [A]^q_p) & + & (S^q_{tr})^T \sigma^q w^q \left(\hat{z}^q_p, z^q_p\right) & & \end{array} \right. \tag{55}$$

- The two first equations are related to the continuous phase. The first is the mass balance defined inside the volume p, while the second one is the mass balance defined at the interfacial zone with the dispersed phase q.

- The two last equations are related to the dispersed phase q. The first is the mass balance defined inside the volume p for the dispersed phase q while the second is the mass balance defined at the interfacial zone with the continuous phase.

- The term $K^0 \sum_{q=1}^{Q} \left([A]_p^0 - \hat{z}^q\right)$ account for the diffusion of the species in the continuous phase toward the different interfacial zones. Note that the diffusion coefficients embedded in the diagonal matrix K^0 do not depend on the interface: these coefficients are assumed to be the same for the diffusion toward all the interfaces. It is not difficult however to generalize the model to this case by defining a matrix $\hat{K}^q$ for each interface.

- Similarly, the terms $K^q \left([A]_p^q - z_p^q\right)$, $q = 1, \dots, Q$ account for the diffusion of the species in the dispersed phase toward their corresponding interface with the continuous phase.

- The assumption of no mass accumulation at the interfaces yields a set of nonlinear algebraic constraints. At this stage, the system is already no longer a simple ODE system, but a DAE system.

4.4. Integration of chemical equilibriums using the invariant method

Consider now the addition of chemical equilibriums to the system (55). These equilibriums should be satisfied in all the phases, and also at the interfacial zones. This can be modeled with the invariant method described in section 2. Indeed, the four mass balances of system (55) can be rewritten with the chemical invariants given by the composition matrices $\widetilde{L}^q$. Then, the resulting system is completed with mass action laws associated with each equilibrium.

$$\left\{\begin{array}{rclll}
\widetilde{L}^0\partial_t m_p^0 & = & \widetilde{L}^0\mathcal{T}_p^0(m_1^0,\ldots,m_P^0) & + \ \widetilde{L}^0(S_{kin}^0)^T v_{kin}^0([A]_p^0) & + \\
 & & \widetilde{L}^0 K^0 \displaystyle\sum_{q=1}^{Q}\left([A]_p^0-\hat{z}^q\right) & & \\
0 & = & \widetilde{L}^0 K^0(\hat{z}_p^q-[A]_p^0) & + \ \widetilde{L}^0(\hat{S}_{tr}^q)^T\sigma^q\hat{w}^q\left(\hat{z}_p^q,z_p^q\right) & + \\
\widetilde{L}^q\partial_t m_p^q & = & \widetilde{L}^q\mathcal{T}_p^q(m_1^q,\ldots,m_P^q) & + \ \widetilde{L}^q(S_{kin}^q)^T v_{kin}^q([A]_p^q) & + \\
 & & \widetilde{L}^q K^q\left([A]_p^q-z_p^q\right) & & \\
0 & = & \widetilde{L}^q(z_p^q-[A]_p^q) & + \ \widetilde{L}^q(S_{tr}^q)^T\sigma^q w^q\left(\hat{z}_p^q,z_p^q\right) & + \\
0 & = & g^0([A]_p^0) & & \\
0 & = & g^0(\hat{z}_p^q) & & \\
0 & = & g^q([A]_p^q) & & \\
0 & = & g^q(z_p^q) & &
\end{array}\right. \tag{56}$$

where $g^q=(g_1^q,\ldots,g_{N_{eq}^q})^T$.

The resulting system is a bound constrained DAE system. Indeed, in order to robustly compute the solution, the number of moles m_p^q and the interfacial concentrations z_p^q and $\hat{z}_p^q$ must remain positive. We give in the next section a numerical example that illustrates the importance of accurately satisfying the bound constraints and emphasizes the interest of using the bound constrained solver CDASSL instead of DASSL.

5. Numerical experiment

This section describes a realistic test case, related to liquid-liquid extraction with a biphasic chemical system in a mixer settler. The transport equations, involving only convection in this kind of equipment, are given in the first subsection. The chemical system is described in the second subsection. The numerical results obtained with DASSL and CDASSL are compared in the third subsection.

Figure 1: Mixer-settler scheme.

5.1. Transport equations in a biphasic mixer-settler

A biphasic mixer-settler is composed of an arbitrary number n stages. Each stage is composed of a mixer, in which an emulsion of the two phases of the chemical system is formed, and a settler, where the two phases are separated. In our model, each stage is composed of three elementary volumes: one for the mixer, and one for each phase (aqueous and organic) in the settler. In order to clarify the description, we assume the organic phase is continuous whereas the only dispersed phase is the aqueous phase. Moreover, we assume the aqueous phase is heavier than the organic phase. In order to produce the counter-current flow of the phases, the aqueous phase is introduced on the right side of the mixer-settler, whereas the organic phase is introduced on the left side. At each stage, the mixer is supplied with the aqueous phase from the settler of the right stage and with the organic phase from the left stage. The emulsion formed in the mixer flows into the settler of the same stage. The aqueous phase of this settler feeds the mixer of the left stage, whereas the organic phases feeds the mixer of the right stage. This counter current flow is illustrated in figure 1.

From a mathematical point of view, this amounts to model the transport by a 0D discretization of the extractor. Only convection is taken into account, which corresponds to the flows between the mixers and settlers.

In addition we assume steady-state transport in the extractor: no retroaction of the chemical phenomena over the transport properties of the fluids is taken into account, and the velocities of the flow between elementary volumes are constant, equal to the velocities of the imposed flux at the extremities of the extractor. The volumes of each phase in the mixers are computed given the total volume of the mixer and the assumption of well stirred flow. For an arbitrary stage i, we have for the mixer

$$\left\{ \begin{array}{l} V_i^0 + V_i^1 = V_i^M \\ \dfrac{V_i^0}{V_i^1} = \dfrac{d_i^0}{d_i^1}, \end{array} \right. \tag{57}$$

where the superscript 0 denotes the organic phase, which is continuous, the superscript 1 denotes the aqueous phase, and d_i^q denotes the outflow of phase q from mixer i.

The situation is simpler for the settlers, for which we assume the interface is fixed. The user can thus decide what are the volumes of each phase in the settlers.

We order the elementary volumes in the following way: at each stage,

- the first elementary volume is the aqueous phase of the settler;
- the second elementary volume is the mixer;
- the third elementary volume is the organic phase of the settler.

For an n-stage-mixer-settler, we end up with a a $3n$ elementary volumes extractor, and two linear operators T^0 and T^1 (where 0 stands for the aqueous phase and 1 for the organic phase, by convention) such that

$$\begin{cases} \mathcal{T}_p^q(m^0) = -d_p^0 m_p^0 + d_{p+1}^0 m_{p+1}^0, & p = 3n \qquad \text{(settler aqueous phase)} \\ \mathcal{T}_p^q(m^0) = -d_p^0 m_p^0 + d_{p+2}^0 m_{p+2}^0, & p = 3n+1 \qquad \text{(mixer aqueous phase)} \\ \mathcal{T}_p^q(m^1) = -d_p^1 m_p^1 + d_{p-2}^1 m_{p-2}^1, & p = 3n+1 \qquad \text{(mixer organic phase)} \\ \mathcal{T}_p^q(m^1) = -d_p^1 m_p^1 + d_{p-1}^1 m_{p-1}^1, & p = 3n+2 \qquad \text{(settler organic phase)} \end{cases} \tag{58}$$

5.2. Chemical system

As already mentioned, the system studied here is biphasic. The aqueous phase contains 6 species, namely

$$A, B, H, D, F, G; \tag{59}$$

and the organic phase contains 7 species, namely

$$E, BE_2, BE_3, BE_4, HE, DE, FE. \tag{60}$$

The process that is modeled is the extraction of component B from the aqueous phase using the extractor E from the organic phase to form the species BE_2, BE_3 and BE_4 in the organic phase. This is performed using an organic phase containing the components E and DE. The extraction efficiency of B involves a strong concentration of H (an acid for instance). Unfortunately, component DE of the organic phase is back-extracted into the aqueous phase in the form of component D which reacts with H to form component F, which degrades the efficiency of the extraction of B. To mitigate this difficulty, a component A is introduced in the aqueous phase, to react with D and form a component G. This is modeled by the following reactions and equilibriums.

5.2.1. Aqueous phase

In the aqueous phase, two kinetic-controlled reactions are defined

$$\begin{cases} D + F \longrightarrow H & (aq1) \\ A + D \longrightarrow G & (aq2) \end{cases} \tag{61}$$

For each of these reactions, a reaction rate is defined

$$\begin{cases} v_{aq1} = k_{aq1}[D][H] \\ v_{aq2} = k_{aq2}[A]^{\gamma}[D] \end{cases} \tag{62}$$

with

$$k_{aq1} = 10^2, \quad k_{aq2} = 10^7, \quad \gamma = 1. \tag{63}$$

Note that coefficient γ is set to 1 for the first experiment but is modified afterward.

5.2.2. Organic phase

Two equilibriums are defined in the organic phase:

$$\begin{cases} BE_2 + E \rightleftharpoons BE_3 & (org1) \\ BE_3 + E \rightleftharpoons BE_4 & (org2) \end{cases} \tag{64}$$

with the corresponding mass action laws

$$\begin{cases} g_{org1} = [BE_3] - K_{org1}[BE_2][E] \\ g_{org2} = [BE_4] - K_{org2}[BE_3][E] \end{cases} \tag{65}$$

where

$$K_{org1} = 1, \quad K_{org2} = 2. \tag{66}$$

These two equilibriums imply the definition of 5 invariants in the organic phase such as follows

$$\begin{cases} u_1 = E + BE_3 + 2BE_4 \\ u_2 = BE_2 + BE_3 + BE_4 \\ u_3 = HE \\ u_4 = DE \\ u_5 = FE \end{cases} \tag{67}$$

Note that the derivation "by hand" of these invariants, corresponding to the conservation of the constitutive elements of the system, would have produced the same system excepted for u_1. The latter would have been replaced by the invariant $u_1^{'}$ expressing the conservation of E

$$u_1^{'} = E + 2BE_2 + 3BE_3 + 4BE_4 + HE + DE + FE \tag{68}$$

This would probably have led to a more complex system.

5.2.3. Mass transfer

There are 4 kinetic reactions defined at the interface between the organic and the aqueous phase:

$$\left\{ \begin{array}{cccl} B + 2E & \longrightarrow & BE_2 & (trans1) \\ DE & \longrightarrow & D + E & (trans2) \\ H + E & \longrightarrow & HE & (trans3) \\ F + E & \longrightarrow & FE & (trans4) \end{array} \right. \tag{69}$$

For each of these interfacial reactions, a reaction rate is defined:

$$\left\{ \begin{array}{l} v_{trans1} = k_{trans1}[B][E]^2[H]^\alpha \\ v_{trans2} = k_{trans2}[DE]^2 \\ v_{trans3} = k_{trans3}[H][E] \\ v_{trans4} = k_{trans4}[F][E], \end{array} \right. \tag{70}$$

with

$$k_{trans_1} = 1, \;\; k_{trans_2} = 0.1, \;\; k_{trans_3} = 0.001, \;\; k_{trans_4} = 1, \;\; \alpha = 1.9 \tag{71}$$

5.3. Numerical results

We use a mixer-settler with 8 stages. Initially, the concentrations of all the species are set to 0. At stage 8, we introduce an aqueous flow with a velocity of $1\mathrm{L.h}^{-1}$ with the composition:

$$[A] = 0.5\mathrm{mol.L}^{-1}, \;\; [B] = 0.5\mathrm{mol.L}^{-1}, \;\; [H] = 1\mathrm{mol.L}^{-1}. \tag{72}$$

At stage 1, we introduce an organic flow with the same velocity of $1\mathrm{L.h}^{-1}$ and with the composition:

$$[E] = 0.5\mathrm{mol.L}^{-1}, \;\; [DE] = 1\mathrm{mol.L}^{-1}. \tag{73}$$

Figure 2: Solution computed by DASSL. Aqueous phase at stage 1 (top), organic phase at stage 8 (bottom).

Figure 3: Solution computed by DASSL. Aqueous phase at stage 1, zoom on the y intercept.

The relative and absolute tolerance required are set to 10^{-10}. The final time is sufficiently long to be sure to reach an equilibrium in the system: $t_f = 100$ h. Finally, the volumes of the mixer and the settlers at each stage are set to 1 L, and the Sauter diameter of the aqueous phase, which is dispersed in the organic phase, is set to 0.3×10^{-3} m.

In what follows, we denote by n_{iter} the number of iterations in time required for one simulation.

5.3.1. First experiment

We first run the simulation using the `Fortran 77` version of DASSL. In this case $n_{iter} = 3351$. The results obtained for the aqueous phase and the organic phase, respectively at stage 1 and stage 8 of the mixer-settler are presented in figure 2.

The plots show the expected behavior of the chemical system: component B is extracted from the aqueous phase to the organic phase in the form BE_2, at equilibrium with BE_3 and BE_4. Component D from the organic phase first degrades the acid H, but is soon degraded by component A which is introduced to this purpose, and finally the concentration of H is stabilized around 0.5 mol.L^{-1}, which allows a proper extraction of B.

Although the results proposed by DASSL appear to be correct, a zoom around the y intercept, in the aqueous phase of the first stage, show significant instabilities for the concentration of A (figure 3).

In order to reduce these instabilities, it is possible to use the nonnegativity option implemented in the `FORTRAN 77` version of DASSL. This corresponds

Figure 4: Solution computed by DASSL using the nonnegativity constraint. Aqueous phase at stage 1, zoom on the y intercept.

Figure 5: Solution computed by CDASSL. Aqueous phase at stage 1 (top), zoom on the y intercept.

roughly to setting to 0 all the components of the solution becoming between 0 and a tolerance rate η, at each time step

$$\eta \leq y_i^n < 0, \quad i = 1, \ldots, s \tag{74}$$

If one component is such that

$$y_i \leq \eta \tag{75}$$

then the solution is refused, the stepsize is reduced, and the solution has to be recomputed. This rather simple algorithm is sometimes sufficient to remove the instabilities of the solution. In this case, it is clearly not efficient, as shown in figure 4, even if the number of time steps required is almost the same as in the previous case: $n_{iter} = 3352$ In particular, we observe that although some instabilities are removed, some components of the solution are still strictly negative, which is quite surprising. This effect is probably due to the fact that in absolute values, the negative components of the computed solution are below a bound that must depend on the absolute and relative tolerance rates defined by the user, under which they are not detected by DASSL.

Instead of computing the solution with DASSL, we use the CDASSL solver with a lower bound set to 0 for each component of the solution. The overall results obtained by CDASSL present the same profile as those obtained by DASSL in figure 2. However, the number of required iterations is largely decreased: $n_{iter} = 1701$. In addition a zoom on the y intercept clearly shows that the instabilities have been removed: the solution is truly positive (fig. 5).

This example clearly underlines the importance of accounting for bound constraints when solving the equations associated to the liquid-liquid extraction model.

Figure 6: Solution computed by CDASSL on test case 2 ($\gamma = 1.1$). Aqueous phase at stage 1 (top), organic phase at stage 8 (bottom).

Figure 7: Solution computed by CDASSL on test case 2 ($\gamma = 1.1$). Aqueous phase at stage 1 (top), zoom on the y intercept.

5.3.2. Second experiment

In order to illustrate the importance in certain cases of preserving the positivity of the concentrations, we perform a second test, differing only slightly from the first one. We change the value of the coefficient γ in the reaction v_{aq2} from 1 to 1.1. This implies some derivatives with a term $[A]^{0.1}$ that could not be computed for $[A] < 0$.

This is clearly what we observe running the test with DASSL: the code returns an error flag after 295 iterations. This comes from the fact that the instabilities on $[A]$ prevents it to correctly evaluating the Jacobian matrix. Conversely, using CDASSL, we obtain the results presented in figures 6 and 7. The solution has the same aspect as for the previous experiment (fig.6), but the jump in the concentration of A is stiffer (fig. 7).

The solution can thus be computed, and the positivity of the solution is preserved. In turn, the computational effort to obtain this result is somewhat greater: $n_{iter} = 6336$. However, unlike DASSL, CDASSL is able to converge and compute the solution.

6. Conclusion

The work presented here provides a general method to account for chemical equilibriums in reactive transport, known as the invariant method. This method is defined at the semi-discretized in space level, and is adapted to any transport equations. From a numerical point of view, the resulting system is a stiff DAE system, with bound constraints: the unknowns that are computed are the number of moles and the concentrations that must remain positive for the sake of robustness. We thus use a numerical method, named CDASSL, specifically designed for the resolution of bound constrained DAE systems. This algorithm is based on the classical DAE solver DASSL, and uses a suitable interior point Newton method named CODOSOL to solve the nonlinear system at each time iteration.

In order to illustrate the efficiency of the invariant method, we present its application to a specific reactive transport application: liquid-liquid extraction. This separation process is widely used in industry, and plays a key role in several domains, including nuclear wastes recycling. We thus propose a liquid-liquid extraction model including chemical equilibriums through the invariant method. The numerical resolution of the resulting bound constrained DAE system is performed using the DASSL and CDASSL solvers. The numerical results demonstrate the importance of robustly taking bound constraints into account. Indeed, when using DASSL, some components of the solution oscillate around 0. While in certain cases, this does not prevent convergence, in other cases, it can lead to critical failure. The use of CDASSL eliminates the instabilities and avoids computational failure.

Acknowledgment

The authors are grateful to S.Bellavia, M.Macconi, S.Pieraccini for their interest in our method and fruitful discussions. They would also like to thank the members of the LSET laboratory at CEA Saclay and the LMPR laboratory at CEA Marcoule for their precious advice, expertise, and warm welcome.

References

S.Bellavia, M.Macconi, S.Pieraccini, Constrained Dogleg Methods for nonlinear systems with simple bounds, Computational Optimization and Applications, DOI 10.1007/s10589-012-9469-8, 2011.

K.E.Brenan, S.L.Campbell, L.R.Petzold, Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, Classics in Applied Mathematics 14, SIAM, North-Holland, New-York, Amsterdam, London, 1995.

J.R.Cash, Efficient numerical methods for the solution of stiff initial-value problems and differential algebraic equations, Proc. R. Soc. Lond.,**459**, 797-815, 2003.

C.W.Gear, L.R.Petzold, ODE Methods for the Solution of Differential/Algebraic Systems, SIAM J. Numer. Anal. **21**, 716-728.

G.H.Gollub, C.F.Van Loan, Matrix Computation, Third Edition, John Hopkins University Press, Baltimore, MD, USA, 1996.

E.Hairer, S.P.Nørsett, G.Wanner, Solving Ordinary Differential Equations, I, II, Springer, Berlin 1996.

C. de Dieuleveult, J.Erhel, M.Kern, A global strategy for solving reactive transport equations, Journal of Computational Physics, **228**, 6395-6410, 2009.

W.K.Lewis, W.G.Whitman, Principles of gas absorption, Ind. Eng. Chem, **16**, p.1215-1220, 1924.

L.Métivier, P.Montarnal, Strategies for solving index one DAE with non-negative contraints: Application to liquid-liquid extraction, Journal of Computational Physics **231**,2945-2962, 2012.

G.Psihoyios, Solving time dependent PDEs via an improved modified extended BDF scheme, Applied Mathematics and Computation, **184**, 104-115, 2007.

N. Di Miceli Raimondi, "Transfert de matière liquide-liquide en micro-canal: application à la réaction chimique ", PhD Thesis, Toulouse University, France, 2008.

W.E.Schiesser, The Numerical Method Of Lines: Integration of Partial Differential Equations, Academic Press, San Diego, 1991.

O.Weinstein, R.Semiat, D.R.Lewin, Modeling, simulation and control of liquid-liquid extraction columns, Chemical Engineering Science, **53**, 325-339, 1998.