A Reduction Scheme for Coupled Multicomponent Transport–Reaction Problems in Porous Media: Generalization to Problems with Heterogeneous Equilibrium Reactions

S. Kräutle, P. Knabner
kraeutle@am.uni-erlangen.de, knabner@am.uni-erlangen.de

Institute for Applied Mathematics, University of Erlangen-Nuremberg, Erlangen, Germany.

Abstract

In this article a systematic approach for the efficient computation of the transport and reaction of a multi-species multi-reaction system is proposed. The objective of this approach is to reformulate the given system of differential or differential-algebraic equations in such a way that the couplings and the nonlinearities are concentrated in a small number of equations, while some linear equations decouple from the system. The resulting system is handled in the spirit of a global implicit approach (‘one step method’) avoiding operator splitting techniques. The reduction of the problem size proposed in this article helps to limit the large computational costs of numerical simulations for such problems. The reduction mechanism is a generalization of the method proposed in a previous paper. Now, problems with mixed mobile/immobile species, homogeneous/heterogeneous kinetic/equilibrium reactions are considered, while the previous publication was restricted to problems without heterogeneous equilibrium reactions (equilibrium sorption). Applications of the reduction mechanism to example problems are given in order to investigate the reduction of the number of coupled nonlinear equations. For one example problem, numerical simulations were performed and the reduction of cpu time was measured.

Keywords. porous media, reactive flow, multispecies, differential algebraic equation, numerical simulation, reduction of problem size

AMS Classification. 1849, 1899, 1832, 1894

1 Introduction

This article is concerned with the efficient numerical solution of a large system of partial differential equations (PDEs) and ordinary differential equations (ODEs) as they occur in the modelling of transport, chemical reactions and biodegradation below the earth’s surface. The topic of this article is the proposition of a method to reformulate the given system of equations describing the problem, in such a way that the resulting system is easier to solve. A rather general approach for a wide class of such biogeochemical problems is proposed in this article. In fact, this article is a generalization of the work by Kräutle and Knabner [8], where equilibrium reactions between mobile and immobile species were excluded.

1 submitted July 2005
The chemical species which are considered are divided into two classes: Mobile (dissolved) species and immobile (sorbed chemical species, minerals, immobile bacteria) ones. The system of equations for the concentrations consists of PDEs for the mobile species and ODEs for the immobile ones, all of them coupled through the reaction terms. The characteristic timescale of the different reactions may cover a large range making it desirable to model some reactions as equilibrium-controlled and others as kinetically controlled, leading to an differential algebraic system of equations (DAE).

Basically there are two different concepts to treat these kinds of problems numerically: By global implicit approaches (GIA) and by sequential iterative and sequential noniterative approaches (SIA/SNIA). The GIA requires most resources per time step, but is usually considered to be the most stable solution method. SIA and SNIA, depending on the specific problem to solve, may suffer from heavy restrictions on the time step size to gain convergence, or from the introduction of large splitting errors, respectively. See, e.g., [15] and the papers cited therein, [17], [14].

In this article, our goal is to avoid such problems by focusing on the GIA. In order to keep the computational effort with respect to memory requirements and cpu time limited, we want to reformulate the given system of PDEs/ODEs or DAE in such a way that some of the equations decouple, leading to a smaller nonlinear system to which we apply the GIA. The reformulation of the given system is performed by (a) taking linear combinations of the given equations and (b) by introducing new variables which are linear combinations of the unknown concentrations. This leads to a decoupling of some scalar linear transport equations and a smaller remaining nonlinear system of PDEs, ODEs, and AEs. In another step, the local equations, i.e., the ODEs and the AEs, are solved for certain variables, and these variables are eliminated in the remaining PDEs, which reduces the size of the coupled system again.

There are many papers dealing with the efficient solution of transport-reaction problems in porous media, e.g., [5, 6, 13, 2, 3, 12, 7, 4], and, recently, the very advanced paradigm system by [10]. A main difference of the method proposed in this article to other reformulations (e.g., [13, 14, 10]) is that when we introduce linear combinations of concentrations or equations, we lay special emphasis on the distinction between mobile species and immobile species, not mixing up mobile and immobile species during the transformation. A benefit of this proceeding compared to other methods is that it avoids difficulties and the necessity of additional assumptions on the stoichiometry of the problem when we want to show that some equations decouple without 'enforcing' a decoupling by splitting techniques. Another advantage can be seen in the fact that the DSA-like treatment of the local equations in our resulting system preserves a very sparse population of the Jacobian (see [8], sec. 3.4), which can be exploited if the linear solver is an iterative method.

The article is structured as follows: In section 2, the equations for the coupled reactive transport are given. In section 3, which is the main part of this article, the general reduction algorithm including the case of heterogeneous equilibrium reactions is derived. The algorithm presented in section 3 requires a certain con-
dition on the stoichiometric matrix. In section 4 it is demonstrated that every stoichiometric system can be written in such a form that the required condition is met. Section 5 presents computational results concerning the reduction of cpu time for a test problem which includes aqueous complexation, equilibrium sorption and biodegradation modeled by the Monod approach. Section 6 performs the reduction method for two more example problems and compares the remaining number of coupled equations for the reduced scheme compared to the size of the given problem formulation. A mathematical proof that the local equations can always be solved for certain variables, if mass action law is assumed for the equilibrium reactions, is given in the appendix.

2 Problem Formulation

Let us consider $I$ mobile species $X_1, ..., X_I$ and $\bar{I}$ sorbed species $X_{I+1}, ..., X_{I+\bar{I}}$. Let us denote their time and space dependent concentrations by $c = (c_1, ..., c_I)^T$, $\bar{c} = (\bar{c}_{I+1}, ..., \bar{c}_{I+\bar{I}})^T$; the mobile $c_i$ are given in moles per fluid volume, the immobile $\bar{c}_i$ in moles per total volume. The mobile species are convected by a given Darcy flow field $q$ and are subject to dispersion. Let us assume that the underlying transport operator $L$ is linear and that it is the same for all mobile species, i.e., $L(c_1, ..., c_I)^T = (L_1 c_1, ..., L_I c_I)^T$ with $L_1 = ... = L_I$. A typical transport operator would be

$$L_i c_i = -\nabla \cdot (D \nabla c_i - q c_i), \quad i = 1, ..., I,$$

with dispersion tensor $D$.

The $J$ chemical reactions can be written as

$$\sum_{i=1}^{I} s_{ij} X_i + \sum_{i=1+I}^{I+\bar{I}} s_{ij} \bar{X}_i \xrightarrow{R_j} 0, \quad j = 1, ..., J,$$

where the stoichiometric coefficients $s_{ij}$ form an $(I + \bar{I}) \times J$-matrix $S$ with entries in $\mathbb{Z}$ or in $\mathbb{R}$ and $R_j = R_j(c, \bar{c})$ is the rate expression for reaction $j$.

The $I + \bar{I}$ mass balance equations are

$$\frac{d}{dt} (\theta c_i) + Lc_i = \sum_{j=1}^{J} s_{ij} R_j(c, \bar{c}), \quad i = 1, ..., I,$$

$$\frac{d}{dt} \bar{c}_i = \sum_{j=1}^{J} s_{ij} R_j(c, \bar{c}), \quad i = I+1, ..., I+\bar{I}. \quad (1)$$

$\theta > 0$ denotes the fraction of the mobile fluid-phase volume. We assume that the $s_{ij}$ are constant in space and time. The rates $R_j$ and $\theta$ may depend on space and time; however, for the sake of simplicity we assume that $\theta$ is constant.

The stoichiometric matrix $S$ consists of an $I \times J$ block $S_1$ of stoichiometric coefficients for the mobile species and an $\bar{I} \times J$ block $S_2$ of coefficients for the immobile species:

$$S = \begin{pmatrix} S_1 \\ S_2 \end{pmatrix}$$
Using the vector notation $R = (R_1, ..., R_J)^t$ we get the compact notation
\[
\theta \frac{d}{dt} c + Lc = S_1 R(c, \bar{c})
\]
\[
\frac{d}{dt} \bar{c} = S_2 R(c, \bar{c})
\]
for (1).

If a rate $R_j$ in (1)/(2) is kinetic, then we prescribe a rate function for this reaction, for example the forward-backward rate formulation
\[
R_j(c, \bar{c}) = k_f^j \prod_{i=1}^{I+I_j} c_i^{-s_{ij}} - k_b^j \prod_{i=1}^{I+I_j} c_i^{s_{ij}}.
\]
(3)

For the sake of simplicity we have omitted the bar atop the immobile species $\bar{c}_{I+1}, ..., \bar{c}_{I+I}$ in (3). However, other rate laws can be considered. For technical reasons we will assume that $\frac{\partial R_j}{\partial c_i}, \frac{\partial R_j}{\partial \bar{c}_i}$ exist and are bounded all over the computational domain for all species $c_i, \bar{c}_i$ and all kinetic reactions $R_j$.

If a reaction modelled with the law of mass action is assumed to be at local equilibrium then the algebraic equation
\[
Q_j(c, \bar{c}) := k_f^j \prod_{i=1}^{I+I_j} c_i^{-s_{ij}} - k_b^j \prod_{i=1}^{I+I_j} c_i^{s_{ij}} = 0
\]
holds at every point $x$ of the computational domain at any time. Note that for such equilibrium reactions $Q_j$ is not the same as the rate $R_j$ in system (1): While $Q_j$ denotes the mathematical expression that is zero in the case of equilibrium, the source term $R_j$ in (1) denotes the actual reaction rate at each point which is 'necessary' to keep the chemical system in equilibrium. These rates $R_j$ are usually unknown, and it is common use to eliminate them from the system (1)/(2) by taking linear combinations of the differential equations. This process leads to the introduction of what is often called components (see, e.g., [13]).

Let us assume reactions $R_1, ..., R_{J_{eq}}$ to be in equilibrium and reactions $R_{J_{eq}+1}, ..., R_J$ to be kinetic, $0 \leq J_{eq} \leq J$. We can split the vector $R$ into a vector of equilibrium reaction rates $R_{eq}$ of size $J_{eq}$, and a vector of kinetic reaction rates $R_{neq}$ of size $J_{neq} = J - J_{eq}$:
\[
R = \begin{pmatrix} R_{eq} \\ R_{neq} \end{pmatrix}
\]

Similar to vector $R$, we split the matrices $S, S_1, S_2$ into a block of $J_{eq}$ columns belonging to the equilibrium reactions, and a block of $J_{neq}$ columns for the kinetic reactions:
\[
S = \begin{pmatrix} S_1 \\ S_2 \end{pmatrix} = \begin{pmatrix} S_{eq}^1 & S_{neq}^1 \\ S_{eq}^2 & S_{neq}^2 \end{pmatrix} = \begin{pmatrix} S_{eq} \\ S_{neq} \end{pmatrix}
\]
(5)

Hence, (2) reads
\[
\theta \frac{d}{dt} c + Lc = S_{eq}^1 R_{eq} + S_{neq}^1 R_{neq}(c, \bar{c})
\]
\[
\frac{d}{dt} \bar{c} = S_{eq}^2 R_{eq} + S_{neq}^2 R_{neq}(c, \bar{c})
\]
(6)
Application of ln to (4) leads to the well known relation

\[ \hat{Q}_j(c, \bar{c}) := \sum_{i=1}^{I} s_{ij} \ln c_i + \sum_{i=I+1}^{I+I} s_{ij} \ln \bar{c}_i - K_j = 0, \]  

(7)

\( j = 1, \ldots, J_{eq} \), where the equilibrium constants \( K_j \) are defined by \( K_j = \ln(k_f^j / k_b^j) \).

The vector \( \hat{Q}_{eq} = (\hat{Q}_1, \ldots, \hat{Q}_{J_{eq}})^t \) of the algebraic equilibrium conditions (7) can be expressed by

\[ \hat{Q}_{eq}(c, \bar{c}) = (S_{eq}^1)^t \ln c + (S_{eq}^2)^t \ln \bar{c} - K \]

(8)

where the vector of equilibrium constants \( K \) is defined by \( K = (K_1, \ldots, K_{J_{eq}})^t \), and \( \ln \) applied to a vector with entries \( c_i > 0 \) is short for the vector with the entries \( \ln c_i \).

The well known way to handle system (6) together with (4) or (8) is to form linear combinations of the equations in (6) in such a way that the unknown reaction terms \( R_{eq} \) vanish. The forming of linear combinations of equations can be expressed by multiplying (2) with certain matrices (see, e.g., [13, 10]. A main difference between different reduction methods lies in the choice of these matrices.

Throughout this article we assume that all species have ideal activity. Especially if minerals are considered, then this assumption seems not reasonable. For species with constant activity, the remarks made in [8], sec. 4.4, apply, i.e., the reduction algorithm is applicable as long as no mineral reaches zero concentration.

For aqueous species with non-ideal activity, one could assume that during each time step, the activity coefficient remains constant. This leads to a formulation of the equilibrium conditions again depending on the concentrations instead of the activities like equation (8), but now, with different equilibrium constants \( K \) incorporating the activity coefficients of the species. By this, the reduction algorithm of the following section is applicable also for non-ideal activities.

3 The decoupling algorithm

3.1 The transformation of the mobile and of the immobile block

Let us recall the basic idea of the transformation in [8].

The first step of the decoupling method is to transform the two blocks of (2) separately: Let \( J_1, J_2 \) be the number of linearly independent columns in the matrix \( S_1, S_2 \), respectively. For \( S_i, i = 1, 2 \), we define \( S_i^* \) as a matrix consisting of a maximum system of linearly independent columns of \( S_i \) and \( A_i \) such that

\[ S_i = S_i^* A_i \]  

(9)

holds. \( A_1 \) is a matrix of size \( J_1 \times J \) and \( A_2 \) of size \( J_2 \times J \). For \( S_i^* \) we define \( S_i^\top \) consisting of a maximum set of linearly independent columns that are orthogonal
to each column of $S_i^*$. By construction,

$$S_i^T S_i^* = 0, \quad S_i^* S_i^T = 0 \quad (10)$$

hold. Since the columns of $S_i^*$ and of $S_i^T$ are linearly independent, the matrices $S_i^* S_i^*$, $S_i^T S_i^T$ are invertible. By using (9) in (2) and then multiplication of each block of (2) by

$$(S_i^* S_i^*)^{-1} S_i^* \text{ and by } (S_i^T S_i^T)^{-1} S_i^T \quad (11)$$

we derive the following four blocks:

$$(S_1^T S_1^T)^{-1} S_1^T (\theta \frac{d}{dt} c + Lc) = 0$$

$$(S_2^T S_2^T)^{-1} S_2^T \frac{d}{dt} \bar{c} = 0 \quad (12)$$

$$(S_1^* S_1^*)^{-1} S_1^* (\theta \frac{d}{dt} c + Lc) = A_1 R(c, \bar{c})$$

$$(S_2^* S_2^*)^{-1} S_2^* \frac{d}{dt} \bar{c} = A_2 R(c, \bar{c})$$

This manipulation corresponds to the forming of linear combinations within each of the two blocks of equations in (2). The number of equations in (12) is the same as the number of equations in (2) or (6). Hence, the reduction is still to come.

Since the matrices and the differential operators in system (12) commute, we can substitute

$$\eta = (S_1^T S_1^T)^{-1} S_1^T c, \quad \xi = (S_1^* S_1^*)^{-1} S_1^* c, \quad \bar{\eta} = (S_2^T S_2^T)^{-1} S_2^T \bar{c}, \quad \bar{\xi} = (S_2^* S_2^*)^{-1} S_2^* \bar{c}. \quad (13)$$

The inversion of relation (13) reads

$$c = S_1^* \xi + S_1^T \eta, \quad \bar{c} = S_2^* \bar{\xi} + S_2^T \bar{\eta}. \quad (14)$$

The vectors $(\xi, \eta) \in \mathbb{R}^I$, $(\bar{\xi}, \bar{\eta}) \in \mathbb{R}^J$ are representations of the vectors $c$, $\bar{c}$, respectively, with respect to another basis. $\xi$ and $\eta$ are linear combinations of only mobile species, and $\bar{\eta}$, $\bar{\xi}$ are linear combinations of only immobile species.

System (12) becomes

$$\theta \frac{d}{dt} \eta + L \eta = 0 \quad (I - J_1 \text{ equ.})$$

$$\frac{d}{dt} \bar{\eta} = 0 \quad (\bar{I} - J_2 \text{ equ.})$$

$$\theta \frac{d}{dt} \xi + L \xi = A_1 R \left( \frac{S_1^* \xi + S_1^T \eta}{S_2^* \bar{\xi} + S_2^T \bar{\eta}} \right) \quad (J_1 \text{ equ.}) \quad (15)$$

$$\frac{d}{dt} \bar{\xi} = A_2 R \left( \frac{S_1^* \xi + S_1^T \eta}{S_2^* \bar{\xi} + S_2^T \bar{\eta}} \right). \quad (J_2 \text{ equ.})$$

In this formulation, the equations for the components $\eta$ and $\bar{\eta}$ are decoupled from the the equations for $\xi$, $\bar{\xi}$. Note that the decoupled equations for $\eta$ are linear and scalar which makes their solution rather fast. The components $\bar{\eta}$ are even
constant throughout the whole simulation. The quantities \( \eta, \bar{\eta} \) can be considered as 'reaction invariants'; the evolution of these quantities is completely determined by the initial and boundary values for \( c, \bar{c} \) and is independent of the reactions.

The number of remaining coupled nonlinear equations for this representation is \( J_1 + J_2 \). If all reactions are kinetic, then the formulation (15) can be used for simulation. However, it is possible to further reduce the system size: After discretization in time, last last block in (15) consists of local algebraic equations for \( \xi \). These equations can be solved for \( \xi \) and substituted in the right hand side of the third block in (15). This reduces the number of remaining coupled nonlinear PDEs to \( J_1 \).

If some of the reactions are assumed to be at equilibrium (formulation (6) together with (8)), then formulation (15) cannot be used directly. It is necessary to eliminate the equilibrium rates \( R_{eq} \) from the system (15) and use the remaining equations together with the equilibrium conditions (8). The way how to eliminate the equilibrium reactions from the system will be motivated by an example in the following section.

### 3.2 The further treatment for an example problem including equilibrium sorption

Let us consider the chemical reaction network

\[
\begin{align*}
2A & \xrightarrow{R_1} B \\
B + C & \xrightarrow{R_2} \bar{D} \\
\bar{D} & \xrightarrow{R_3} \bar{E}
\end{align*}
\]

consisting of 3 mobile species \( A, B, C \) and 2 immobile species \( \bar{D}, \bar{E} \) and 3 chemical reactions \( R_1, R_2, R_3 \). The corresponding nonlinear system of PDEs reads

\[
\begin{align*}
\theta A + LA &= -2R_1(A, B) \\
\theta B + LB &= R_1(A, B) - R_2(B, C, \bar{D}) \\
\theta C + LC &= -R_2(B, C, \bar{D}) \\
\bar{D} &= R_2(B, C, \bar{D}) - R_3(B, \bar{D}, \bar{E}) \\
\bar{E} &= R_3(B, \bar{D}, \bar{E})
\end{align*}
\]

Let us apply the algorithm of Section 3.1 to (16)/(17): The stoichiometric matrix reads

\[
S = \begin{pmatrix}
S_1 \\
S_2
\end{pmatrix} = \begin{pmatrix}
-2 & 0 & 0 \\
1 & -1 & 0 \\
0 & -1 & 0 \\
0 & 1 & -1 \\
0 & 0 & 1
\end{pmatrix}
\]

The extraction of the linearly independent columns yields

\[
S_1^* = \begin{pmatrix}
-2 & 0 \\
1 & -1 \\
0 & -1
\end{pmatrix}, \quad S_2^* = \begin{pmatrix}
1 & -1 \\
0 & 1
\end{pmatrix}
\]
THE DECOUPLING ALGORITHM

\[ A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \]

and for the orthogonal complement of \( S^*_1 \) we choose

\[ S^\top_1 = \begin{pmatrix} 1/2 \\ 1 \\ -1 \end{pmatrix}, \]

and \( S^\top_2 = () \) is the empty matrix. Hence, the algorithm of section 3.1 yields the system

\[
\begin{align*}
(\theta \frac{d}{dt} + L) \xi_1 &= R_1 \\
(\theta \frac{d}{dt} + L) \xi_2 &= R_2 \\
(\theta \frac{d}{dt} + L) \eta_1 &= 0 \\
\frac{d}{dt} \xi_1 &= R_2 \\
\frac{d}{dt} \xi_2 &= R_3
\end{align*}
\]

where

\[
\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \frac{1}{9} \begin{pmatrix} -4 & 1 & -1 \\ -2 & -4 & -5 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \end{pmatrix},
\]

\[ \eta_1 = \frac{4}{9} \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \end{pmatrix}, \]

\[ \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} D \\ E \end{pmatrix}. \]

Equation (18) corresponds to the general form (15).

Let us analyse (18) with respect to the case that reactions can be at equilibrium. In this case, our goal is a formulation in which every equilibrium reaction occurs only in one equation, and where in each equation, there is at most one equilibrium reaction term. Then we replace the corresponding PDE or ODE by the corresponding algebraic equilibrium condition (4) or (7). If \( R_1 \) or \( R_3 \) is at equilibrium in (18), then we can immediately proceed like this. If \( R_2 \) is at equilibrium, then we first have to replace the second or the fourth line in (18) by the difference of these two equations:

\[
\begin{align*}
(\theta \frac{d}{dt} + L) \xi_1 &= R_1 \\
\frac{d}{dt}(\theta \xi_2 - \xi_1) + L \xi_2 &= 0 \\
(\theta \frac{d}{dt} + L) \eta_1 &= 0 \\
\frac{d}{dt} \xi_1 &= R_2 \\
\frac{d}{dt} \xi_2 &= R_3
\end{align*}
\]

In this formulation, an arbitrary subset of the three reactions can be assumed to be at equilibrium, since we can replace the corresponding PDEs/ODEs by the corresponding algebraic equilibrium conditions \( Q_j = 0 \) or \( \dot{Q}_j = 0 \).
3.3 The general algorithm capable of handling heterogeneous equilibrium reactions

Analysis of this example problem shows under which assumptions we can generalize the proceeding: Reactions $R_1$ and $R_3$ are homogeneous; they only occur in exactly one of the two blocks (17). Therefore they occur only in one of the blocks of the system (18). Due to the special structure of $A_1$, $A_2$, which consist of unit vector columns and zero columns (this is a consequence of the fact that $S_1$, $S_2$ consist of linear independent and zero vectors), they occur only once in the whole system. However, reaction $R_2$ is heterogeneous, hence, it occurs in both blocks of (17) and therefore occurs in both blocks of the system (18). Again, thanks to the structure of $A_1$, $A_2$, the heterogeneous reaction occurs exactly in one equation of each block of the system (18). By taking the difference of these two equations, as it was done in (19), one occurrence of $R_2$ can be eliminated.

So what we need to apply the method used for this example, is basically that those columns of $A_1$ and $A_2$ that correspond to equilibrium reactions, are certain unit vectors or zero columns. For this, we need that $S^1_{eq}$ and $S^2_{eq}$ consist of linearly independent columns plus possibly some zero columns; see section 3.3 for the details.

3.3 The general algorithm capable of handling heterogeneous equilibrium reactions

Let us sort the vector of equilibrium reactions in the following way: We start with those equilibrium reactions whose participants are all mobile, then we take the heterogeneous equilibrium reactions (i.e., those having participants both in the mobile and in the immobile phase) and at last those equilibrium reactions in the immobile phase:

$$R_{eq} = (R_{mob}, R_{sorp}, R_{immo}),$$
$$R = (R_{eq}, R_{neq}) = (R_{mob}, R_{sorp}, R_{immo}, R_{neq})$$

Note the slight abuse of notation, since $R_{sorp}$ not only contains equilibrium sorption reactions, but also ion exchange reactions and heterogeneous classical chemical reactions. For the kinetic reactions $R_{neq}$, no special order is required. The size of the subvectors is $J_{mob}, J_{sorp}, J_{immo}, J_{neq}$; $J_{mob} + J_{sorp} + J_{immo} = J_{eq}$; $J_{eq} + J_{neq} = J$.

Due to this order, the stoichiometric matrix has the following shape:

$$S = \begin{pmatrix}
S^1_1 \\ S^1_2 \\
S^2_1 \\ S^2_2
\end{pmatrix} =
\begin{pmatrix}
S^1_{eq} & S^1_{neq} \\
S^2_{eq} & S^2_{neq}
\end{pmatrix}
\begin{pmatrix}
S^1_{mob} \\ S^1_{sorp} \\
0 \\ S^2_{sorp}
\end{pmatrix}
\begin{pmatrix}
S^1_{immo} \\ S^2_{immo} \\
S^1_{neq} \\ S^2_{neq}
\end{pmatrix}

(20)

Note that for representation (20) we made no assumptions on the stoichiometry; every reactive systems has such a representation. A usual assumption on the stoichiometric matrix is that all the columns of $S$ are linearly independent. What we require is that at least the columns of the equilibrium part $S_{eq}$ are linearly independent. Hence, also the columns in $S^1_{mob}$, in $S^2_{immo}$ and in $\begin{pmatrix}S^1_{sorp} \\ S^2_{sorp}\end{pmatrix}$ are linearly independent.
As an additional requirement, motivated by the example in section 3.2, we postulate that both

the columns in $(S_{\text{mob}}^1, S_{\text{sorp}}^1)$ and in $(S_{\text{sorp}}^2, S_{\text{immo}}^2)$ are linearly independent. (21)

This condition is not met for arbitrary stoichiometric matrices (20), but in section 4 we will show that without loss of generality, each chemical system can be formulated in a way fulfilling condition (21).

As a consequence of (21), we can choose a maximum set of linearly independent columns from $S_1$ containing $(S_{\text{mob}}^1, S_{\text{sorp}}^1)$, and a maximum set of linearly independent columns from $S_2$ containing $(S_{\text{sorp}}^2, S_{\text{immo}}^2)$. As in section 3.1, we denote these matrices by $S_1^*, S_2^*$ again. We get the block structure

$$S_1^* = \begin{pmatrix} S_{\text{mob}}^1 & S_{\text{sorp}}^1 & S_{\text{neq}}^{1'} \\ 0 & 0 & 0 \end{pmatrix},$$

$$S_2^* = \begin{pmatrix} S_{\text{sorp}}^2 & S_{\text{immo}}^2 & S_{\text{neq}}^{2'} \\ 0 & 0 & 0 \end{pmatrix},$$

(22)

where $S_{\text{neq}}^{1'}, S_{\text{neq}}^{2'}$ consist of columns taken from $S_{\text{neq}}^1, S_{\text{neq}}^2$, respectively. Let $J_{\text{neq}}^{1'}, J_{\text{neq}}^{2'}$ be the number of columns of $S_{\text{neq}}^{1'}, S_{\text{neq}}^{2'}$.

The selection process is described by

$$S_1 = S_1^* A_1, \quad S_2 = S_2^* A_2,$$

where $A_1, A_2$ have, thanks to (21) and (22), the block structure

$$A_1 = \begin{pmatrix} \text{Id}_{J_{\text{mob}}} & 0 & 0 & A_{\text{mob}}^1 \\ 0 & \text{Id}_{J_{\text{sorp}}} & 0 & A_{\text{sorp}}^1 \\ 0 & 0 & 0 & A_{\text{neq}}^1 \end{pmatrix},$$

(23)

$$A_2 = \begin{pmatrix} 0 & \text{Id}_{J_{\text{sorp}}} & 0 & A_{\text{sorp}}^2 \\ 0 & 0 & \text{Id}_{J_{\text{immo}}} & A_{\text{immo}}^2 \\ 0 & 0 & 0 & A_{\text{neq}}^2 \end{pmatrix},$$

(24)

Exactly as in (9)-(15) we define $S_1^T, S_2^T$, multiply the two blocks of the given system (2) by the matrices (11) and substitute $c, \bar{c}$ by the new variables $\eta, \xi, \bar{\eta}, \bar{\xi}$. Additionally we introduce the splitting into subvectors

$$\xi = (\xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{neq}}), \quad \bar{\xi} = (\bar{\xi}_{\text{sorp}}, \bar{\xi}_{\text{immo}}, \bar{\xi}_{\text{neq}})$$

of size $J_{\text{mob}}, J_{\text{sorp}}, J_{\text{neq}}^{1'}; J_{\text{sorp}}, J_{\text{immo}}, J_{\text{neq}}^{2'}$, and we make use of the block structure
The kinetic reaction rates exactly twice. The assumption (21) was necessary to guarantee these properties. We get:

### 3.3 The general algorithm capable of handling heterogeneous equilibrium reactions

(23), (24). We get:

\[
\begin{align*}
(1) & \quad \left( \frac{\partial}{\partial t} + L \right) \eta = 0 \\
(2) & \quad \frac{d}{dt} \eta = 0 \\
(3) & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{mob}} = R_{\text{mob}} + A_{\text{mob}}^1 R_{\text{neq}}(c, \varphi) \\
(4) & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{sorp}} = R_{\text{sorp}} + A_{\text{sorp}}^1 R_{\text{neq}}(c, \varphi) \\
(5) & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{neq}} = A_{\text{neq}}^1 R_{\text{neq}}(c, \varphi) \\
(6) & \quad \frac{d}{dt} \xi_{\text{sorp}} = R_{\text{sorp}} + A_{\text{sorp}}^2 R_{\text{neq}}(c, \varphi) \\
(7) & \quad \frac{d}{dt} \xi_{\text{immo}} = R_{\text{immo}} + A_{\text{immo}}^2 R_{\text{neq}}(c, \varphi) \\
(8) & \quad \frac{d}{dt} \xi_{\text{neq}} = A_{\text{neq}}^2 R_{\text{neq}}(c, \varphi),
\end{align*}
\]

This formulation corresponds completely to the representation (15); the only difference is that we used the block structure of $S, A_1, A_2$. Formulation (25) also corresponds to formulation (18) of the example problem: Those equilibrium reactions which involve only mobile species or only immobile species $(R_{\text{mob}}, R_{\text{immo}})$ occur exactly once, and the heterogeneous equilibrium reactions $R_{\text{sorp}}$ occur exactly twice. The assumption (21) was necessary to guarantee these properties. The kinetic reaction rates $R_{\text{neq}}$ can occur multiple times.

As in the example of Section 3.2, we have to eliminate one occurrence of $R_{\text{sorp}}$ by taking the difference of block 4 and block 6 in (25):

\[
\begin{align*}
(1) & \quad \left( \frac{\partial}{\partial t} + L \right) \eta = 0 \\
(2) & \quad \frac{d}{dt} \eta = 0 \\
(3) & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{mob}} = R_{\text{mob}} + A_{\text{mob}}^1 R_{\text{neq}}(c, \varphi) \\
(4-6) & \quad \frac{d}{dt}(\theta \xi_{\text{sorp}} - \xi_{\text{sorp}}) + L \xi_{\text{sorp}} = A_{\text{sorp}}^1 R_{\text{neq}}(c, \varphi) \\
(5) & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{neq}} = A_{\text{neq}}^1 R_{\text{neq}}(c, \varphi) \\
(6) & \quad \frac{d}{dt} \xi_{\text{sorp}} = R_{\text{sorp}} + A_{\text{sorp}}^2 R_{\text{neq}}(c, \varphi) \\
(7) & \quad \frac{d}{dt} \xi_{\text{immo}} = R_{\text{immo}} + A_{\text{immo}}^2 R_{\text{neq}}(c, \varphi) \\
(8) & \quad \frac{d}{dt} \xi_{\text{neq}} = A_{\text{neq}}^2 R_{\text{neq}}(c, \varphi),
\end{align*}
\]

where we have set $A_{\text{sorp}} := A_{\text{sorp}}^1 - A_{\text{sorp}}^2$. Now, since all equilibrium reactions $(R_{\text{mob}}, R_{\text{sorp}}, R_{\text{immo}}) = R_{\text{eq}}$ occur exactly once, we can replace blocks (3), (6), (7) by the AE describing the equilibrium. If mass action law is assumed for all equilibrium reactions, then this AE is (4) for $j = 1, ..., J_{\text{eq}}$ or (8):

\[
\begin{align*}
(1') & \quad \left( \frac{\partial}{\partial t} + L \right) \eta = 0 \quad (I \text{ eq.}) \\
(2') & \quad \frac{d}{dt} \eta = 0 \quad (I \text{ eq.}) \\
(3') & \quad Q_{\text{eq}}(c, \varphi) = 0 \quad (J_{\text{eq}} \text{ eq.}) \\
(4') & \quad \frac{d}{dt} \xi_{\text{neq}} = A_{\text{neq}}^2 R_{\text{neq}}(c, \varphi) \quad (J_{\text{eq}}^2 \text{ eq.}) \\
(5') & \quad \frac{d}{dt}(\theta \xi_{\text{sorp}} - \xi_{\text{sorp}}) + L \xi_{\text{sorp}} = A_{\text{sorp}} R_{\text{neq}}(c, \varphi) \quad (J_{\text{eq}} \text{ eq.}) \\
(6') & \quad \left( \frac{\partial}{\partial t} + L \right) \xi_{\text{neq}} = A_{\text{neq}}^1 R_{\text{neq}}(c, \varphi) \quad (J_{\text{eq}}^1 \text{ eq.}) \\
\end{align*}
\]

The system is closed (see (14)) by

\[
c = S^T_1 \xi + S^T_1 \eta
\]
THE DECOUPLING ALGORITHM

\[
\begin{align*}
S_1 \xi_{\text{mob}} + S_1' \xi_{\text{neq}} + S_1^T \eta, \\
S_2 \xi + S_2^T \eta \\
= S_2 \xi_{\text{sorp}} + S_2' \xi_{\text{immo}} + S_{\text{neq}} \xi_{\text{neq}} + S_2^T \eta.
\end{align*}
\]

The system (27) consists of the decoupled linear problems (1') for \( \eta \), a block of 'local' problems (3'), (4'), and a system of PDEs (5'), (6'). The character of equation (5') is that of a 'generalized equilibrium sorption process'. As the next sections shows, the blocks of local equations (3'), (4') can be solved for the unknowns \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}} \) (or for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}}, \xi_{\text{neq}} \)), and substituted into the remaining PDEs in the sense of a DSA. Note that (different from the scheme arising from method by Friedly and Rubin [6], if considered as a GIA with DSA technique), the introduction of the resolution functions for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}} \) in (5'), (6') does not take place under the transport operator. Hence the sparsity pattern for the linear problems arising from the discretization of (27) is more convenient for efficient numerical solution with iterative linear solvers. See [8], sec. 3.4, for a more detailed discussion of the sparsity pattern.

After elimination of the local equations (3'), (4'), the size of remaining coupled nonlinear equations (5'), (6') is \( J_{\text{sorp}} + J_{\text{neq}}' \), which is less than or equal to \( J_{\text{sorp}} + J_{\text{neq}} \), which is again less than or equal to the total number of reactions \( J \). The number of homogeneous equilibrium reactions \( (J_{\text{mob}}, J_{\text{immo}}) \) does not influence the size of the resulting system.

3.4 The implicit elimination process

In this section, the solvability of the blocks of local equations in scheme (27) is discussed under rather general assumptions.

Theorem. Let assumption (21) hold and let there be a positive lower bound for all concentrations \( c_i, \bar{c}_i \), uniform with respect to \( x \).

(a) \( (i) \) If the equilibrium reactions (3') are governed by the mass action law, then there is a local resolution function for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}} \) (depending on \( \xi_{\text{sorp}}, \xi_{\text{neq}}, \xi_{\text{neq}} \)).

(ii) If block (4') is discretized with the implicit Euler method and if the timestep size \( \Delta t \) is sufficiently small, or if the explicit Euler method is used and \( \Delta t > 0 \) is arbitrary, then there is a local resolution function of this block for \( \xi_{\text{neq}} \) (depending on \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{neq}}, \xi_{\text{immo}} \)).

(iii) Under the assumptions of (i) and (ii), there is a local resolution function of block (3'), (4') for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}}, \xi_{\text{neq}} \) (depending on \( \xi_{\text{sorp}}, \xi_{\text{neq}} \)).

(b) The assertion (i)-(iii) is also true if we exchange \( \xi_{\text{sorp}} \) and \( \xi_{\text{immo}} \) in (a).

Proof: See appendix.

Note that the choice to solve (3') for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}} \) (or for \( \xi_{\text{mob}}, \xi_{\text{sorp}}, \xi_{\text{immo}} \)) corresponds to the segregation of the so-called secondary dependent variables from the primary ones. So for our method, this segregation is not done for the original
unknowns \( c_i, \bar{c}_i \), as it is done, e.g., in [9], and by most of the authors cited in section 1, but for the new unknowns \( \xi_i, \bar{\xi}_i \).

Let us mention that it is possible the show that under the assumptions of the theorem, the system of differential equations we get after substitution of the local variables is parabolic and does not degenerate.

### 3.5 Newton’s method for the reduced system

Each time step for our reduced system (27) consists of the following steps:

First we perform the time step for the linear scalar decoupled equations for \( \xi \). Then we perform a Newton or Newton-like iteration for the nonlinear problem consisting of equations (5'), (6') for the variables \( \xi_{sorp} \) and \( \xi_{neq} \). We refer to these variables as 'global' variables, since they are coupled by a system of PDEs. The variables \( \xi_{mob}, \bar{\xi}_{sorp}, \bar{\xi}_{immo}, \bar{\xi}_{neq} \) are considered to be eliminated from (5'), (6') by solving (3'), (4') for these variables. It is not necessary (and also not realistic) that we find the solution function of (3'), (4') explicitly; it is sufficient that we can compute the Jacobian for the reduced problem. Notice that the Newton step for (5'), (6') only gives an update for the global variables. In order to get an update for the remaining so-called 'local' variables, we have to perform a nested Newton iteration for problem (3'), (4') with fixed (updated) values of the global variables. This nested problem consists of small local decoupled problems for the different nodes or control volumes of the computational domain. Numerical test runs confirm that the costs for solving these local problems are negligible compared to the cpu time for the global problem.

### 4 Reduction for arbitrary stoichiometric systems

In section 3.3 the assumption (21) on the equilibrium reactions was made in order to guarantee the applicability of the reduction algorithm. The condition was used

1. to derive representation (25) such that the generalized sorption formulation (26) could be derived and

2. to guarantee the solvability of the equilibrium conditions (3') in (27) (see proof of the theorem part (i) in the appendix).

In this section we will show that every reactive system can be formulated in such a way that this condition is met. We will see that two simple preprocessing steps for the given problem are sufficient.

For the following, let us consider an arbitrarily given stoichiometric matrix. Without loss of generality we can assume that as least the equilibrium reaction (i.e., the columns of \( S_{eq} \)) are linearly independent.
4 REDUCTION FOR ARBITRARY STOICHIOMETRIC SYSTEMS

4.1 Preprocessing step I

Example 3. Let us consider the 6 species $A, B, C, \bar{D}, \bar{E}, \bar{F}$ and the following $J_{eq} = J_{sorp} = 4$ heterogeneous equilibrium reactions $R_1$-$R_4$:

$$B \xrightleftharpoons{R_1} A + \bar{F}$$
$$\bar{E} \xrightleftharpoons{R_2} C + 3\bar{F}$$
$$C + \bar{F} \xrightleftharpoons{R_3} 2\bar{D}$$
$$B \xrightleftharpoons{R_4} A + \bar{E}$$

The corresponding equilibrium conditions are

$$\frac{c_A c_{\bar{F}}}{c_B} = K_1 \quad \frac{c_C c_{\bar{F}}^3}{c_{\bar{E}}} = K_2$$
$$\frac{c_{\bar{D}}^2}{c_C c_{\bar{F}}} = K_3 \quad \frac{c_A c_{\bar{E}}}{c_B} = K_4$$

and the corresponding stoichiometric matrix is

$$S_{eq} = \left( \begin{array}{c|c}
S_{eq}^1 & S_{eq}^2 \\
\end{array} \right) = \begin{pmatrix}
1 & 0 & 0 & 1 \\
-1 & 0 & 0 & -1 \\
0 & 1 & -1 & 0 \\
0 & 0 & 2 & 0 \\
0 & -1 & 0 & 1 \\
1 & 3 & -1 & 0
\end{pmatrix}$$

(30)

Clearly, both the columns of $\left( S_{mob}^1 \mid S_{sorp}^1 \right) = S_{eq}^1$ and the columns of $\left( S_{sorp}^2 \mid S_{immo}^2 \right) = S_{eq}^2$ are linearly dependent, while the columns of $S_{eq}$ are linearly independent; condition (21) is not met.

The preprocessing step I now consists of a Gaussian (row-based) elimination for $S_{eq}^1$ (or, equivalently, a column-based Gaussian elimination for $S_{eq}$):

$$S_{eq}^1 = \begin{pmatrix}
1 & -1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & -1 & 3 \\
0 & 0 & -1 & 2 & 0 & -1 \\
1 & -1 & 0 & 0 & 1 & 0
\end{pmatrix} \rightarrow \begin{pmatrix}
1 & -1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 2 \\
0 & 0 & 0 & 1 & 0 & \frac{1}{2} \\
0 & 0 & 0 & 0 & 1 & -1
\end{pmatrix}$$

(31)

After transposition we get the following matrix, which will, to keep the notation simple, be denoted by $S_{eq}$ again:

$$S_{eq} = \begin{pmatrix}
1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
1 & 2 & \frac{1}{2} & -1
\end{pmatrix}$$

(32)
The Gaussian elimination does not modify the stoichiometry: An addition of two columns of \( S_{eq} \) corresponds, for example, to the multiplication of one equilibrium condition in (29) by another.

Since all \( J_{eq} \) columns of the given matrix (30) are linearly independent, the resulting matrix (32) has \( J_{eq} \) rows which equal the \( J_{eq} \) unit vectors in \( \mathbb{R}^{J_{eq}} \). Each column in matrix (32) contains exactly one of the unities (printed in bold fonts). Each column in the upper part \( S_{eq}^{1} \) of matrix (32) contains either a unity or is completely zero. From this representation it is clear that we can write (32) as

\[
S_{eq} = \begin{pmatrix}
S_{mob}^{1} & S_{sorp}^{1} & 0 \\
0 & S_{sorp}^{2} & S_{immo}^{2}
\end{pmatrix}
\]  

(33)
such that

all columns in the submatrix \( \begin{pmatrix} S_{mob}^{1} & S_{sorp}^{1} \end{pmatrix} \)

are linearly independent.

The columns in \( S_{immo}^{2} \) are also linearly independent, since all columns in \( \begin{pmatrix} 0 \\ S_{eq} \end{pmatrix} \)

are linearly independent.

Note that for Example 3, the stronger condition (21) is still not met after the preprocessing.

### 4.2 Preprocessing step II

Let us consider a matrix of type (33) with linearly independent columns which fulfills (34). We will transform it into a matrix fulfilling (21) using column operations:

Let us denote the columns of \( S_{eq}^{1}, S_{eq}^{2} \) by \( s_{j}^{1}, s_{j}^{2}, j = 1, ..., J_{eq} \). Suppose the columns in \( \begin{pmatrix} S_{sorp}^{2} & S_{immo}^{2} \end{pmatrix} \) are linearly dependent. Then there is a linear combination

\[
\sum_{j=J_{mob}+1}^{J_{eq}} \alpha_{j} s_{j}^{2} = 0
\]

where at least one \( \alpha_{j_{0}} \neq 0 \). Since the columns of \( S_{immo}^{2} \) are linearly independent, the \( j_{0} \) can be chosen \( J_{mob} + 1 \leq j_{0} \leq J_{mob} + J_{sorp} \). Since

\[
s_{j_{0}}^{2} = -\frac{1}{\alpha_{j_{0}}} \sum_{j=J_{mob}+1}^{J_{eq}} \alpha_{j} s_{j}^{2},
\]

the addition of

\[
-\frac{1}{\alpha_{j_{0}}} \sum_{j=J_{mob}+1}^{J_{eq}} \alpha_{j} \begin{pmatrix} s_{j}^{1} \\ s_{j}^{2} \end{pmatrix}
\]

to column \( \begin{pmatrix} s_{j_{0}}^{1} \\ s_{j_{0}}^{2} \end{pmatrix} \) cancels vector \( s_{j_{0}}^{2} \), leaving us with a matrix with \( J_{mob} \) increased by one and \( J_{sorp} \) decreased by one. Obviously the property (34) remains unaffected by this addition.

The process is repeated until the columns of \( \begin{pmatrix} S_{sorp}^{2} & S_{immo}^{2} \end{pmatrix} \) are linearly independent.
Note that the preprocessing does not affect the reaction invariants, since the orthogonal complement of the space generated by the columns of representation (30) and of representation (32) is clearly the same.

Returning to our example (32), we may choose \( j_0 = 1 \) and add \(-1/2\) times column 2 to column 1, and get the new stoichiometric matrix, which is again denoted by \( S_{eq} \),

\[
S_{eq} = \begin{pmatrix}
1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
-\frac{1}{2} & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 2 & \frac{1}{2} & -1
\end{pmatrix}
\]

with \( J_{mob} = 1, J_{sorp} = 1, J_{immo} = 2 \).

5 Numerical results

In the following we consider a test problem adapted from the PHREEQC User’s Guide [11], example 15, which was developed by Tebes-Steven and Valocchi in 1997. The problem covers a variety of processes such as bacterially mediated degradation of organic substrate (kinetic), bacterial cell growth and decay, aqueous speciation (equilibrium), and metal sorption (also assumed to be at equilibrium). Our main focus lies in the measurement of the cpu time reduction achieved by our reduction scheme.

We consider the following, slightly simplified system of reactions, where we have omitted some species (like \( \text{NTA}^{3-} \)) which have very small concentrations.

\[
\begin{align*}
R_1 : & \quad \text{NH}_4^+ & \leftrightarrow & \text{NH}_3 + \text{H}^+ \\
R_2 : & \quad (\text{H}_2\text{O}) & \leftrightarrow & \text{OH}^- + \text{H}^+ \\
R_3 : & \quad \text{CO}_2 + (\text{H}_2\text{O}) & \leftrightarrow & \text{HCO}_3^- + \text{H}^+ \\
R_4 : & \quad \text{HNTA}^{2-} + \text{Co}^{2+} & \leftrightarrow & \text{CoNTA}^- + \text{H}^+ \\
R_5 : & \quad (2\text{Cl}^-) + \text{Co}^{2+} & \leftrightarrow & \text{CoCl}_2(\text{s}) \\
R_6 : & \quad (\text{Na}^+) + \text{CoNTA}^- & \leftrightarrow & \text{NaCoNTA}(\text{s}) \\
R_7 : & \quad \text{HNTA}^{2-} + 1.62\text{O}_2 + 2.42\text{H}^+ & \rightarrow & (0.575\text{C}_5\text{H}_7\text{O}_2\text{N}) + (1.848\text{H}_2\text{O}) + 3.12\text{CO}_2 + 0.424\text{NH}_4^+
\end{align*}
\]

Our scenario is a pulse of water containing \( \text{HNTA}^{2-}, \text{CoNTA}^- \) (NTA = nitritotriacetate) being injected into the 2-D computational domain. \( R_1-R_4 \) are homogeneous equilibrium reactions in the mobile phase. \( R_5-R_6 \) are sorption reactions which are also assumed to be fast enough to be modeled as equilibrium reactions. \( R_7 \) is a kinetic biodegradation reaction modeled with the Monod approach. The rate for the cell growth, including a death rate, but no biomass limitation, is \( R_8 = 0.24R_7 - 8.75 \cdot 10^{-5}c_X \) with

\[
R_7 = 0.95c_X \frac{c_{\text{HNTA}^{2-}}}{K_{\text{HNTA}^{2-} + c_{\text{HNTA}^{2-}}}} \frac{c_{\text{O}_2}}{K_{\text{O}_2 + c_{\text{O}_2}}} \frac{c_{\text{H}^+}}{K_{\text{H}^+ + c_{\text{H}^+}}}
\]

(see [8], sec. 5, for the handling of Monod reactions in the reduction scheme). The species in brackets are not included in the numerical simulation: \( \text{H}_2\text{O}, \text{Cl}^-, \text{Na}^+ \).
are assumed to be available in large amounts everywhere during the computation and not significantly affected by the reactions, and C₅H₇O₂N only occurs as a product and is therefore dropped from the list of species. Notice that of course the reduction method could also handle the full system if the assumption of certain concentrations to be almost constant seems inappropriate.

The stoichiometric matrix reads

\[
S = \begin{pmatrix}
S_1^1 & S_1^2 \\
S_2^1 & S_2^2
\end{pmatrix} = \begin{pmatrix}
S_{mob}^1 & S_{sorp}^1 & S_{neq}^1 \\
0 & S_{mob}^2 & S_{sorp}^2 & S_{neq}^2
\end{pmatrix}
\]

where the rows correspond to the \( I = 10 \) mobile species \( \text{NTA}^{2-}, \text{O}_2, \text{H}^+, \text{CO}_2, \text{NH}_4^+, \text{Co}^{2+}, \text{CoNTA}^-, \text{NH}_3, \text{OH}^-, \text{HCO}_3^- \) and the \( \bar{I} = 3 \) immobile species \( \text{CoCl}_2(s), \text{NaCoNTA}(s) \) and biospecies \( X \) (in this order). Species concentrations are measured in \( 10^{-6} \) moles per unit volume for the immobile ones and \( 10^{-6} \) moles per unit volume of the fluid phase for the mobile ones. The Monod parameters of reaction \( R_7, R_8 \) are \( K_{\text{HNTA}^-} = 0.764, K_{\text{O}_2} = 6.25, K_{\text{H}^+} = 0 \). We use the equilibrium constants for reactions \( R_1-R_6 \) of \( 5 \cdot 10^{-4}, 10^{-2}, 0.447, 25, 19, 2 \), respectively. That means that, for example, the third reaction corresponds to \( c_{\text{HCO}_3^-} c_{\text{H}^+}/c_{\text{CO}_2} = 0.447 \).

Let us apply the reduction method. The rank of \( S_1 \) equals 7; hence there are \( 10 - 7 = 3 \) decoupled linear equations for reaction invariants \( \eta_1, \eta_2, \eta_3 \). Vector \( \xi \) has seven components, four for \( \xi_{mob} \), two for \( \xi_{sorp} \), and one for \( \xi_{neq} \). The rank of \( S_2 \) equals three; hence, there are no \( \bar{\eta} \) variables, and \( \bar{\xi} \) consists of three components, two for \( \bar{\xi}_{sorp} \) and one for \( \bar{\xi}_{neq} \). The remaining nonlinear system of PDEs due to formulation (27) consists of three equations; two for \( \xi_{sorp} \) and one for \( \xi_{neq} \), into which the seven local equations (four for \( \xi_{mob} \), two for \( \xi_{sorp} \), and one for \( \xi_{neq} \)) are substituted.

The algorithm by Molins et al. ([10], eq. (24)) leads to a nonlinear coupled problem consisting of six equations; \( J_{neq} = 2 \) nonlinear equations and \( I + \bar{I} - J - N_{ki} = 13 - 8 - 1 = 4 \) linear component equations. The component equations do not decouple from the rest, since there are immobile species with non-constant activity participating in equilibrium reactions. For such a problem the authors suggest splitting techniques to decouple the two nonlinear PDEs from the linear equations.

For our computation we use a Darcy flux of 1 length unit/time unit, a volumetric water content of 0.4, a longitudinal/transversal dispersion length of 0.05/0.02
length units and a rectangular computational domain of $10 \times 6$ length units.

For the 2-D simulation we use the software basis M++ by Wieners [18]. The discretization uses bilinear finite elements with mass lumping for the mass term and for the reactive terms, and an implicit Euler stepping in time. The non-linear timesteps are linearized by a Newton-Armijo method, and the linear solver is GMinRes(k) with Jacobi preconditioner.

The computation covers 120 time steps (40 with $\Delta t = 0.1$ followed by 80 with $\Delta t = 0.2$ time units) and uses 8580 elements.

We are comparing the cpu times for the reduced scheme with the nonreduced scheme. For the nonreduced scheme, the kinetic reactions are modeled as 'fast' kinetic reactions. Table 1 shows a reduction of cpu time by a factor of 4.2 from 582 minutes to 137 minutes. Notice that the reduced scheme performs slightly fewer Newton steps than the nonreduced scheme, which is due to the stiffness/nonlinearity caused by larger kinetic rate coefficients used for modeling the equilibrium reactions in the nonreduced scheme.

We observed that for an increasing number of elements (mesh refinement), the relative gain of cpu time improves beyond the values from table 1. This can be explained by the fact that for larger meshes, the time for the linear solver dominates the time for matrix assembling, and the time for the linear solver is stronger reduced by the reduction scheme than the time for assembling. However, both the time for solving and the time for assembling are reduced by the scheme. Note that the solution of the local problems of dimension $7 \times 7$ at each grid point require less than 10 percent of the total cpu time.

Table 1: Comparison of cpu time (in minutes) for the simulation and the average number of Newton steps per time step for the nonreduced and the reduced scheme for the problem of section 5. The 'local Newton steps' refer to the average number of Newton steps for the local problems (see section 3.5) per time step per grid point.

<table>
<thead>
<tr>
<th></th>
<th>nonreduced scheme</th>
<th>reduced scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>cpu assembling</td>
<td>104.5</td>
<td>42.2</td>
</tr>
<tr>
<td>cpu lin. solver</td>
<td>473.4</td>
<td>81.9</td>
</tr>
<tr>
<td>cpu local problems</td>
<td>—</td>
<td>12.9</td>
</tr>
<tr>
<td>total cpu</td>
<td>582.1</td>
<td>137.4</td>
</tr>
<tr>
<td>Newton steps</td>
<td>3.40</td>
<td>3.03</td>
</tr>
<tr>
<td>loc. Newton steps</td>
<td>—</td>
<td>1.72</td>
</tr>
</tbody>
</table>
6 Application of the reduction scheme to other examples

6.1 Degradation of EDTA (equilibrium and non-equilibrium sorption and Monod kinetics)

In this section we consider an example by [2] which was also considered in [4]. It consists of 14 species: 6 mobile species Co(II), Co(II)EDTA, Fe(III)EDTA, EDTA, Co(III)EDTA, O₂, and 7 immobile chemical species S⁻Co, S⁻, S⁺Co(II)EDTA, S⁺, S⁺Fe(III)EDTA, S⁺EDTA, S⁺Co(III)EDTA, and one (immobile) biospecies X. There are 10 reactions considered: five equilibrium reactions (which are all heterogeneous), three kinetic chemical reactions and two biological Monod reactions. We consider this example as a 'hard' problem for the reduction method, since there are many heterogeneous equilibrium reactions and no homogeneous ones. Let us apply the reduction mechanism of section 3.3 to this chemical reaction network with equilibrium/non-equilibrium reactions and mobile/immobile species, including biological decay:

As in section 5, we follow the idea presented in the examples 2 and 3 in [8] to represent each Monod reaction term by two reactions each, one to describe the rate term for the chemical species and one for the growth rate of the biospecies. The stoichiometric matrix has \( I + \bar{I} = 6 + 8 = 14 \) rows (each row corresponding to one species, in the above order of species) and \( J = 12 \) columns and reads as follows:

\[
S = \left( \begin{array}{c}
\frac{S_1}{S_2} = \\
\left( \begin{array}{cccccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{array} \right) \\
\end{array} \right)
\]

Comparing this stoichiometric matrix with the scheme (20) we see that the blocks \( S_{\text{mob}}^1, S_{\text{immo}}^2 \) are empty (i.e., \( J_{\text{mob}} = J_{\text{immo}} = 0, J_{\text{eq}} = J_{\text{sorp}} = 5 \)). We observe that all columns in \( S_{\text{eq}}^1 = S_{\text{sorp}}^1 \) and all columns in \( S_{\text{eq}}^2 = S_{\text{sorp}}^2 \) are linearly independent. Hence, the condition (21) of our reduction method is met; no preprocessing is required.

Let us determine the size of the different blocks of the reduced scheme. The column rank of \( S_1, S_2 \) is \( J_1 = J_2 = 6 \), each. Therefore \( S_1^*, S_2^* \) have 6 columns each.
To get a complete basis of the space of the \( c_i, \tilde{c}_i \), which is \( \mathbb{R}^I = \mathbb{R}^6 \), \( \mathbb{R}^I = \mathbb{R}^8 \), respectively, we have to add \( I - J_1 = 0 \) columns to \( S_1 \) and \( I - J_2 = 2 \) columns to \( S_2 \). Hence, there is no \( \eta \)-block and an \( \bar{\eta} \)-block of size 2. The \( J_{\text{sorp}} = 5 \) linearly independent columns in \( S_{\text{sorp}}^1 \), \( S_{\text{sorp}}^2 \) yield 5 generalized sorption equations for \( \xi_{\text{sorp}} \) and 5 local algebraic equations for \( \bar{\xi}_{\text{sorp}} \). The difference between the column rank of \( S_1 \) and \( S_{\text{eq}} \) is \( J_{\text{neq}} = J_1 - J_{\text{sorp}} = 1 \), therefore the \( \bar{\xi}_{\text{neq}} \)-block of PDEs has size 1. Analogously, the difference between the column rank of \( S_2 \) and \( S_{\text{eq}}^2 \) is \( J_2 - J_{\text{sorp}} = 1 \), therefore the \( \xi_{\text{neq}} \)-block of ODEs has size 1.

After elimination of the local equations, the reduced system consists of \( J_{\text{sorp}} + J_{\text{neq}}^I = 5 + 1 = 6 \) coupled nonlinear equations for \( \xi_{\text{sorp}}, \xi_{\text{neq}} \), while the given problem consists of 14 coupled equations. The local problem for \( \xi_{\text{sorp}}, \xi_{\text{neq}} \) is of size \( J_{\text{sorp}} + J_{\text{neq}}^I = 6 \) at each grid point.

The method by Friedly and Rubin [6] would lead to \( J + J_1 = 12 + 6 = 18 \) coupled equations for this example, which is more than the size of the original problem. To this system one could apply a splitting technique, as proposed in that paper, or a DSA-like technique, as proposed in [8], sec. 3.3 and 3.4, reducing the size of the coupled system from 18 to \( J_1 = 6 \). For this special example, a DSA could be applied such that no coupling terms under the transport operator would occur [sec. 3.4], which is in general (e.g., if the system is coupled to any equilibrium reaction in the mobile phase) not the case.

Note that the application of the method by Molins et al. [10] to this problem is not so obvious, since in that publication the assumption is made that in each reaction only one immobile species occurs. Just by operations on the columns of \( S \), this structure seems not attainable.

### 6.2 A large reaction network

The decrease of the number of coupled equations is especially large if there are many homogeneous equilibrium equations, since \( J_{\text{mob}} \) and \( J_{\text{immo}} \) do not affect the size of the resulting coupled problem. To show this, let us consider an example taken from [4]:

hom. equilibrium reactions in the mobile phase:

\[
\begin{align*}
(R1) & \quad C_5 + C_6 \leftrightarrow C_7 & (R11) & \quad 2C_1 \leftrightarrow 2C_2 + C_{18} \\
(R2) & \quad C_2 + C_5 + C_6 \leftrightarrow C_8 & (R12) & \quad C_2 + C_4 + C_5 \leftrightarrow C_{19} \\
(R3) & \quad C_6 \leftrightarrow C_2 + C_9 & (R13) & \quad C_4 \leftrightarrow C_2 + C_{20} \\
(R4) & \quad C_1 + C_2 + C_5 \leftrightarrow C_{11} & (R14) & \quad C_4 \leftrightarrow 2C_2 + C_{21} \\
(R5) & \quad C_1 + C_5 \leftrightarrow C_2 + C_{12} & (R15) & \quad C_4 \leftrightarrow 3C_2 + C_{22} \\
(R6) & \quad C_1 + C_5 \leftrightarrow 2C_2 + C_{13} & (R16) & \quad C_2 + C_5 \leftrightarrow C_{23} \\
(R7) & \quad C_1 \leftrightarrow C_2 + C_{14} & (R17) & \quad 2C_2 + C_5 \leftrightarrow C_{24} \\
(R8) & \quad C_1 \leftrightarrow 2C_2 + C_{15} & (R18) & \quad 3C_2 + C_5 \leftrightarrow C_{25} \\
(R9) & \quad C_1 \leftrightarrow 3C_2 + C_{16} & (R19) & \quad 4C_2 + C_5 \leftrightarrow C_{26} \\
(R10) & \quad C_1 \leftrightarrow 4C_2 + C_{17} & (R20) & \quad C_2 + C_{27} \leftrightarrow C_{28}
\end{align*}
\]
heterogeneous equilibrium reactions:

(R21) \( S_1 \rightleftharpoons S_2 + C_2 \)
(R22) \( S_1 + C_2 \rightleftharpoons S_3 \)
(R23) \( S_1 + 3C_2 + C_5 \rightleftharpoons S_4 \)
(R24) \( C_6 + 2 \text{site-}C_{30} \rightleftharpoons \text{site-}C_6 + 2C_{30} \)

hom. algebraic equation in the immobile phase:

(R0) \( M = S_1 + S_2 + S_3 + S_4 + S_5 + S_6 + S_7 + S_8 \)

kinetic reactions:

(R25) \( M + 3C_2 \rightleftharpoons C_1 \)
(R26) \( C_3 \rightleftharpoons C_4 + C_5 \)
(R27) \( C_1 + C_5 \rightleftharpoons C_{10} \)
(R28) \( S_1 + C_1 + C_2 + C_5 \rightleftharpoons S_5 \)
(R29) \( S_1 + C_2 + C_4 + C_5 \rightleftharpoons S_6 \)
(R30) \( S_1 + C_4 \rightleftharpoons S_7 + C_2 \)
(R31) \( S_1 + C_2 + C_5 + C_6 \rightleftharpoons S_8 \)
(R32) \( C_{29} + 2 \text{site-}C_{30} \rightleftharpoons \text{site-}C_{29} + 2C_{30} \)

The system contains aqueous complexation, ion exchange, mineral dissolution reactions. For further explanations see [4].

Since the algebraic equation (R0) describing the formation of the adsorbing sites \( S_1 \ldots S_8 \) has not the structure of a mass action law, we will use this algebraic equation to eliminate the variable \( M \) from the system (i.e., we substitute \( M \) in the kinetic rate term for \( R_{25} \) by the sum of the \( S_i \)) before we apply our reduction scheme.

The remaining system consists of \( I = 30 \) mobile species \( C_1, \ldots, C_{30} \), \( \bar{I} = 11 \) immobile species \( S_1, \ldots, S_8, \text{site-}C_{29}, \text{site-}C_{29}, \text{site-}C_{30} \), and \( J = 32 \) reactions (R1)-(R32), from which \( J_{\text{neq}} = 8 \) are kinetic, \( J_{\text{sorp}} = 4 \) are heterogeneous equilibrium reactions \( J_{\text{mob}} = 20 \) are homogeneous equilibrium reactions \( J_{\text{immo}} = 0 \). The stoichiometrix matrix \( S \) is displayed in Table 2.

Let us determine the size of the resulting system:

The column rank of block \( S_{eq}^1 \) is 23, i.e. one smaller than the number of columns. Hence, in the sense of section 4, we get by taking a linear combination of columns 21 and 22 a stoichiometric matrix with \( J_{\text{mob}} = 20, J_{\text{sorp}} = 3, J_{\text{immo}} = 1 \). After this preprocessing, the columns of matrix \( \left( S_{\text{sorp}}^1 S_{\text{immo}}^1 \right) \) are linearly independent. The column rank of \( S_1 \) is \( J_1 = 29 \). So there is \( I - J_1 = 1 \) linear decoupled scalar equation for \( \eta \), and there is a coupled system of 9 PDEs \( (J'_{\text{neq}} = J_1 - J_{\text{mob}} - J_{\text{sorp}} = 6 \) for \( \xi_{\text{neq}} \) and \( J_{\text{sorp}} = 3 \) for \( \xi_{\text{sorp}} \)).

In the originally given formulation, the problem consisted of 41 coupled equations; \( I = 30 \) PDEs and \( \bar{I} = 11 \) ODEs.

7 Summary and Outlook

We have proposed a new reduction scheme for multicomponent reactive transport problems, which is able to handle coupled mobile and immobile species and mixed
equilibrium and nonequilibrium reactions. The basic philosophy is to figure out how much the size of the nonlinear system of differential equations can be reduced without using splitting techniques (SNIA/SIA) or imposing restrictions on the stoichiometry.

A special property of the scheme is that it allows to eliminate local equations resulting from equilibrium or nonequilibrium reactions, and that this elimination process does not lead to coupling terms under the transport operator, which would cause many nonzero entries in the Jacobian. The sparsity of the system matrix can be exploited if iterative linear solver are used, which seems reasonable at least for problems with fine discretization. Under rather general conditions, assuming mass action law for the equilibrium reactions, the applicability of the reduction method and the existence of the resolution functions can be guaranteed. Application to problems including minerals are possible as long as the mineral concentrations do

Table 2: Stoichiometric matrix $S$ for the problem from section 6.2. Only the nonzero entries are displayed.

$$
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -2 \\
1 & -1 & 1 & 2 & 1 & 2 & 3 \\
-1 & 1 & 2 & -1 & 1 & 2 & 3 \\
-2 & 4 & 2 & -1 & 1 & 2 & 3 \\
-1 & -1 & 1 & 2 & -1 & 1 & 2 \\
-1 & -1 & 1 & 2 & -1 & 1 & 2 \\
-1 & -1 & 1 & 2 & -1 & 1 & 2 \\
-1 & -1 & 1 & 2 & -1 & 1 & 2 \\
\end{pmatrix}
\begin{pmatrix}
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
\end{pmatrix}
\begin{pmatrix}
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
1 & -1 & 1 & 3 \\
\end{pmatrix}
$$
not vanish.

The number of coupled nonlinear PDEs of the reduced scheme is smaller than the number of given species, and also usually smaller than the number of reactions: It lies between $J_{\text{sorp}}$ and $J_{\text{sorp}} + J_{\text{neq}}$, where $J_{\text{sorp}}$ is the number of heterogeneous equilibrium reactions and $J_{\text{neq}}$ is the number of kinetic reactions. Hence, if no heterogeneous equilibrium reactions are present, then the number of remaining coupled equations is less than or equal to the number of kinetic reactions.

No general rule can be given whether the method by Molins et al. [10] or the one proposed here leads to a system with a smaller number of coupled PDEs, as long as no artificial splitting technique is applied to the resulting system. Also be aware that not only the number of remaining equations, but also the choice of the linear solver, the sparsity pattern of the Jacobian, the stiffness of the problem, and the size and the difficulty of the local problems can affect the total cpu time.

Future work could try to consider the generalization of the reduction scheme to situations where constant activity species (minerals) are present whose concentrations may reach zero at some places within the computational domain. Reactions and equilibrium conditions including such species are more difficult to describe, for example, by a restrained minimization principle for the free Gibbs energy, where the equality constraints are the equations of mass conservation (PDEs) and the inequality constraints describe the nonnegativity of the species concentrations. Using this approach, the reduction method proposed in this article could be applied to the set of equality constraints of the minimization problem.
Appendix

Proof of the theorem in section 3.4

part (a), (i) We have to check that
\[
\dot{Q}_{eq}(\xi_{mob}, \tilde{\xi}_{sorp}, \tilde{\xi}_{immo}, \xi_{sorp}, \xi_{neq}, \tilde{\xi}_{neq}) = S'_{eq} \ln(c, \bar{c}) - K = 0
\]  
(35)
Theorem in section 3.4

with \(c, \bar{c}\) given by (28) has a resolution function
\[
(\xi_{sorp}, \xi_{neq}, \tilde{\xi}_{neq}) \mapsto (\xi_{mob}, \tilde{\xi}_{sorp}, \tilde{\xi}_{immo}).
\]  
(36)

Due to the implicit function theorem we have to verify that the matrix \(\partial Q_{eq}/\partial (\xi_{mob}, \tilde{\xi}_{sorp}, \tilde{\xi}_{immo})\) is invertible. We compute the block matrix
\[
\frac{\partial Q_{eq}}{\partial (\xi_{mob}, \tilde{\xi}_{sorp}, \tilde{\xi}_{immo})} = \begin{pmatrix}
(S'_{eq})^t D_c S'_{mob} & (S'_{eq})^t D_{\tau} S'_{sorp} & (S'_{eq})^t D_{\tau} S'_{immo} \\
(S'_{mob})^t D_c S'_{mob} & 0 & 0 \\
0 & (S'_{sorp})^t D_{\tau} S'_{sorp} & (S'_{immo})^t D_{\tau} S'_{immo}
\end{pmatrix}
\]  
(37)
where \(D_c, D_{\tau}\) are the diagonal matrices
\[
D_c = \begin{pmatrix}
1/c_1 & 0 & \ldots & 0 \\
0 & 1/c_2 & \ldots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & \ldots & 0 & 1/c_I
\end{pmatrix}, \quad D_{\tau} = \begin{pmatrix}
1/\tau_1 & 0 & \ldots & 0 \\
0 & 1/\tau_2 & \ldots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & \ldots & 0 & 1/\tau_T
\end{pmatrix}.
\]
The lower right block in (37) can be simplified:
\[
\begin{pmatrix}
(S'_{sorp})^t D_{\tau} S'_{sorp} \\
(S'_{immo})^t D_{\tau} S'_{immo}
\end{pmatrix} = \begin{pmatrix}
S'_{sorp} & S'_{immo}
\end{pmatrix}^t D_{\tau} \begin{pmatrix}
S'_{sorp} \\
S'_{immo}
\end{pmatrix}
\]  
(38)
Thanks to assumption (21), the columns of \(S'_{sorp}, S'_{immo}\) are linearly independent.

Due to the same argument used in [8], appendix, proof of the theorem, part (i), matrix (38) is symmetric positive definite and has a complete basis of eigenvectors with positive eigenvalues. The same argument holds for block \(S'_{mob}^t D_c S'_{mob}\) in (37). So both diagonal blocks in (37) are regular. Therefore matrix (37) is regular.

(ii) We have to prove that there is a resolution function \((\xi_{mob}, \xi_{sorp}, \xi_{neq}, \tilde{\xi}_{sorp}, \tilde{\xi}_{immo}) \mapsto \tilde{\xi}_{neq}\) of equation \(Q_{eq} = 0\), where
\[
Q_{eq} := \tilde{\xi}_{neq} - \tilde{\xi}_{neq, old} - \Delta t A_{neq}^2 R_{neq}(c, \tau)
\]  
(39)
if the implicit Euler method is used, and
\[
Q_{eq} := \xi_{neq} - \xi_{neq, old} - \Delta t A_{neq}^2 R_{neq}(c, \tau_{old})
\]  
(40)
if the explicit Euler method is used (\(\xi_{neq, old}, c, \tau_{old}\) taken from the previous timestep). For (40) there is apparently even an explicit resolution function. For (39) we get
\[
\frac{\partial Q_{eq}}{\xi_{neq}} = \text{Id} - \Delta t A_{neq}^2 \frac{\partial R_{neq}}{\partial \tau} S'_{neq}^2
\]
which is a regular matrix at least if $\Delta t < \| A_{\text{nqq}}^2 \partial R_{\text{nq}} / \partial t \|^{-1}$.

(iii) We prove the existence of a local resolution function

$$(\xi_{\text{sor}}, \xi_{\text{nq}}) \mapsto (\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}}, \xi_{\text{nq}})$$

of the equations $\hat{Q}_{\text{eq}} = 0, Q_{\text{nq}} = 0$. We compute

$$\frac{\partial (Q_{\text{nq}}, \hat{Q}_{\text{eq}})}{\partial (\xi_{\text{nq}}, \xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}})} = \begin{pmatrix} \frac{\partial Q_{\text{nq}}}{\xi_{\text{nq}}} & \frac{\partial Q_{\text{nq}}}{\xi_{\text{mob}}} \\ \frac{\partial Q_{\text{nq}}}{\xi_{\text{mob}}} & \frac{\partial Q_{\text{nq}}}{\xi_{\text{sor}}} & \frac{\partial Q_{\text{nq}}}{\xi_{\text{imm}}} \end{pmatrix} = \begin{pmatrix} \partial(Q_{\text{nq}}, \hat{Q}_{\text{eq}}) / \partial(\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}}) \end{pmatrix}$$

$$= \begin{pmatrix} \text{Id} - \Delta t A_{\text{nq}}^2 / \partial R_{\text{nq}} \partial S_{\text{nq}}' / \partial t S_{\text{sor}} / \partial \hat{Q}_{\text{eq}} / \partial (\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}}) \\ 0 \end{pmatrix} \begin{pmatrix} \hat{Q}_{\text{eq}} / \partial (\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}}) \end{pmatrix}$$

The two blocks on the diagonal are regular due to (i), (ii). The regularity of the whole block matrix for $\Delta t$ sufficiently small follows as in the proof of the theorem, part (iii).

(b) We want to verify the existence of a resolution function

$$(\xi_{\text{sor}}, \xi_{\text{nq}}, \xi_{\text{nq}}) \mapsto (\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}})$$

of (35). We get

$$\frac{\partial \hat{Q}_{\text{eq}}}{\partial (\xi_{\text{mob}}, \xi_{\text{sor}}, \xi_{\text{imm}})} = \begin{pmatrix} (S_{\text{mob}}^1)^T D_c S_{\text{mob}}^1 \quad (S_{\text{sor}}^1)^T D_c S_{\text{sor}}^1 \quad 0 \\ (S_{\text{sor}}^1)^T D_c S_{\text{sor}}^1 \quad (S_{\text{sor}}^1)^T D_c S_{\text{sor}}^1 \quad 0 \\ 0 \quad 0 \quad (S_{\text{imm}}^2)^T D_c S_{\text{imm}}^2 \end{pmatrix}$$

The structure of this matrix is similar to that of (37), and therefore the rest of the proof is similar to (a). □
Notation.

I  number of mobile species
$\bar{I}$  number of immobile species
J  number of reactions
$J_{eq}$  number of equilibrium reactions
$J_{neq}$  number of kinetically controlled reactions
$J_1$  number of linearly independent reactions, if all immobile species are ignored (rank of matrix $S_1$)
$J_2$  number of linearly independent reactions, if all mobile species are ignored (rank of matrix $S_2$)
$J_{mob}, J_{immo}$  number of heterogeneous equil. reactions in the mobile/immobile phase
$J_{sorp}$  number of heterogeneous equilibrium reactions
$c$  concentration vector of mobile species, size $I$
$c^*$  concentration vector of immobile species, size $\bar{I}$
L  a linear transport (advection-dispersion-diffusion) operator
R  vector of reaction expressions, size $J$
$R_{eq}, R_{neq}$  vector of equilibrium/non-equilibrium reaction expressions, size $J_{eq}, J_{neq}$
$Q_{eq}$  vector of equilibrium expressions after ln is applied to each component $Q_j, j = 1, \ldots, J_{eq}$
$S$  $(I + \bar{I}) \times J$-matrix of stoichiometric coefficients $s_{ij}$
$S_1$  $I \times J$-submatrix of $S$ of stoich. coeff. of mobile species
$S_2$  $\bar{I} \times J$-submatrix of $S$ of stoich. coeff. of immobile species
$S_{eq}, S_{neq}$  $(I + \bar{I}) \times J_{eq} \text{ and } (I + \bar{I}) \times J_{neq}$-submatrices of $S$ corresponding to equilibrium/nonequilibrium reactions
$S_{11}$  $I \times J_{mob}$-submatrix of $S_1$ of linearly independent columns
$S_{21}$  $\bar{I} \times J_{immo}$-submatrix of $S_2$ of linearly independent columns
$S_{11}^{mob}, S_{11}^{sorp}$  submatrices of $S_{11}^{eq}$ defined by (20)
$S_{21}^{mob}, S_{21}^{sorp}$  submatrices of $S_{21}^{eq}$ defined by (20)
$S_{11}^{eq}$  $J_{eq} \times I \times J_{neq} \times I$-submatrix of $S_1$ for the equilibrium/kinetic reactions
$S_{21}^{eq}$  $J_{eq} \times \bar{I} \times J_{neq} \times \bar{I}$-submatrix of $S_2$ for the equilibrium/kinetic reactions
$S_{neq}^{eq}, S_{neq}^{sorp}$  submatrices of $S_{neq}^{eq}, S_{neq}^{sorp}$, resp., defined by (22)
$A_i$  matrix defined by (9), substructured as in (23), (24)
Id$_k$  The $k \times k$ identity matrix
$\top$  a matrix $M^\top$ consists of a maximum system of linearly independent columns that are orthogonal to the columns of matrix $M$
$\upsilon$  $M^\top$ is the transpose of $M$
$\| \cdot \|$  any submultiplicative matrix norm
$\Delta \upsilon$  timestep size
$x$  an arbitrary point in the computational domain

Acknowledgments. The coding of the reduction method using the software basis M++ was done by J. Hoffmann in the course of his diploma thesis at the Institute of Applied Mathematics, Erlangen.
References


