PRECONDITIONING A COUPLED MODEL
FOR REACTIVE TRANSPORT IN POROUS MEDIA [*]

LAILA AMIR[†] AND MICHEL KERN[‡]

Key words. Reactive transport, porous media, precondtioning, non-linear systems

AMS subject classifications. 76S05, 65H10, 65F10, 65N22,

Abstract. We study numerical methods for solving reactive transport problems in porous media that allow a separation of transport and chemistry at the software level, while keeping a tight numerical coupling between both subsystems. We give a formulation that eliminates the local chemical concentrations and keeps the total concentrations as unknowns, then recall how each individual subsystem can be solved. The coupled system is solved by a Newton–Krylov method. The block structure of the model is exploited both at the nonlinear level, by eliminating some unknowns, and at the linear level by using block Gauss-Seidel or block Jacobi preconditioning. The methods are applied to a 1D case of the MoMaS benchmark.

1. Introduction. Reactive transport in porous media studies the coupling of reacting chemical species with mass transport in the subsurface, see [5, sec. 7.9] for an introduction, or [3, 7] for more comprehensive references. It plays an important role in several applications when modeling subsurface flow [61, 60, 72] :

- chemical trapping is one of the mechanism by which the safety of geological sequestration of carbon dioxide in deep saline aquifers can be ensured [4, 24, 49, 54, 63]
- nuclear waste storage is based on a multiple barrier concept so as to delay the arrival of radionuclides in the bio-sphere. The concrete barrier that seals the repository may be attacked by oxidized compounds, and again its safety needs to be assessed [18, 42, 55];
- bio-geochemistry involves interactions with organic chemicals, and is important for studies of soil pollution, and also for implementing bio-remediation [15, 44];
- chemical reactions also play a role in some studies of hydro-geothermal reservoirs, as mineral dissolution and precipitation effects could have a major impact on their long-term performance [68, 69]

This paper studies numerical methods for the simulation of reactive transport for flow in porous media. We consider only a simplified physical and chemical setup (one phase flow, no mineral reactions, no kinetic reactions) so as to concentrate on the numerical issues related to the coupling between transport and chemistry.

The numerical simulation of reactive transport has been the topic of numerous work. The survey by Yeh and Tripathi [70] has been very influential in establishing a mathematical formalism for setting up models, and also for establishing the "operator splitting" approach (see below) as a standard. More recent surveys, detailing several widely used computer codes and their applications, can be found in the book [72] and the article [61].

It has been traditional to distinguish two main approaches for solving reactive transport problems:

Sequential Iterative Approach approaches (SIA, also known as operator splitting) where transport and chemistry are solved alternatively [14, 39, 45, 56, 71]. This has the advantage that a code for reactive transport can be built from pre-existing transport

[†]Faculté des sciences et techniques, UCAM, Marrakech, Morocco, `l.amir@uca.ma`
[‡]Centre de recherche Inria de Paris, 2 rue Simone Iff, 75012 Paris, France, `michel.kern@inria.fr`

and chemistry codes [31], and that no global system of equations needs to be solved. On the other hand, the splitting between chemistry and transport may restrict the time step in order to ensure convergence of the method, and the splitting may also introduce mass errors that need to be controlled [64]. As the above references show, the method can be quite successful if it is implemented carefully.

Direct substitution approaches (known as DSA) where the chemical mass action laws are substituted into the transport equations [24, 26]. Here the balance is the opposite of what it was for SIA: the method does not introduce spurious mass errors, and can converge with large time steps, but it leads to a large and difficult to solve system of non-linear equations coupling all the chemical species at all grid points. This system needs to be solved efficiently for DSA to be a viable option (for instance, the code PFLOTRAN, described in [26], uses PETSc to solve the non-linear system in parallel, resulting to highly efficient parallel implementation).

However, several methods have recently been introduced that do not easily fall in the above categories: they do solve the coupled problem globally (and as such have to solve a large non-linear problem), but they do not substitute the mass action law into the transport equation. Instead they solve some form of the local chemical system, and use that solution as an additional term in a transformed form of the transport. The method by Knabner et al. [27, 37, 38] starts by a reduction step, so as to obtain as small a linear system as possible. The chemical concentrations are then eliminated locally. Erhel et al. [16, 21, 22] formulate the coupled problem as a Differential Algebraic system, so that the high quality software for such system can be used. Work by Amir and Kern [2], which forms the basis of the present work, tries to combine the best features of SIA (separate software modules for chemistry and transport) and DSA (global numerical coupling), and solves the nonlinear coupled system by a Newton–Krylov method.

In addition to the surveys mentioned above, the methods have been compared in several studies or benchmarks [11, 43, 53]. It is fair to say that no method or code emerges as a clear winner for all situations.

We finish this short (and far from exhaustive) review of the literature by noting that most of the work cited deal with one-phase flow and transport. The methods have recently been extended to the case of two-phase flow [1, 24, 54, 58] .

In previous work [2], we introduced a global approach that couples chemistry and transport, but the equations of chemistry are not directly substituted in the transport equation. Rather, an implicitly defined function is used to represent the effect of the chemistry, so that chemistry is treated as a block box. The transport and chemistry codes remain separate, which leads to a natural block structure in the Jacobian matrix. The coupled problem is solved by a Newton-Krylov method.

This paper presents several improvements to the method proposed in [2]. Its main contribution is a more systematic study of the nonlinear problem for the coupled system. We also propose a more intrinsic way for solving the chemical equilibrium system that does not require an a priori separation of the species between principal and secondary.

The convergence of Krylov solvers can be slow, so we compare several ways for accelerating the convergence of the basic method: by eliminating variables involved in the transport step, we get rid of the main source of ill conditioning. We will see that this is related to a block preconditioning of the Jacobian of the full system by Jacobi or Gauss–Seidel methods. This provides an extension to multi-component transport of previous work by Kern and Taakili [34] that deals with preconditioning the model of one species undergoing sorption.

We distinguish between two types of preconditioning :

- Linear preconditioning where the linear system is preconditioned respecting the

block structure of the Jacobian matrix

- Nonlinear preconditioning where the original nonlinear system is replaced by a new one by eliminating some unknowns.

An outline for the rest of this paper is as follows: we set up the mathematical model in section 2, and show how to reduce the problem by eliminating the chemical concentrations. Numerical methods for solving the local chemical equilibrium problem and the advection-diffusion equations for transport are the topic of sections 3.1 and 3.2 respectively. Section 4 deals with the formulation of the coupled problem, and its solution by a Newton-Krylov method. Preconditioners for the linear system and elimination methods for the non-linear system are detailed in section 5. Finally, in section 6 the methods are validated on the 1D MoMaS reactive transport benchmark, which is a fairly difficult test case for reactive transport [11].

2. Reactive transport model. We consider a set of species subject to transport by advection and diffusion and to chemical reactions in a porous medium. Transport of a chemical species in a domain $\Omega \subset \mathbf{R}^d$ ($d = 1, 2, 3$, though in this work we only consider the 1D case) is described by an advection–diffusion equation for its concentration $c : \Omega \times]0, T_f[\to \mathbf{R}$ (dispersion could also be included) :

$$(2.1) \qquad \phi \frac{\partial c}{\partial t} + \nabla. \left(uc - D\nabla c \right) = \phi q \qquad \text{on } \Omega \times]0, T_f[,$$

where ϕ is the porosity (fraction of void in a Representative Elementary Volume available for the flow), u is the Darcy velocity (we assume here permanent flow, so that u is considered as known), D is a diffusion–dispersion coefficient and q is given source term. We assume that the diffusion coefficient is independent of the species. This is a strong restriction on the model, but one that is commonly assumed to hold [2, 37, 52, 70]. The model is completed by appropriate initial and boundary conditions. We denote by $\mathcal{L}$ the spatial part of the advection-diffusion operator:

$$\mathcal{L}c = \nabla. \left(uc - D\nabla c \right).$$

The chemical phenomena will involve both homogeneous and heterogeneous reactions. Homogeneous reactions, in the aqueous phase, include water dissociation, acid–base reactions and oxydo–reduction, whereas heterogeneous reactions occur between the aqueous and solid phases, and include surface complexation, ion exchange and precipitation and dissolution of minerals (see [3] for details on the modeling of specific chemical phenomena). Accordingly, we assume there are N_s mobile species $(X_j)_{j=1,\dots,N_S}$ in the aqueous phase and $\bar{N}_s$ immobile species in the solid phase $(\bar{X}_j)_{j=1,\dots,\bar{N}_S}$, and that there are N_r homogeneous reactions, and $\bar{N}_r$ heterogeneous reactions.

In this work, we only consider equilibrium reactions, which means that the chemical phenomena occur on a much faster scale than transport phenomena, see [50]. This assumption is justified for aqueous phase and ion–exchange reactions, but may not hold for reactions involving minerals. Such reactions should be modeled as kinetic reactions, with specific rate laws [5, 24].

We can write the chemical system as

$$(2.2) \qquad \begin{aligned} \sum_{j=1}^{N_s} (S_{cc})_{ij} X_j &\leftrightarrows 0 \quad i = 1, \dots, N_r, \\ \sum_{j=1}^{N_s} (S_{\bar{c}c})_{ij} X_j + \sum_{j=1}^{\bar{N}_s} (S_{\bar{c}\bar{c}})_{ij} \bar{X}_j &\leftrightarrows 0 \quad i = 1, \dots \bar{N}_r, \end{aligned}$$

or in condensed form

$$\begin{pmatrix} S_{cc} & 0 \\ S_{\bar{c}c} & S_{\bar{c}\bar{c}} \end{pmatrix} \begin{pmatrix} X \\ \bar{X} \end{pmatrix} \leftrightarrows \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{2.3}$$

We let

$$S = \begin{pmatrix} S_{cc} & 0 \\ S_{\bar{c}c} & S_{\bar{c}\bar{c}} \end{pmatrix} \tag{2.4}$$

be the stoichiometric matrix, with the sub-matrices $S_{cc} \in \mathbf{R}^{N_r \times N_s}$, $S_{\bar{c}c} \in \mathbf{R}^{\bar{N}_r \times N_s}$ and $S_{\bar{c}\bar{c}} \in \mathbf{R}^{\bar{N}_r \times \bar{N}_s}$.

We assume that both the global stoichiometric matrix S and the "aqueous" stoichiometric matrix S_{cc} are of full rank. As there will usually be more species than reactions, this just means that all reactions are "independent" (in the linear algebra sense !), and that this is also true of the reactions in the aqueous phase.

Each reaction gives rise to a mass action law, linking the activities of the species. For simplicity, we assume all species follow an ideal model, so that their activity is equal to their concentration. We denote by c_j (resp. $\bar{c}_j$) the concentration of species X_j (resp. $\bar{X}_j$). It will be convenient to write the mass action law in logarithmic form, so for a vector c with positive entries, we denote by $\log c$ the vector with entries $\log c_j$. We then have

$$\begin{pmatrix} S_{cc} & 0 \\ S_{\bar{c}c} & S_{\bar{c}\bar{c}} \end{pmatrix} \begin{pmatrix} \log c \\ \log \bar{c} \end{pmatrix} = \begin{pmatrix} \log K \\ \log \bar{K} \end{pmatrix}, \tag{2.5}$$

where $K \in \mathbf{R}^{N_r}$ and $\bar{K} \in \mathbf{R}^{\bar{N}_r}$ are the equilibrium constants for their respective reactions.

Next, we write the mass conservation for each species, considering both transport by advection and diffusion and chemical reaction terms:

$$\begin{aligned} \phi\partial_t c & + \mathcal{L}c = S_{cc}^T r + S_{\bar{c}c}^T \bar{r}, \\ \phi\partial_t \bar{c} & = S_{\bar{c}\bar{c}}^T \bar{r}, \end{aligned} \tag{2.6}$$

where $r \in \mathbf{R}^{N_r}$ and $\bar{r} \in \mathbf{R}^{\bar{N}_r}$ are vectors containing the reaction rates. Since we assume that all reactions are at equilibrium, these rates are unknown and we now show how they can be eliminated.

2.1. Elimination of the reaction rates – the coupled problem. We follow the approach of Saaltink et al.[52] (see also a more general approach in [37, 38]) by introducing a *kernel matrix* U such that $US^T = 0$, i.e. such that columns of U^T form of basis for the null-space of S. This can be done in several ways (see the above references, and section 3.1). Here we outline how one can compute such a matrix with the same structure as that of S.

LEMMA 2.1. *Assume that the matrix S is as defined in equation (2.4), that it has full rank, and that the submatrix S_{cc} also has full rank. Then there exists a matrix*

$$U = \begin{pmatrix} U_{cc} & U_{c\bar{c}} \\ 0 & U_{\bar{c}\bar{c}} \end{pmatrix}, \qquad \begin{matrix} U_{cc} \in \mathbf{R}^{(N_r-N_s) \times N_s}, \ U_{c\bar{c}} \in \mathbf{R}^{(N_r-N_s) \times \bar{N}_s} \\ U_{\bar{c}\bar{c}} \in \mathbf{R}^{(\bar{N}_r-\bar{N}_s) \times \bar{N}_s} \end{matrix} \tag{2.7}$$

such that $US^T = 0$.

Proof.

The existence of a kernel matrix U is well known (see any standard text on linear algebra, such as [62] or [46]). The content of the lemma is that the kernel matrix can be chosen with a block upper triangular structure.

To see that this is the case, we compute the product US^T. This gives the 3 block equations

$$
\begin{aligned}
U_{cc}S_{cc}^T & & = 0 \\
U_{cc}S_{\bar{c}c}^T & + U_{c\bar{c}}S_{\bar{c}\bar{c}}^T = 0 \\
U_{c\bar{c}}S_{\bar{c}\bar{c}}^T & & = 0.
\end{aligned}
$$

Since we have assumed that both the whole matrix S and the submatrix S_{cc} are of full rank, the sub-matrix $S_{\bar{c}\bar{c}}$ must also be of full rank (because S is block triangular). Thus, kernel matrices U_{cc} and $U_{c\bar{c}}$ with the indicated dimension exist (a specific choice will be made in section 3.1), and there remains to check that we can define $U_{c\bar{c}}$ satisfying the second equation above. But this is true because $S_{\bar{c}\bar{c}}$ is of full rank, and because we assume that $\bar{N}_s \geq \bar{N}_r$, this matrix is surjective so that the transpose of the second equation can be solved for $U_{c\bar{c}}$. □

With the kernel matrix constructed in lemma 2.1, we can eliminate the reaction terms in equation (2.6). We multiply the system on the left by U, to obtain

$$
(2.8) \qquad
\begin{aligned}
\phi\partial_t\left(U_{cc}\,c + U_{c\bar{c}}\,\bar{c}\right) + U_{cc}\,\mathcal{L}c & = 0 \\
\phi\partial_t U_{\bar{c}\bar{c}}\,\bar{c} & = 0.
\end{aligned}
$$

It is natural to define the quantities suggested by the above transformation

$$
(2.9) \qquad
\begin{pmatrix} T \\ \bar{T} \end{pmatrix} = \begin{pmatrix} U_{cc} & U_{c\bar{c}} \\ 0 & U_{\bar{c}\bar{c}} \end{pmatrix} \begin{pmatrix} c \\ \bar{c} \end{pmatrix}.
$$

T and $\bar{T}$ are the total analytic concentration for the mobile and immobile species respectively (these are the same as various total quantities defined in the classical survey by Yeh and Tripathi [70]). We also define the total mobile and immobile concentrations for the species in the aqueous phase

$$
(2.10) \qquad C = U_{cc}\,c, \quad \overline{C} = U_{c\bar{c}}\,\bar{c},
$$

so that the total concentrations are given by

$$
(2.11) \qquad T = C + \bar{C} = U_{cc}\,c + U_{c\bar{c}}\,\bar{c}.
$$

These new unknowns are all of dimension N_c, where $N_c = N_s - N_r$. Because of our assumption that D is the same for all chemical species, multiplication by U commutes with the differential operator, and the system can be rewritten as

$$
(2.12) \qquad
\begin{aligned}
\phi\partial_t C + \phi\partial_t\overline{C} + \mathcal{L}C & = 0, \\
\phi\partial_t\bar{T} & = 0.
\end{aligned}
$$

The coupled system consists of the $(N_s - N_r) + (\bar{N}_s - \bar{N}_r)$ conservation PDEs and ODEs (2.12), together with the $N_r + \bar{N}_r$ mass action laws (2.5) and the relations connecting concentration and totals (2.10), for the $N_s + \bar{N}_s$ concentrations and $2(N_s - N_r) + \bar{N}_s - \bar{N}_r$ totals. Note that the ODEs for $\bar{T}$ are decoupled from the rest of the system. In section 4, we show how to eliminate the individual concentrations from the system, so that only the totals C and $\bar{C}$ remain as unknowns.

REMARK 2.1. *The formulation for the chemical system given in equations* (2.5) *and* (2.9) *generalizes the well known Morel formulation [47], where a set of "principal" species is identified, and the remaining "secondary" species are written in terms of the principal ones.*

The stoichiometric matrix S is split naturally in blocks, the mass action laws allow the elimination of the secondary unknowns, and the conservation laws lead to a non-linear system of equations for the principal species.

In the next section, we show that the "local chemical system" (2.5)–(2.9) can be solved in a similar way, without having first to explicitly identify the principal and secondary species.

3. Methods for solving chemistry and transport. We briefly recall in this section how the chemical equilibrium system, and the transport equation are solved.

3.1. The chemical equilibrium problem. The subsystem formed by the mass action laws (2.5) and the definition of the totals (2.9) is a closed system that enables computation of the individual concentrations c and $\bar{c}$ given the totals T and $\bar{T}$. This is actually the same system that would be obtained in a closed chemical system where now T and $\bar{T}$ would be known as the input total concentrations. This subsystem is what we call in the following the *chemical equilibrium problem*. It is a small (its size is the number of species) nonlinear system, that is notoriously difficult to solve numerically (see for example [13, 17, 27, 41]).

It will be convenient in this subsection to temporarily ignore the distinction between mobile and immobile species. We thus define the vectors

$$\xi = \begin{pmatrix} c \\ \bar{c} \end{pmatrix} \in \mathbf{R}^{N_\xi}, \quad \kappa = \begin{pmatrix} \log K \\ \log \bar{K} \end{pmatrix} \in \mathbf{R}^{N_\kappa} \quad \tau = \begin{pmatrix} T \\ \bar{T} \end{pmatrix} \in \mathbf{R}^{(N_\xi - N_\kappa)}$$

($N_\xi = N_s + \bar{N}_s$ is the total number of species and $N_\kappa = N_r + \bar{N}_r$ is the total number of reactions), and write the problem for the chemical equilibrium as

$$(3.1) \qquad \begin{aligned} S \log \xi &= \kappa, \\ U\xi &= \tau, \end{aligned}$$

with the stoichiometric matrix $S \in \mathbf{R}^{\mathbf{N}_\kappa \times N_\xi}$, and the kernel matrix $U \in \mathbf{R}^{(N_\kappa - N_\xi) \times N_\xi}$ is such that $US^T = 0$.

It has been shown by several authors [2, 13, 27] that taking the logarithms of the concentrations as new unknowns in (3.1) is beneficial from the numerical point of view, as it automatically ensure that all concentrations will be positive (which might otherwise be difficult to enforce), and it makes all the unknowns to be of comparable size (in some cases, such as oxydo–reduction reactions, the concentrations have been seen to vary over several orders of magnitude). We take the same convention to define the exponential of a vector as for the logarithm: for a vector z the vector $\exp(z)$ is defined so that its ith entry is $\exp(z_i)$. By defining $z = \log(\xi) \in \mathbf{R}^{N_\xi}$, system (3.1) thus becomes

$$(3.2) \qquad \begin{aligned} Sz &= \kappa, \\ U \exp(z) &= \tau, \end{aligned}$$

In order to solve system (3.2), we take a cue from the solution of constrained least squares problem (see [8, chap. 5], and also [20] in the same context): we consider the first equation as a constraint, and determine its general solution as the sum of a particular solution, and an unknown of smaller dimension, that will be found by substituting it into the second equation. Following the references above, we first compute a QR decomposition of S^T, as $Q^T S^T = \begin{pmatrix} R \\ 0 \end{pmatrix}$, with $Q = (Q_1, Q_2) \in \mathbf{R}^{N_\xi \times N_\xi}$ orthogonal and $R \in \mathbf{R}^{N_\kappa \times N_\kappa}$ upper triangular (and invertible, as S is assumed to be of full rank).

The columns of $Q_1 \in \mathbf{R}^{N_\xi \times N_\kappa}$ form an orthonormal basis of range S^T, while those of $Q_2 \in \mathbf{R}^{N_\xi \times (N_\xi - N_\kappa)}$ form an orthonormal basis of Ker S. A first consequence is that one may choose $U = Q_2^T$ in the second equation of (3.2).

Now, any $z \in \mathbf{R}^{N_\xi}$ may be written uniquely as $z = Q_1 y_1 + Q_2 y_2$, with $y_1 \in \mathbf{R}^{N_\kappa}$ and $y_2 \in \mathbf{R}^{N_\xi - N_\kappa}$. If we substitute this expression in the first equation of (3.2), we obtain

$$R^T y_1 = \kappa,$$

which determines y_1 uniquely. We set $b_1 = Q_1 R^{-T} \kappa$, so that $z = b_1 + Q_2 y_2$, with b_1 known.

We now use the expression for z in the second equation of (3.2), and we obtain a non-linear system of size $N_\xi - N_\kappa$ for $y_2 \in \mathbf{R}^{N_\xi - N_\kappa}$:

$$(3.3) \qquad H(y_2) \underset{\text{def}}{=} Q_2^T \exp(b_1 + Q_2 y_2) - \tau = 0.$$

This is the system that is to be solved numerically, and that is analogous to the non-linear system for the principal species obtained via the Morel formalism. The number of unknowns has been reduced from the number of species to the number of species minus the number of reactions, that is the number of principal species.

As a side remark, let us notice that the Jacobian matrix for this reduced chemical system has the form

$$J_c = Q_2^T \operatorname{diag}(c)\, Q_2, \quad c = b_1 + Q_2 y_2,$$

and this form is again the same as that obtained from the Morel formulation, with the difference that here matrix Q_2 is orthogonal. This may be important as the Jacobian matrices occuring in the solution of the chemical equilibrium problem may be very ill conditioned, as shown in [40]. Thus using orthogonal matrices limits the ill-conditioning to the diagonal matrix $\operatorname{diag}(c)$ where it is harmless.

To solve the reduced chemical problem (3.3), we used a variant of Newton's method. As is well known Newton's method does not always converge in practice and, especially for a code that is designed to be used in coupling applications, it is essential to ensure that the solver "always" works. We have found that using a globalized version of Newton's method (using a line search, cf. [32]) was effective in making the algorithm converge from an arbitrary initial guess.

We now return to the context of solving the coupled problem, where it is important to distinguish between mobile and immobile species. Indeed, what is needed is the partition of the species between their mobile and immobile forms, rather than the individual concentrations (though they are still needed as intermediate quantities). The totals can be computed a posteriori using their definitions in equation (2.10).

It will be convenient to condense the chemical sub-problem by a function

$$(3.4) \qquad \begin{aligned} \psi_C: \quad & \mathbf{R}^{Nc} \to \mathbf{R}^{Nc} \\ & T \mapsto \psi_C(T) = \bar{C} = U_{c\bar{c}}\, \bar{c} \end{aligned}$$

where $\bar{c}$ is obtained by solving the chemical problem (3.2), given T (and $\bar{T}$, which we take as a constant) for ξ, and computing $\bar{C}$ as indicated above.

3.2. Transport model. The transport of a single species through a 1D porous media $\Omega =]0, L[$, subject to advection and diffusion phenomena, follows the 1D version of the general advection-diffusion equation (2.1):

$$(3.5) \qquad \phi \frac{\partial c}{\partial t} + \frac{\partial}{\partial x}\left(-D\frac{\partial c}{\partial x} + uc\right) = \phi q, \quad 0 < x < L, \quad 0 < t < T_f.$$

With this specialization to 1D, the advection-diffusion operator becomes

$$\mathcal{L}(c) = \frac{\partial}{\partial x}\left(-D\frac{\partial c}{\partial x} + uc\right).$$

The initial condition is $c(x,0) = c_0(x)$ and, in view of the applications, the boundary conditions are a Dirichlet condition (given concentration) $c(0,t) = c_d(t)$ at the left boundary ($x = 0$) and zero diffusive flux $\dfrac{\partial c}{\partial x} = 0$ at the right boundary ($x = L$). We could easily take into account more general boundary conditions.

3.2.1. Discretization in space. We treat the space and time discretization separately, as we will use different time discretizations for the different parts of the transport operator.

For space discretization we use a cell-centered finite volume scheme [23]. The interval $]0, L[$ is divided into N_h intervals $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ of length h_i, where $x_{\frac{1}{2}} = 0, x_{N_h+\frac{1}{2}} = L$. For $i = 1, .., N_h$ we denote by x_i the center and $x_{i+1/2}$ the extremity of the element i. We denote by $c_i, i = 1, .., N_h$ the approximate concentration in cell i.

We write equation (3.5) in the form

$$(3.6) \qquad \phi\frac{\partial c}{\partial t} + \frac{\partial \varphi}{\partial x} = \phi q,$$

and we split the flux $\varphi(x,t) = -D\dfrac{\partial c}{\partial x} + uc = \varphi_d + \varphi_a$ as the sum of a diffusive flux $\varphi_d = -D\dfrac{\partial c}{\partial x}$ and an advective flux $\varphi_a = uc$.

We integrate equation (3.6) over a cell $]x_{i-1/2}, x_{i+1/2}[$, to obtain

$$(3.7) \quad \phi_i h_i \frac{dc_i}{dt} + \varphi_{d,i+\frac{1}{2}} + \varphi_{a,i+\frac{1}{2}} - \varphi_{d,i-\frac{1}{2}} - \varphi_{a,i-\frac{1}{2}} = h_i \phi_i q_i, \qquad i = 2, \ldots, N_h.$$

Our flux approximations come from finite differences. For the diffusive flux, we use harmonic averages for the diffusion coefficient (as used in mixed finite element methods) :

$$(3.8) \qquad \varphi_{d,i+\frac{1}{2}} = -D_{i+\frac{1}{2}}\left(\frac{c_{i+1} - c_i}{h_{i+\frac{1}{2}}}\right)$$

with

$$D_{i+\frac{1}{2}} = \frac{2D_i D_{i+1}}{D_i + D_{i+1}}, \quad D_{\frac{1}{2}} = D_1 \quad D_{N_h+\frac{1}{2}} = D_{N_h} \quad \text{and} \quad h_{i+\frac{1}{2}} = \frac{h_i + h_{i+1}}{2}$$

For the advective flux, we use an upwind approximation, so that (assuming for simplicity that $u > 0$), $\varphi_{a,i+\frac{1}{2}} = uc_i$

These approximations are corrected to take into account the boundary conditions, both at $x = 0$ and at $x = L$. We give a matrix formulation, keeping time continuous for now. Since we will be using different discretizations for the diffusive and advective parts (see next section), we keep the matrices for advection and diffusion separate. With the notation:

$$\alpha_i = \frac{D_{i+\frac{1}{2}}}{h_{i+\frac{1}{2}}} + \frac{D_{i-\frac{1}{2}}}{h_{i-\frac{1}{2}}}, \; \beta_i = \frac{D_{i-\frac{1}{2}}}{h_{i-\frac{1}{2}}}, \; \gamma_i = \frac{D_{i+\frac{1}{2}}}{h_{i+\frac{1}{2}}}, \quad i = 2, \ldots, N_h - 1,$$

we define the matrices (with appropriate modifications for the boundary terms)

$$A_d = \text{tridiag}(\beta_i, \alpha_i, \gamma_i), \quad A_a = \text{tridiag}(-u, u, 0)$$

as well as $M = \text{diag}(\phi_i h_i)$.

The semi-discrete system can be written as

$$(3.9) \qquad\qquad M\frac{dc}{dt} + (A_d + A_a)c = Mq,$$

with he initial condition $c_i = \frac{1}{h_i}\int_{x_{1-1/2}}^{x_{i+1/2}} c_0(x_i)$.

3.2.2. Time discretization. As the transport operator contains both advective and diffusive terms, it makes sense to use different time discretization methods for the different terms. Specifically, the diffusive terms should be treated implicitly, and the advective terms are better handled explicitly. Similarly to what was done above, we discretize the interval $[0, T_f]$ with a time step Δt, which we take as a constant for simplicity, and we denote by c_i^n the (approximate) value of $c_i(n\Delta t)$, and by c^n the corresponding vector.

We compare several methods for discretization in time.

Fully implicit : both the diffusive and advective terms are treated implicitly: at each time step, we solve a linear system

$$(M + \Delta t(A_d + A_a))c^{n+1} = Mc^n + M\Delta t q^n.$$

Explicit advection and implicit diffusion : the diffusive terms are treated implicitly, and advective terms are handled explicitly under a CFL condition. With this scheme, the system to be solved at each time step is

$$(M + \Delta t A_d)c^{n+1} = (M - \Delta t A_a)c^n + M\Delta t q^n.$$

As it has an explicit part, the scheme just defined is stable under the CFL condition $u\Delta t \leq \max_i h_i$.

As this may be too severe a restriction (some of our applications require integration over a very large time interval), we use an operator splitting scheme proposed by Siegel et al. [57] (see also [28]) that is both unconditionally stable, and has a good behavior in advection dominated situations.

Splitting (Explicit advection and implicit diffusion), with sub-time steps : this scheme works by taking several small time steps of advection, controlled by CFL condition within a large time step of diffusion. Thus CFL impacts advection, and larger time steps can be taken for diffusion.

More precisely, the time step Δt will be used as the diffusion time step, it is divided into N_a time steps of advection Δt_a such that $\Delta t = N_a \Delta t_a$ where $N_a \geq 1$, the advection time step will be controlled by CFL condition. Note that taking a single advection step amount to using the implicit–explicit method seen previously. We solve equation (3.6) over the time step $[t^n, t^{n+1}]$ by first solving the advection equation $\phi\frac{\partial c}{\partial t} + \frac{\partial}{\partial x}(uc) = 0$ over N_a steps of size Δt_a each, and then solve the diffusion equation $\phi\frac{\partial c}{\partial t} + \frac{\partial}{\partial x}(-D\frac{\partial c}{\partial x}) = \phi q$ starting from the value at the end of the advection step.

Advection step. Denote the intermediate times by $t^{n,m}, m = 0, \ldots, N_a$, with $t^{n,0} = t^n$, $t^{n,N_a} = t^{n+1}$. Each interval $[t^n, t^{n+1}]$ is then divided into N_a intervals $[t^{n,m}, t^{n,m+1}], m = 0, \ldots N_a - 1$. Let $c^{n,m}$ be the approximate concentration c at time $t^{n,m}$ and $c^{n,0} = c^n$. We discretize the advection equation in time, using the explicit Euler method, we obtain

$$(3.10) \qquad Mc^{n,m+1} = (M + \Delta t_a A_a)c^{n,m}, \quad m = 0, \ldots, N_a - 1.$$

Diffusion step. The diffusion part is discretized by an implicit Euler scheme, starting from c^{n,N_a} :

$$(3.11) \qquad (M + \Delta t A_d)c^{n+1} = Mc^{n,N_a} + M\Delta t q^n.$$

For further use, we note that all three discretizations methods can be written in a similar way, as

$$(3.12) \qquad Ac^{n+1} = Bc^n + N\Delta t q^n,$$

where the matrices A and b are defined in each case as:
Fully implicit $A = M + \Delta t(A_d + A_a)$ and $B = M$,
Explicit–implicit $A = M + \Delta t A_d$ and $B = M - \Delta t A_a$,
Splitting $A = M + \Delta t A_d$ and B is defined implicitly by (3.10)

Alternatively, we may want to follow the pattern begun in section 3.1, and abstract one transport step as a mapping from c^n to c^{n+1} under the action of the source q^n. We define

$$(3.13) \qquad \begin{aligned} \psi_T : \quad & \mathbf{R}^{N_h} \times \mathbf{R}^{N_h} \to \mathbf{R}^{N_h} \\ & (c,q) \mapsto \psi_T(c,q) = A^{-1}(Bc + M\Delta t q) \end{aligned}$$

4. Methods for solving the coupled system. In this section, we discuss methods for solving the coupled transport–chemistry system. It may be difficult to compute and store the Jacobian matrix (it may be very large, and contains a block that is the inverse of the chemical solution operator, that may be difficult to compute). Consequently we advocate a Newton–Krylov approach, as this requires only the multiplication of the Jacobian matrix by a given vector, the Jacobian matrix itself does not have to be stored. By exploiting the block structure of the Jacobian matrix, the matrix–vector product can be computed in terms of the individual Jacobians.

The efficiency of the Newton–Krylov method rests on the choice of an adequate preconditioner (see [36] for several illustrations). In this paper, we show that block preconditioners for the Jacobian can be related to physics–based approximations, and also that a block Gauss–Seidel preconditioner is closely related to an elimination method at the non-linear level.

4.1. Formulation of the coupled system. We start with the coupled system that was obtained at the end of section 2. It consists of the (transformed) transport equations (2.12), together with the mass action laws (2.5). They are linked by the definition of the transformed variables T, C and $\bar{C}$ in equations (2.10) and (2.11).

Because chemistry is local, we can eliminate the individual concentrations at each point by using the "chemical solution" operator ψ_C defined in equation (3.4). This only leaves C, $\bar{C}$, T and $\bar{T}$ as unknowns that are solution of the following system

$$(4.1) \qquad \begin{aligned} \phi\partial_t C + \phi\partial_t \bar{C} + \mathcal{L}C \ &= 0, \\ \phi\partial_t \bar{T} \ &= 0, \\ T \ &= C + \bar{C}, \\ \bar{C} \ &= \psi_C(T). \end{aligned}$$

As is clear from the second equation, $\bar{T}$ is constant in time (we will see that it is simply the cationic exchange capacity in the example below). To simplify the notation, the equation for $\bar{T}$ will be omitted in the sequel, and it has been dropped from the definition of ψ_C.

The next step is to discretize the transport equations in space and time, using any of the methods that were discussed in section 3.2. For ease of notation, we denote the discrete

unknowns (vectors of size N_h) by the same letter in bold typeface as their continuous counterpart. Moreover, all quantities at time t^n will be indicated by a superscript n.

We make use of a notational device, inspired from Matlab, that was introduced in [2] to take into account the dependence of the unknowns both on the grid cell index and the chemical species index. For a block vector u_{ij}, where $i \in [1, N_c]$ represents the chemical index and $j \in [1, N_h]$ represents the spatial index, we denote by

- $u_{:,j}$ the column vector of concentrations of all chemical species in grid cell j.
- $u_{i,:}$ the row vector of concentrations of species i at all grid points;

The unknowns are thus numbered first by chemical species, then by grid points, so that all the unknowns for a single grid point are numbered contiguously. As proposed by Erhel et al. [22], we make use of the Kronecker product and the vec notation (see for instance [25]). Given $V = (V_1, V_2, \ldots, V_{N_c}) \in \mathbf{R}^{N_x \times N_c}$ (V_i denotes the ith column of V) we denote by $\text{vec}(V)$ the vector in $\mathbf{R}^{N_x N_c}$ such that

$$\text{vec}(V) = \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_{N_c} \end{pmatrix}.$$

We also extend the solution operator ψ_C (defined in (3.4)) to operate on the global vector:

$$(4.2) \quad \begin{aligned} \Psi_C : \quad & \mathbf{R}^{N_c \times N_h} \mapsto \mathbf{R}^{N_c \times N_h} \\ & \mathbf{T} \to \Psi_C(\mathbf{T}), \quad \text{with } \Psi_C(\mathbf{T})_{:,j} = \psi_C(\mathbf{T}_{:,j}), \text{ for } j = 1, \ldots, N_h \end{aligned}$$

and define the right hand side vector by $\mathbf{b}^n = B\mathbf{C}^n + M\bar{\mathbf{C}}^n$.

With these conventions, the discretized version of the coupled system (4.1) becomes

$$(4.3) \quad \begin{aligned} & A(\mathbf{C}_{i,:}^{n+1})^T + M(\bar{\mathbf{C}}_{i,:}^{n+1})^T - (\mathbf{b}_{i,:}^n)^T = 0, \quad i = 1, \ldots, N_c, \\ & \mathbf{T}_{ij}^{n+1} - \mathbf{C}_{ij}^{n+1} - \bar{\mathbf{C}}_{ij}^{n+1} = 0, \quad i = 1, \ldots, N_c, \ j = 1, \ldots, N_h, \\ & \bar{\mathbf{C}}_{:,j}^{n+1} - \Psi_C(\mathbf{T}_{:,j}^{n+1}) = 0, \quad j = 1, \ldots, N_h. \end{aligned}$$

This is a non-linear system of equations to be solved at each time step for the three unknowns $(\mathbf{C}^{n+1}, \mathbf{T}^{n+1}, \bar{\mathbf{C}}^{n+1}) \in \mathbf{R}^{3N_c N_h}$, defined by the function $f : \mathbf{R}^{3N_c N_h} \mapsto \mathbf{R}^{3N_c N_h}$, such that

$$(4.4) \quad f\begin{pmatrix} \mathbf{C} \\ \mathbf{T} \\ \bar{\mathbf{C}} \end{pmatrix} = \begin{pmatrix} (A \otimes I)\mathbf{C} + (M \otimes I)\bar{\mathbf{C}} - \mathbf{b}^n \\ \mathbf{T} - \mathbf{C} - \bar{\mathbf{C}} \\ \bar{\mathbf{C}} - \Psi_C(\mathbf{T}) \end{pmatrix} = 0$$

4.2. Solution of the coupled system by Sequential Iterative Approach. The most classical method for solving the coupled problem (4.4) is the Sequential Iterative Approach (SIA) that consists of separately solving the transport equations and the chemical equations cf. [14, 39, 52, 70].

It will be convenient to extend the solution operator for transport ψ_T (defined in (3.13)) to operate on global vectors

$$(4.5)$$
$$\begin{aligned} \Psi_T : \quad & \mathbf{R}^{N_c \times N_h} \times \mathbf{R}^{N_c \times N_h} \mapsto \mathbf{R}^{N_c \times N_h} \\ & (\mathbf{C}, \mathbf{Q}) \to \Psi_T(\mathbf{C}, \mathbf{Q}), \quad \text{with } \Psi_T(\mathbf{C}, \mathbf{Q})_{i,:} = \psi_T(\mathbf{C}_{i,:}, \mathbf{Q}_{i,:}), \text{ for } i = 1, \ldots, N_c. \end{aligned}$$

The solution operators Ψ_C and Ψ_T emphasize that both the transport and chemistry steps can be seen as black boxes. This is the basis for the approach followed in [31], where a flexible

code that allow switching the transport and chemistry components is presented. We show in the next section that this black box approach is not restricted to the SIA.

At each time step, we iterate on the transport and chemistry problem. More precisely, for each iteration k, we first solve the transport equations:

(4.6)
$$\mathbf{C}^{n+1,k+1} = \Psi_T \left(\mathbf{C}^n, M \frac{(\bar{\mathbf{C}}^{n+1,k} - \bar{\mathbf{C}}^n)}{\Delta t} \right) = -(A^{-1} \otimes I) \left((M \otimes I)\bar{\mathbf{C}}^{n+1,k} - \mathbf{b}^n) \right)$$

for $\mathbf{C}^{n+1,k+1}$ (there is one transport equation for each species). The new mobile total concentrations is added to the immobile concentrations $\bar{\mathbf{C}}^{n+1,k}$ (from the previous iteration) to obtain $\mathbf{T}^{n+1,k+1}$, and this last concentrations provides the data for the local chemical problems

(4.7)
$$\bar{\mathbf{C}}^{n+1,k+1} = \Psi_C(\mathbf{T}^{n+1,k+1}).$$

The iterations are stopped when the difference between the solutions in the iterations k and $k + 1$ is small enough, relative to a specified tolerance

(4.8)
$$\frac{\parallel \mathbf{C}^{n+1,k+1} - \mathbf{C}^{n+1,k} \parallel}{\parallel \mathbf{C}^{n+1,k+1} \parallel} + \frac{\parallel \bar{\mathbf{C}}^{n+1,k+1} - \bar{\mathbf{C}}^{n+1,k} \parallel}{\parallel \bar{\mathbf{C}}^{n+1,k+1} \parallel} < \epsilon$$

In order both to control the errors due to the splitting, and to ensure convergence, this method requires a small time step.

4.3. Solution of the non-linear system by a Newton-Krylov methods. In the geochemical literature, the SIA method is known as an operator splitting approach, but it is more properly a block Gauss–Seidel methods on the coupled system. This suggests that other, potentially more efficient, methods could be used to solve system (4.4). A natural candidate is Newton's method. Since the system is large, and also because it involves implicitly defined functions, we turn to the Newton-Krylov variant.

Recall that at each step of the "pure" form of Newton's method for solving $f(Z) = 0$, one should compute the Jacobian matrix $J = f'(Z^k)$, solve the linear system

(4.9)
$$J \, \delta Z = -f(Z^k),$$

usually by Gaussian elimination, and then set $Z^{k+1} = Z^k + \delta Z$. In practice, one should use some form of globalization procedure in order to ensure convergence from an arbitrary starting point. If a line search is used, the last step should be replaced by $Z^{k+1} = \delta Z + \lambda Z^k$, where λ is determined by the line search procedure [32].

The Newton–Krylov method (see [32, 36] and [26], to which our work is closely related) is a variant of Newton's method where the linear system (4.9) that arises at each step of Newton's method is solved by an *iterative* method (of Krylov type), for instance GMRES [51]. As the linear system is not solved exactly, the convergence theory for Newton's method does not apply directly. However, the theory has been extended to the class of Inexact Newton methods, of which the Newton–Krylov methods are representatives. The theory leads to a practical consequence, by giving specific strategies for the forcing term, that is the tolerance within which the linear system has to be solved (see [32] or the short discussion in [2]).

The main advantage of this type of method is that the full (potentially very large) Jacobian is not needed, one just needs to be able to compute the product of the Jacobian with a vector. As this Jacobian matrix vector product is a directional derivative, this leads to Jacobian free methods, where this product is approximated by finite differences. However, for the system considered here, this has several drawbacks: in addition to its inherent inaccuracy, computing

the derivative by finite difference entails solving one more chemistry problem. It was shown in [2] how this computation could be performed exactly. This is both cheaper and more accurate. Moreover, the above reference also shows how the natural block structure present in the coupled system (4.3) can be exploited to efficiently compute the residual. This is what is done in the present work, and we now give some details.

To compute the Jacobian matrix times vector product, start from equation (4.4). The Jacobian matrix of f has the following block form

$$(4.10) \qquad J_f = \begin{pmatrix} A \otimes I & 0 & M \otimes I \\ -I & I & -I \\ 0 & -J_C & I \end{pmatrix},$$

where J_C is a block diagonal matrix, whose jth block is $\psi'_C(T_{:j})$, for $j = 1, \ldots, N_h$, and the action of the Jacobian on a vector $v = \left(v_C^T, v_T^T, v_{\bar{C}}^T\right)^T$ can be computed as

$$(4.11) \qquad J_f \begin{pmatrix} v_C \\ v_T \\ v_{\bar{C}} \end{pmatrix} = \begin{pmatrix} (A \otimes I)v_C + (M \otimes I)v_{\bar{C}} \\ -v_C + v_T - v_{\bar{C}} \\ v_{\bar{C}} - J_C v_T \end{pmatrix}.$$

Of course, the Kronecker product is just a notational device, and the matrices $A \otimes I$ or $M \otimes I$ are never actually formed. Instead, we use a well know property of the Kronecker product (see [25]) :

$$(4.12) \qquad (A \otimes I)\operatorname{vec}(V) = \operatorname{vec}(VA^T),$$

and this just requires to multiply the matrix A by the concentration vector for each species.

Since the Krylov solvers can stagnate, resulting in slow convergence, possible strategies for preconditioning the linear system will be investigated in the next section.

5. Non linear and linear preconditioning. Since the Jacobian matrix is not explicitly computed (it will just be used as a theoretical device in what follows), the only available options for preconditioning the system are those that respect the block structure of the matrix. We introduce two preconditioners derived from classical block-iterative methods, and we show that these methods have strong links to the SIA method from section 4.2, and also to a non-linear elimination method, to be introduced in section 5.2.

5.1. Block preconditioning for the coupled system. Using a preconditioner means that instead of solving, at each Newton iteration, the system

$$(5.1) \qquad J_f \delta Z = -f(Z^k)$$

one solves one of the (mathematically equivalent) systems:
right preconditioning $J_f P^{-1} \delta y = -f(Z^k)$, and then $\delta Z^k = P^{-1} \delta y$,
left preconditioning $P^{-1} J_f \delta Z = -P^{-1} f(Z^k)$.
The matrix P is called a preconditioner, and it should be chosen so as to fulfill the often conflicting goals :

- P should be close to J_f so that GMRES converges faster for the preconditioned system than for the original one,
- P is "easy" to invert, so that each iteration is not much more expensive than an iteration for J_f alone.

As already mentioned, in the context of the coupled problem (4.4), the entries of the Jacobian matrix J_f are not known explicitly, so that methods that depend on the entries of the matrix (such as incomplete factorization methods) cannot be used. On the other hand, we know the block structure of J_f (see (4.10)), and we can invert the upper left diagonal block (see remark 5.1). These two facts can be exploited by resorting to block preconditioners that respect the bock structure of the Jacobian. Specifically, we investigate block methods derived from classical iterative methods, namely Jacobi and Gauss–Seidel.

Block Jacobi preconditioning The preconditioning matrix $\mathbf{P}$ for block Jacobi is:

$$\mathbf{P_{BJ}} = \begin{pmatrix} A \otimes I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix},$$

so that the action of the left-preconditioned matrix on a vector $v = (v_C^T, v_T^T, v_{\bar{C}}^T)^T$ is :

$$(5.2) \qquad \mathbf{P_{BJ}^{-1}J_f}v = \begin{pmatrix} v_C + ((A^{-1}M) \otimes I)v_{\bar{C}} \\ v_C + v_T - v_{\bar{C}} \\ -J_C v_T \end{pmatrix},$$

Block Gauss–Seidel preconditioning The preconditioning matrix $\mathbf{P}$ for block Gauss–Seidel and its inverse are :

$$\mathbf{P_{BGS}} = \begin{pmatrix} A \otimes I & 0 & 0 \\ -I & I & 0 \\ 0 & -J_C & I \end{pmatrix}, \qquad \mathbf{P_{BGS}^{-1}} = \begin{pmatrix} A^{-1} \otimes I & 0 & 0 \\ A^{-1} \otimes I & I & 0 \\ J_C(A^{-1} \otimes I) & J_C & I \end{pmatrix},$$

so that the action of the left-preconditioned matrix on a vector $v = (v_C^T, v_T^T, v_{\bar{C}}^T)^T$ is :

$$(5.3) \qquad \mathbf{P_{BGS}^{-1}J_f}v = \begin{pmatrix} v_C + & ((A^{-1}M) \otimes I)\, v_{\bar{C}} \\ v_T + & ((A^{-1}M) \otimes I)\, v_{\bar{C}} \\ v_{\bar{C}} - J_C v_{\bar{C}} + J_C \left((A^{-1}M) \otimes I \right) v_{\bar{C}}. \end{pmatrix},$$

Here again, as in (4.11), neither A^{-1} nor the Kronecker products are computed. Rather, to compute $w = \left((A^{-1}M) \otimes I \right) v_{\bar{C}}$, for a given vector $v_{\bar{C}} \in \mathbf{R}^{N_c N_h}$, one defines $W \in \mathbf{R}^{N_x \times N_c}$ as the solution of

$$(5.4) \qquad\qquad\qquad\qquad AW = MV_{\bar{C}}^T$$

where $v_{\bar{C}} = \mathrm{vec}(V_{\bar{C}})$, then let $w = \mathrm{vec}(W)$.

This means that the action of the preconditioner can be computed by solving a transport step for each chemical species, and a multiplication by the Jacobian of the chemical operator (which in turns requires solving a linearized chemical problem for each grid cell). These are also the building blocks for the SIA formulation, and this shows that the fully coupled approach can be implemented at roughly the same cost per iteration as the SIA approach.

REMARK 5.1. *In both cases, notice that the preconditioning step involves the inverse of the transport block. In our 1D case, this is not a difficulty, as this block can be easily inverted. When we move to 2D or 3D problems, this step would become more problematic. However, the preconditioners could then be defined by replacing the matrix A in the definition of J_f by the application of a (spectrally equivalent) matrix, such as several iterations of a multigrid*

solver. A multigrid method such as that proposed recently in [59] would be particularly appropriate.

REMARK 5.2. *In both cases (exact or inexact transport solver), one should expect mesh independent convergence, as the operator being solved by GMRES becomes bounded independently of the mesh. This expectation will be confirmed by the numerical results in section 6.5.*

5.2. Elimination of unknown as non-linear preconditioning. In this section, we consider alternative solution strategies, based on eliminating some of the unknowns from the system (4.4). We group these strategies under the heading of "non-linear preconditioning" because the elimination is done before the linearization, but it must be noted that we take advantage of the fact that the first (block) equation of (4.4) is linear and can be solved species by species. In this context, this strategy should be reminiscent of the "non-linear preconditioning" as introduced by Cai and Keyes [9], where block of unknowns are eliminated locally.

We first eliminate $\mathbf{T}$ from the original system (4.4), leading to a system with only $\mathbf{C}$ and $\bar{\mathbf{C}}$ as unknowns

$$g\begin{pmatrix}\mathbf{C}\\\bar{\mathbf{C}}\end{pmatrix} = 0$$

with

$$(5.5) \qquad g\begin{pmatrix}\mathbf{C}\\\bar{\mathbf{C}}\end{pmatrix} = \begin{pmatrix}(A\otimes I)\mathbf{C} + (M\otimes I)\bar{\mathbf{C}} - \mathbf{b}^n\\\bar{\mathbf{C}} - \Psi_C(\mathbf{C} + \bar{\mathbf{C}})\end{pmatrix}.$$

The Jacobian of the new system is

$$J_g = \begin{pmatrix}A\otimes I & M\otimes I\\-J_C & I - J_C\end{pmatrix}.$$

and the Jacobian by vector product is :

$$(5.6) \qquad J_g\begin{pmatrix}v_C\\v_{\bar{C}}\end{pmatrix} = \begin{pmatrix}(A\otimes I)v_C + (M\otimes I)v_{\bar{C}}\\-J_C(v_C + v_{\bar{C}}) + v_{\bar{C}}\end{pmatrix},$$

One can even go one step further, by eliminating both the unknowns $\mathbf{T}$ and $\mathbf{C}$, to obtain a system with $\bar{\mathbf{C}}$ as the single unknown:

$$(5.7) \qquad h(\bar{\mathbf{C}}) = \bar{\mathbf{C}} - \Psi_C\left(\Psi_T\left(\mathbf{C}^n, \frac{\bar{\mathbf{C}} - \bar{\mathbf{C}}^n}{\Delta t}\right) + \bar{\mathbf{C}}\right) = 0.$$

This equation is presented in fixed point form, and solving it by fixed point method recovers the SIA method described in Section 4.2. But equation (5.7) can also be solved by Newton's method. We presented a Jacobian-free Newton-Krylov method in [2]. Its main advantage is to require that one be able to evaluate the nonlinear residual function h, and the Jacobian times vector product is approximated by finite difference. Since it involves solving an additional chemical problem, this step is very expensive. In this work, we use the exact Jacobian of h, which requires that we "open the black box", to look in more detail at the structure of the Jacobian. To do this, we rewrite the equation above by making use of the precise definition of the transport operator Ψ_T:

$$(5.8) \qquad h(\bar{\mathbf{C}}) = \bar{\mathbf{C}} - \Psi_C\left((A^{-1}\otimes I)\left(\mathbf{b}^n - (M\otimes I)\bar{\mathbf{C}}\right) + \bar{\mathbf{C}}\right) = 0.$$

If we denote by $J_T = I - \left((A^{-1}M) \otimes I\right)$ the Jacobian matrix of Ψ_T, we easily see that the Jacobian by vector product for h is:

$$(5.9) \qquad J_h v = v - J_C J_T v = v - J_C v - J_C \left((A^{-1}M) \otimes I\right) v.$$

Both the alternative formulation and its Jacobian involve the resolution of transport at each Newton step. One again expects that convergence becomes independent of the mesh size [34]. This will be confirmed in Section 6.

5.3. Links between block preconditioning and elimination. The main advantage of the SIA approach is that it allows the reuse of existing software modules for solving transport and chemistry. This is an important practical issue, as these modules may have been developed independently, even by different groups. This advantage is offset by the possibly slower convergence when compared to the Direct Substitution Approach (DSA, see for example [26]). The "h" method proposed in section 5.2 (keeping only $\bar{C}$ as unknown) aims at combining the advantages of SIA and DSA:

- it allows to keep transport and chemistry as separate modules. Equation (5.8) show that the evaluation of the residual when solving the system with h requires solving a transport step for each species, and then solving a local chemical equilibrium system at each grid cell.
- it is a global method, in the sense that it solves both chemistry and transport in a single iteration. This method is in the same spirit as the one proposed by Knabner and his group [27, 37, 38]. The reduction method proposed in these papers is more general than the formulation in section 4.1, but their "resolution function" is identical to the chemical solution operator Ψ_C.

The cost for evaluating the non-linear residual is the same as one iteration of SIA. One then has to balance the added cost of solving a linear system at each iteration with the benefit of taking a smaller number of iterations.

It is interesting to note that both the SIA method and the "h" function method from section 5.2 can be interpreted in terms of the block preconditioners introduced in section 5.1.

Indeed, the elimination method can be interpreted as a (linear) change of variables, given by

$$(5.10) \qquad \begin{cases} \tilde{\mathbf{C}} = \mathbf{C} & + & \left((A^{-1}M) \otimes I\right) \bar{\mathbf{C}} \\ \tilde{\mathbf{T}} = & \mathbf{T} - \left((I - A^{-1}M) \otimes I\right) \bar{\mathbf{C}} \,, \\ \tilde{\bar{C}} = & & \bar{\mathbf{C}} \end{cases}$$

whose matrix

$$B = \begin{pmatrix} I & 0 & (A^{-1}M) \otimes I \\ 0 & I & -(I - A^{-1}M) \otimes I \\ 0 & 0 & I \end{pmatrix}.$$

is block triangular, so that the transformation is easily inverted. The transformed system

$$\tilde{f} \begin{pmatrix} \tilde{\mathbf{C}} \\ \tilde{\mathbf{T}} \\ \tilde{\bar{C}} \end{pmatrix} = f \begin{pmatrix} \mathbf{C} \\ \mathbf{T} \\ \bar{\mathbf{C}} \end{pmatrix}$$

takes the simple form

$$(5.11) \qquad \tilde{f}\begin{pmatrix} \tilde{\mathbf{C}} \\ \tilde{\mathbf{T}} \\ \tilde{\bar{C}} \end{pmatrix} = \begin{pmatrix} (A \otimes I)\tilde{\mathbf{C}} - \mathbf{b} \\ -\tilde{\mathbf{C}} + \tilde{\mathbf{T}} \\ \tilde{\bar{C}} - \Psi_C\left(\tilde{\mathbf{T}} + \left((I - A^{-1}M) \otimes I\right)\tilde{\bar{C}}\right) \end{pmatrix} = 0.$$

and the last equation is obviously identical to (5.8), after having eliminated the first two unknowns.

Because the transformation used is linear, and because Newton's method is invariant under a linear transformation, the iterates between the original and the transformed system will be related by the same transformation (provided the initial guesses are related similarly). Note that this will *not* necessarily be true for an inexact Newton's method, such as the Newton-Krylov method used in this work. However, Deuflhard points out [19, sec. 2.2.4] that because the Newton residual is invariant under affine transformations, GMRES is the natural choice for solving the linear system arising at each Newton iteration. Moreover, left preconditioning can be used with GMRES provided the preconditioning matrix

- is kept constant throughout the Newton iterations,
- and is incorporated in the convergence monitoring criteria.

Note that the Jacobian of the transformed system is the block triangular matrix

$$(5.12) \qquad J_{\tilde{f}} = \begin{pmatrix} A \otimes I & 0 & 0 \\ -I & I & 0 \\ 0 & -J_c & S \end{pmatrix}$$

where $S = I - J_c\left((I - A^{-1}M) \otimes I\right)$ is actually the Schur complement of J_f with respect to its first two variables.

This confirms the close link between the non-linear elimination method from section 5.2 and block Gauss–Seidel preconditioning in (5.3): as noted above, the Jacobian of h is exactly the Schur complement of the Jacobian of f. In both cases, one needs to compute $(J_C((A^{-1}M) \otimes I))v$. Of course, we are once again taking advantage of the fact that one of the operators is linear !

An interesting consequence of the change of variables is that, because $J_{\tilde{f}}$ and the change of variable matrix B are both block triangular, differentiating the identity $f = \tilde{f} \circ B$ leads to a block triangular factorization of J_f,

$$(5.13) \qquad J_f = J_{\tilde{f}}B^{-1} = \begin{pmatrix} A \otimes I & 0 & 0 \\ -I & I & 0 \\ 0 & -J_c & S \end{pmatrix} \begin{pmatrix} I & 0 & (A^{-1}M) \otimes I \\ 0 & I & -(I - A^{-1}M) \otimes I \\ 0 & 0 & I \end{pmatrix}.$$

This factorization could be used as a basis for constructing efficient preconditioners for J_f, much as in the spirit of [29], [30] (see also [48]). The matrix B is upper triangular with unit diagonal, so that all its eigenvalues are equal to 1. As noted in [65], under these conditions, Krylov subspace methods for the preconditioned system $J_{\tilde{f}}^{-1}J_f$ converge in 1 iteration, giving $J_{\tilde{f}}$ as a perfect preconditioner.

Because it contains S as its $(3,3)$ block, $J_{\tilde{f}}$ cannot be formed, let alone inverted. A first solution is to replace S by an approximation $\tilde{S}$ that is easier to invert. The simplest choice is to take $\tilde{S} = I$, which gives the block Gauss–Seidel preconditioner from section 5.1. But actually, systems with S can be solved by an iterative method, as the matrix vector product Sv can be computed by proceeding as for the Gauss–Seidel preconditioner at the end of section 5.1 (see equation (5.4)).

The numerical results in section section 6.5 (cf. figure 6.10) will confirm that the performance of SIA and block Jacobi on the one hand and the elimination method and block Gauss–Seidel on the other hand are very similar.

6. Numerical results.

6.1. MoMaS Benchmark : 1D easy advective case. The MoMaS Benchmark has been designed to compare numerical methods for reactive transport model in 1D and 2D. Different methods for coupling have been used to solve this benchmark. The definition has been published in [12] and the results of participants are compared in the synthesis article [11].

The geometry of the test case is show in Figure 6.1. For the 1D test case, the domain is heterogeneous and composed of two porous media A and B. Medium A is highly permeable with low porosity and low reactivity in comparison with medium B. Their physical properties are given in Table 6.1. The pore velocity is constant over the domain and is equal to $\phi u = 5.5 \, 10^{-3} \mathrm{LT}^{-1}$.
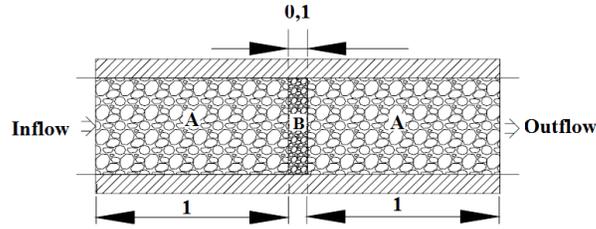
FIG. 6.1. *Geometry of the domain*

	Medium A	Medium B
Porosity ϕ $(-)$	0.25	0.5
Total immobile concentration T_S	1	10

TABLE 6.1
Physical properties of the materials

The chemical reactions are summarized in Table 6.2. The 7 reactions involve 9 mobile species (in the aqueous phase) and 3 immobile species (in the solid phase).

The characteristic feature of this chemical system is that it contains large stoichiometric coefficients that range from -4 to 4 and a large variation of equilibrium constants between 10^{-12} and 10^{35}.

Boundary and initial conditions are presented in Table 6.3. The simulation involves an injection period followed by a leaching period, so that the system is returned to its initial state. Note that the total for the immobile species is given in Table 6.1.

6.2. Sample results. The results obtained in [11] indicate that one needs to refine the mesh around medium B if one is to obtain accurate results. For all test cases, we use a mesh such that $h_A = 4h_B$.

Figure 6.2 shows profiles of the concentrations of several species. The left and middle images show concentrations of (a part of) mobile species $C1$ and immobile species S at an early time $t = 10$, whereas the right image shows concentration of aqueous species C_2 on a smaller interval during the leaching period, at $t = 5010$. The middle image highlights the effect of the heterogeneity at $x = 1$.

	X_1	X_2	X_3	X_4	S	K
C_1	0	-1	0	0	0	$\mathbf{10^{-12}}$
C_2	0	1	1	0	0	1
C_3	0	-1	0	1	0	1
C_4	0	**-4**	1	3	0	10^{-1}
C_5	0	4	3	1	0	$\mathbf{10^{+35}}$
CS_1	0	3	1	0	1	10^{+6}
CS_2	0	-3	0	1	2	10^{-1}

TABLE 6.2

Morel tableau for Chemical equilibrium

	T_1	T_2	T_3	T_4
Total				
Conc.	0	-2	0	2
Injection	0.3	0.3	0.3	0
t∈[0,5000]				
Leaching	0	-2	0	2
t∈[5000,..]				

TABLE 6.3

Boundary conditions
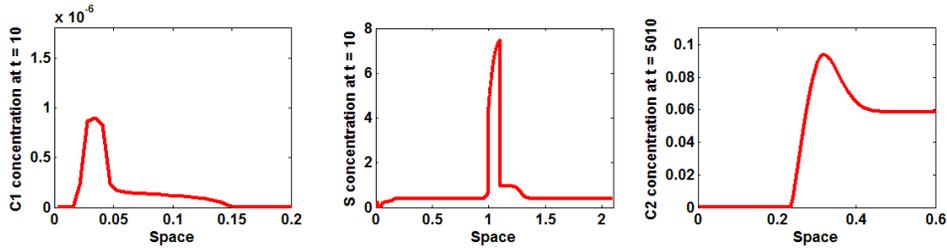
FIG. 6.2. *Concentrations profiles. Left figure: C_1 at $t = 10$, middle figure S at $t = 10$, right figure C_2 at $t = 5010$.*

Figure 6.3 shows elution curves, that is evolution of the concentrations at the end of the domain ($x = 2.1$) as a function of time, for a mesh with 384 elements. Notice that the evolution of $C5$ follows a fairly complex pattern. It turns out that the accuracy for both species $X3$ and $C5$ is quite sensitive to the mesh size used. We come back to this point in detail in the next subsection.
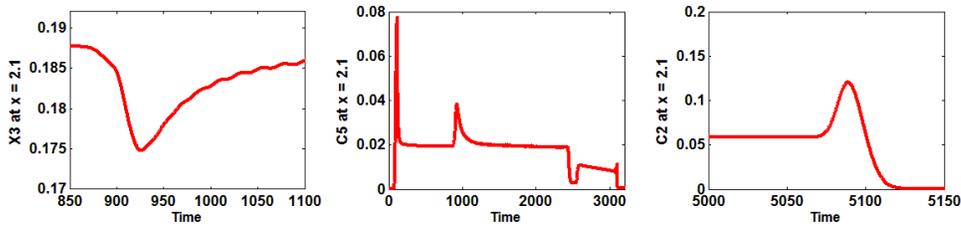
FIG. 6.3. *Elution curves (concentrations at right end of the domain as a function of time). Left image: species $X3$, middle image: species $C5$, right image: species $C2$*

The results shown on these figures are in good agreement with those showed by various groups in the comparison paper[11].

6.3. Influence of spatial and temporal discretizations. As mentioned above, some of the species show a very sensitive dependence on the mesh and time step used. It has proven necessary to use very fine meshes, as well as small time steps, in order to resolve these species accurately.

6.3.1. Influence of spatial discretization. The evolution of species $X3$ and $C5$, as a function of time, exhibits unphysical oscillations. The origin of these oscillations has been

explained in [39], and is due to the interplay between the very stiff reactions and the spatial discretizations. They should decrease as the mesh is refined.

Figures 6.4 shows that this is indeed what happens as we increase the number of discretization points. A mesh with 384 points gives qualitatively correct results, but we have also made use of a finer meshes in order to obtain more accurate results.
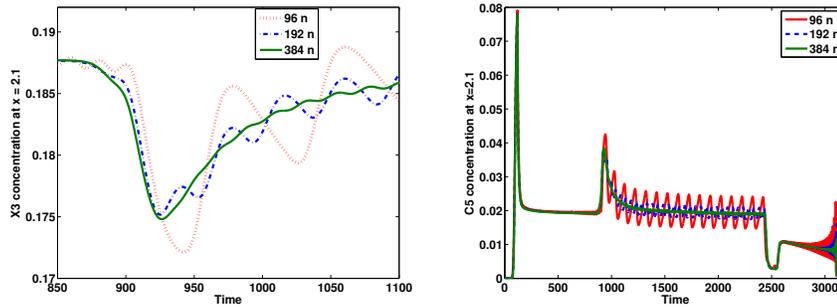

FIG. 6.4. *Elution curves of X_3 (left) and C_5 (right) concentration, for various mesh resolutions*

The effect of the mesh size on the accuracy of the results is also shown by looking in detail at two specific species: mobile species C_1 and immobile species S, both at time $t = 10$. They both exhibit a sharp peak, and we focus on the accuracy with which the location and amplitude of the peak can be determined. These elements were part of the comparison criteria for the benchmark, and were examined in detail in [11].

We also discuss the influence of the discretization scheme, by comparing the three schemes introduced in section 3.2 from the point of view of accuracy. Their relative efficiencies are compared in section 6.4.

Figure 6.5 compares the concentrations of species C_1 on the interval $[0, 0.2]$, at time $t = 10$, as computed by the three discretization schemes on increasingly refined meshes. As soon as the mesh has sufficiently many points to successfully resolve the solution, all three schemes give identical solutions. On the other hand, one needs at least 384 points (and preferably 768) to obtain a satisfactory solution.
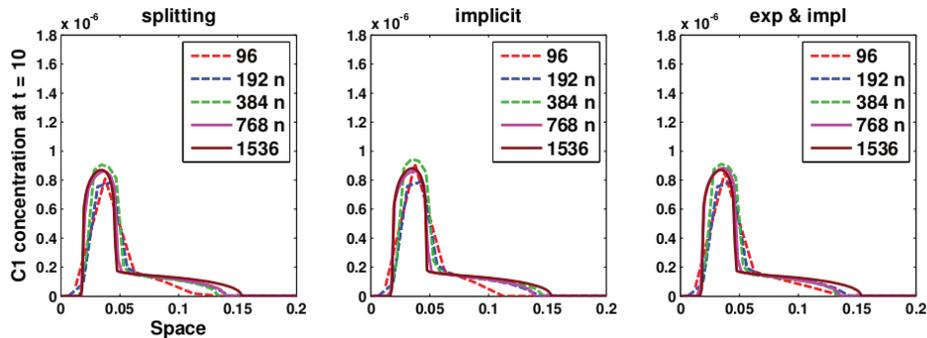

FIG. 6.5. *Concentration of species C_1 at $t = 10$ for the different discretization schemes and various mesh resolutions*

Figure 6.6 compares the concentration of species S (a sorbed species) on the interval

bracketing the location of the peak $[0, 0.15]$, at time $t = 10$, as computed by the three discretization schemes on increasingly refined meshes. Here also, the three schemes give identical solutions when the mesh is fine enough. This time, one needs at least 768 mesh points to obtain a converged solution. One should compare Figure 6 in [11], and also refer to Table 4 there, where the location and amplitude of the peak are tabulated for all the methods used in the benchmark. We have obtained values of $x = 0.0175$ for the location of the peak, and $S = 0.985$ for its height. These values are in the range reported by the other teams, but are different form the "mean" values as reported in [11]. They are however very close to the "reference" values found in [10].
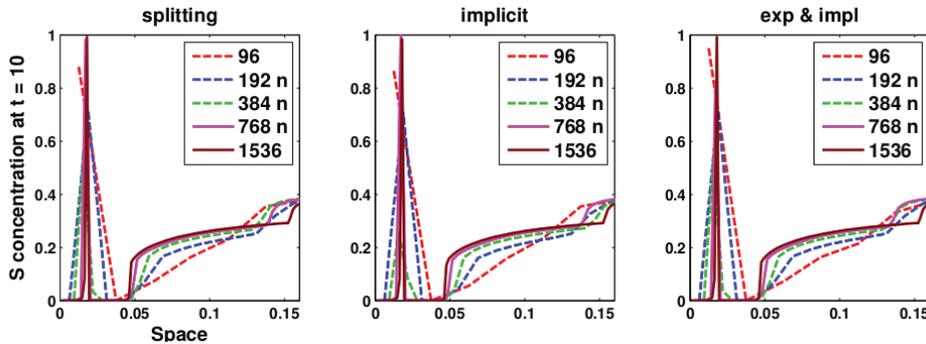
FIG. 6.6. *Concentration of species S at $t = 10$ for the different discretization schemes and various mesh resolutions*

6.3.2. Influence of temporal discretization. We now discuss the influence of the time step, or more precisely the value of the CFL coefficient (given by $CFL = u\phi\Delta t/h_{\min}$ on the accuracy.

The three schemes used for time discretization have different behavior with respect to the choice of time step:

The splitting method allows the use of different time steps (and different numerical methods) for the advection and the diffusion step. The advection time step is restricted by the CFL condition (it is an explicit sub-step), whereas the time step for diffusion is not restricted by stability. We have used three different choices: chose the same time step for advection and diffusion (respecting the CFL condition), once with a CFL condition of 1, and once with a CFL condition of 0.1, and choose a diffusion time step 3 times larger than the advection time step (the latter chosen by the CFL condition).

The fully implicit method here there is only one time step, and no stability restriction. We have compared three time steps, corresponding to CFL coefficients of 0.1, 1 and 3 respectively.

The explicit–implicit method this method also imposes a single time step, and in addition it is subject to the stability condition of the explicit method, so the time step is restricted to $CFL \leq 1$. We compare 2 time steps, corresponding to CFL coefficients of 0.1 and 1 respectively.

Figure 6.7 and 6.8 compare the concentration of species S as computed by the three methods, for the time step sizes chosen as explained above, for 2 different mesh resolutions.

For all 3 methods, it is necessary to use a small time step ($CFL = 0.1$) if one wishes to obtain accurate results. Indeed, for larger values of the time step, the location of the

peak was correctly determined, but its amplitude was underestimated. For the finest mesh resolution (1536 points), all 3 methods gave identical results, provided a small time step was used. However, for the coarser mesh resolution (768 points), both the fully implicit and the splitting methods gave comparable results, with the former being slightly less sensitive to the choice of time step.

The explicit-implicit method was the most sensitive to the mesh size, as it needs a finer mesh to obtain the correct amplitude for the peak.
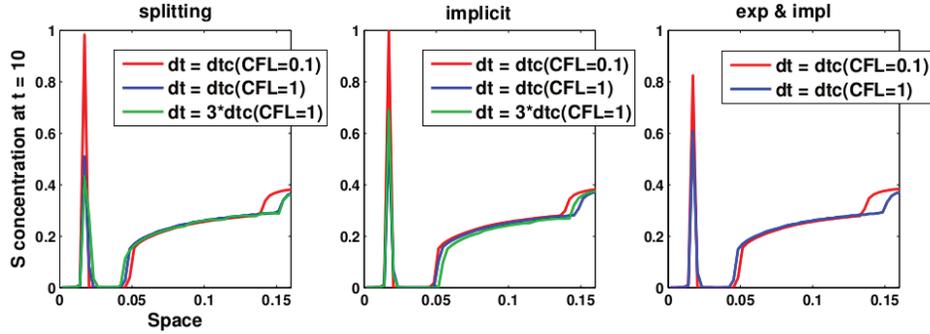
FIG. 6.7. *the effect of CFL condition on the concentrations in S (768 nodes)*
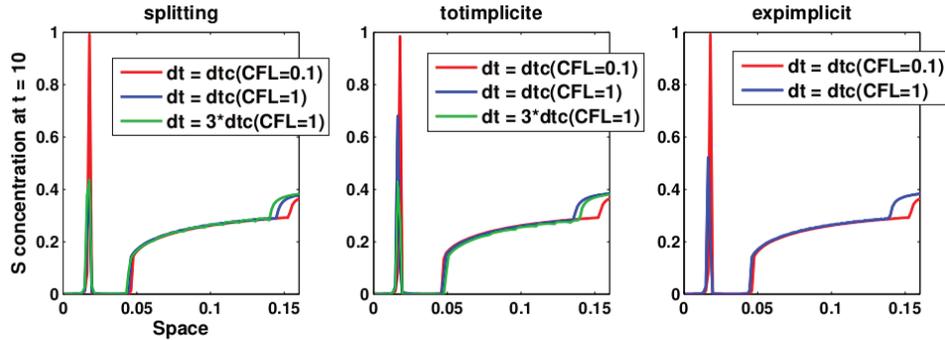
FIG. 6.8. *the effect of CFL condition on the concentrations in S (1536 nodes)*

6.4. CPU time. We now compare the relative efficiency of the three time discretization methods. One should keep in mind that the problem under study is a 1D model, so that solving a linear system is a not very time consuming. The conclusions reached below would have to be re-examined for a 2D (and even more for a 3D !) model.

We have also compared the effect of using the fully implicit method with a variable time step (albeit not an adaptive choice, a sequence of pre-computed time steps is used for different phases of the solution evolution).

Figure 6.9 compares the times required by the different methods. As expected, using a variable time step results in large savings (while maintaining the accuracy). Among the 3 schemes with fixed time step, the explicit–implicit method is the most expensive, with the fully implicit and the splitting methods leading to comparable costs.

6.5. Influence of preconditioning strategy. In this section, we compare the various preconditioning strategies discussed in section 5. Our main criteria will be the number of
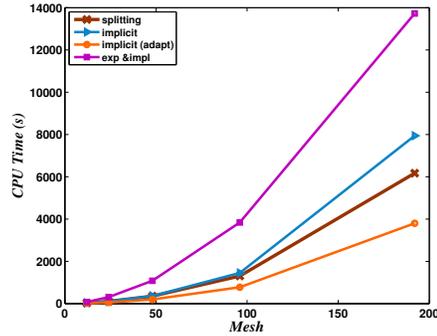
FIG. 6.9. *CPU time required by the time discretization methods*

linear and non-linear iterations. We have not tried to optimize the inexact Newton strategy, but have just relied on the default choices as provided in the Newton–Krylov code (we use the `nsoli` code from the book by C. T. Kelley [33].

Figure 6.10 shows how these numbers change as the mesh is refined. The various linear preconditioning strategies (applied to the coupled system) are compared with the elimination, or nonlinear preconditioning, strategy. As predicted, the nonlinear elimination strategy has the smallest number both for non-linear and for linear iterations. It also shows a behavior that is independent of the mesh size.

The unpreconditioned method is unsurprisingly non scalable, at least as far as the linear iterations are concerned. The number of non-linear iterations grows only weakly with the number of mesh points. The same is true for Jacobi preconditioning.

Gauss–Seidel preconditioning, on the other hand, show only a modest increase in the number of linear iterations, and a behavior for the non-linear iterations that is in between that of the unpreconditioned and the elimination strategies.

As explained in section 5, a limitation of the elimination strategy is that it requires an exact solution of the transport step. In this case, the Gauss–Seidel preconditioner might prove useful: replacing the transport solve step by an approximation, such as several iterations of a multi-grid solver, should lead to a more efficient solution method, with similar convergence behavior. We plan to explore this strategy in a forthcoming work.
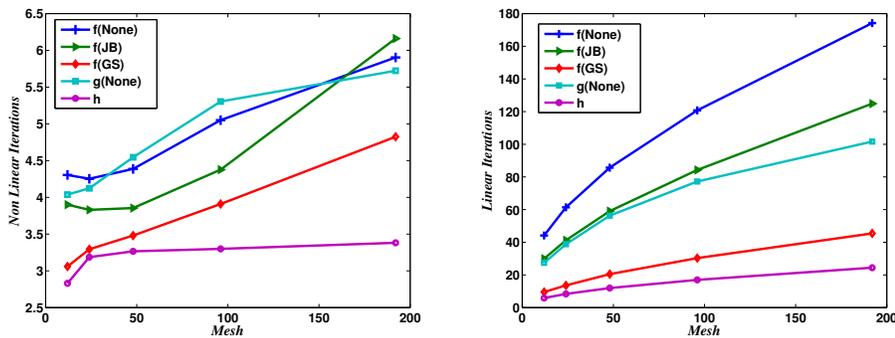

FIG. 6.10. *Non-linear (left) and linear (right) iterations for different preconditioning strategies*

Last, figure 6.11 shows the time required by the various methods. Since the cost of the

methods is comparable, the ordering is the same as that in the previous figure. It confirms the good efficiency of the elimination strategy, with Gauss–Seidel preconditioning as a distant second.
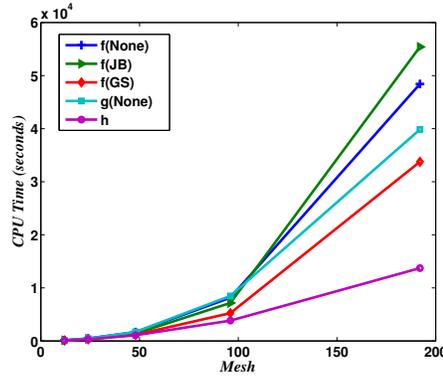

FIG. 6.11. *CPU time as a function of the mesh size for the various solution methods*

We can try and summarize the relative performances of the various methods as follows:

- the original f formulation is not numerically scalable, neither at the non-linear level, nor at the linear level;
- the block Jacobi preconditioner applied to f does not bring any improvement;
- the block Gauss–Seidel preconditioner improves the linear performance, but does have a significant non-linear effect (nor was it expected);
- the g formulation improves the linear performance, but not as much as Gauss–Seidel preconditioning;
- the h formulation, after elimination of the C unknowns is the only method that gives a convergence independent of the mesh size.

Of course, these conclusions are more or less natural: methods f and g keep the ill-conditioning from the second order operator (except for g at the linear level). The elimination method, leads to a bounded operator on L^2 (at least formally), and is expected to give mesh independent convergence.

This good performance of the elimination method, at least on the linear level, can be confirmed by looking at the *field of values* of the matrix J_h. The field of values of a matrix is the subset of the complex plane defined by

$$W(J) = \left\{ \frac{x^H J x}{x^H x}, x \in \mathbf{C}, x \neq 0 \right\}.$$

It includes the eigenvalues and is a convex set. For a non-symmetric matrix, the convergence of GMRES is better described by the field of values than by the eigenvalues (see [6] or [35]). We conjecture that the field of values of the Schur complement S can be bounded away from 0, independently of N_x. Though we have no proof at the moment, we have the following numerical confirmation on figure 6.12, which shows the field of values of the Jacobian matrix J_h for 2 different mesh sizes. One can see that the convex hull of the field of values is indeed approximately independent of the mesh size. The figure was obtained thanks to the Eigtool software [66, 67].
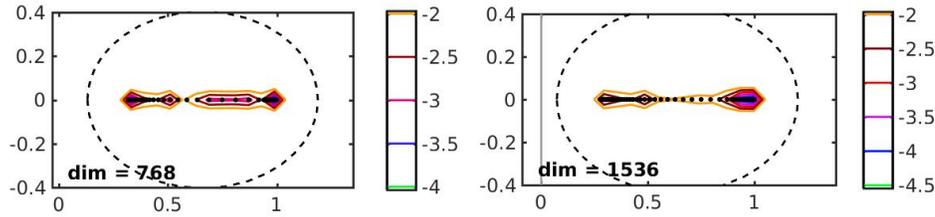
FIG. 6.12. *Field of values for matrix S, with $N_x = 768$ and $N_x = 1536$*

7. Conclusion and perspectives. In this work, several methods for improving the efficiency of a global approach for coupling transport and chemistry based on a Newton-Krylov method were studied.

An alternative formulation and block preconditioners for linear system were used to accelerate the convergence of the Krylov method and to reduce CPU time. The results show that the alternative formulation require less CPU time than other preconditioners, and the number of linear and non linear iterations becomes almost independent of the mesh.

The reactive transport benchmark 1D problem proposed by GNR MoMaS was used to demonstrate the efficiency of the method.

Natural extensions of this work to multidimensional situations are under way, as well as extensions to handle kinetic reactions.

REFERENCES

[1] E. AHUSBORDE, M. KERN, AND V. VOSTRIKOV, *Numerical simulation of two-phase multicomponent flow with reactive transport in porous media: application to geological sequestration of CO2*, ESAIM: Proc., 50 (2015), pp. 21–39.

[2] L. AMIR AND M. KERN, *A global method for coupling transport with chemistry in heterogeneous porous media*, Computat. Geosci., 14 (2010), pp. 465–481. 10.1007/s10596-009-9162-x.

[3] C. A. J. APPELO AND D. POSTMA, *Geochemistry, Groundwater and Pollution*, CRC Press, 2nd ed., 2005.

[4] P. AUDIGANE, I. GAUS, I. CZERNICHOWSKI-LAURIOL, K. PRUESS, AND T. XU, *Two-Dimensional reactive transport modeling of CO2 injection in a saline aquifer at the Sleipner site, North sea*, American journal of science, 307 (2007), pp. 974–1008.

[5] J. BEAR AND A. H.-D. CHENG, *Modeling Groundwater Flow and Contaminant Transport*, vol. 23 of Theory and Applications of Transport in Porous Media, Springer, 2010.

[6] M. BENZI AND M. A. OLSHANSKII, *Field-of-values convergence analysis of augmented Lagrangian preconditioners for the linearized Navier–Stokes problem*, SIAM J. Numer. Anal., 49 (2011), pp. 770–788.

[7] C. M. BETHKE, *Geochemical and Biogeochemical Reaction Modeling*, Cambrdige University Press, 2nd ed., 2008.

[8] Å. BJÖRCK, *Numerical methods for least squares problems*, Society for Industrial and Applied Mathematics, Philadelphie, 1996.

[9] X.-C. CAI AND D. E. KEYES, *Nonlinearly preconditioned inexact newton algorithms*, SIAM J. Sci. Comput., 24 (2002), pp. 183–200.

[10] J. CARRAYROU, *Looking for some reference solutions for the reactive transport benchmark of MoMaS with SPECY*, Computat. Geosci., 14 (2010), pp. 393–403.

[11] J. CARRAYROU, J. HOFFMANN, P. KNABNER, S. KRÄUTLE, C. DE DIEULEVEULT, J. ERHEL, J. VAN DER LEE, V. LAGNEAU, K. U. MAYER, AND K. T. B. MACQUARRIE, *Comparison of numerical methods for simulating strongly nonlinear and heterogeneous reactive transport problems-the MoMaS benchmark case*, Computat. Geosci., 14 (2010), pp. 483–502.

[12] J. CARRAYROU, M. KERN, AND P. KNABNER, *Reactive transport benchmark of MoMaS*, Computat. Geosci., 14 (2010), pp. 385–392. 10.1007/s10596-009-9157-7.

[13] J. CARRAYROU, R. MOSÉ, AND P. BEHRA, *New efficient algorithm for solving thermodynamic chemistry*, Aiche J., 48 (2002), pp. 894–904.

[14] J. CARRAYROU, R. MOSÉ, AND P. BEHRA, *Operator-splitting procedures for reactive transport and comparison of mass balance errors*, J. Contam. Hydrol., 68 (2004), pp. 239–268.

[15] A. CHILAKAPATI, T. GINN, AND J. SZECSODY, *An analysis of complex reaction networks in groundwater modeling*, Water Resour. Res., 34 (1998), pp. 1767–1780.

[16] C. DE DIEULEVEULT AND J. ERHEL, *A global approach to reactive transport: application to the MoMas benchmark*, Computat. Geosci., 14 (2010), pp. 451–464.

[17] C. DE DIEULEVEULT, J. ERHEL, AND M. KERN, *A global strategy for solving reactive transport equations*, J. Comput. Phys., 228 (2009), pp. 6395–6410.

[18] L. DE WINDT, D. PELLEGRINI, AND J. VAN DER LEE, *Coupled modeling of cement/claystone interactions and radionuclides migration*, J. Contam. Hydrol., 68 (2004), pp. 165–182.

[19] P. DEUFLHARD, *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*, vol. 35 of Springer Series in Computational Mathematics, Springer, 2011.

[20] J. ERHEL AND T. MIGOT, *Characterizations of Solutions in Geochemistry: Existence, Uniqueness and Precipitation Diagram.* working paper or preprint, Sept. 2017.

[21] J. ERHEL AND S. SABIT, *Analysis of a global reactive transport model and results for the momas benchmark*, Math. Comput. Simul., 137 (2017), pp. 286–298.

[22] J. ERHEL, S. SABIT, AND C. DE DIEULEVEULT, *Solving partial differential algebraic equations and reactive transport models*, in Developments in Parallel, Distributed, Grid and Cloud Computing for Engineering, B. H. V. Topping and P. Ivànyi, eds., Computational Science, Engineering and Technology Series, Saxe Coburg Publications, 2013, pp. 151–169.

[23] R. EYMARD, T. GALLOUËT, AND R. HERBIN, *Finite volume methods*, in Handbook of Numerical Analysis, P. G. Ciarlet and J. L. Lions, eds., vol. VII, North–Holland, 2000, pp. 713–1020.

[24] Y. FAN, L. J. DURLOFSKY, AND H. A. TCHELEPI, *A fully-coupled flow-reactive-transport formulation based on element conservation, with application to CO 2 storage simulations*, Adv. Water Resour., 42 (2012), pp. 47–61.

[25] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, 4th ed., 2013.

[26] G. E. HAMMOND, A. VALOCCHI, AND P. LICHTNER, *Application of Jacobian-free Newton–Krylov with physics-based preconditioning to biogeochemical transport*, Adv. Water Resour., 28 (2005), pp. 359–376.

[27] J. HOFFMANN, S. KRÄUTLE, AND P. KNABNER, *A parallel global-implicit 2-D solver for reactive transport problems in porous media based on a reduction scheme and its application to the momas benchmark problem*, Comput Geosci, 14 (2009), pp. 421–433.

[28] H. HOTEIT, P. ACKERER, AND R. MOSÉ, *Nuclear waste disposal simulations: Couplex test cases: Simulation of transport around a nuclear waste disposal site: The COUPLEX test cases (editors: Alain Bourgeat and Michel Kern)*, Computat. Geosci., 8 (2004), pp. 99–124.

[29] V. E. HOWLE, R. C. KIRBY, AND G. DILLON, *Block preconditioners for coupled physics problems*, SIAM J. Sci. Comput., 35 (2013), pp. S368–S385.

[30] I. C. F. IPSEN, *A note on preconditioning nonsymmetric matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 1050–1051.

[31] D. JARA HEREDIA, J.-R. DE DREUZY, AND B. COCHEPIN, *TReacLab: an object oriented implementation of non intrusive operator splitting methods to couple independent transport and geochemical software*, Comput. Geosci., (2017). to appear.

[32] C. T. KELLEY, *Iterative methods for linear and nonlinear equations*, vol. 16 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1995. With separately available software.

[33] ———, *Solving Nonlinear Equations with Newton's Method*, vol. 1 of Fundamentals of Algorithms, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2003. With separately available software.

[34] M. KERN AND A. TAAKILI, *Linear and nonlinear preconditioning for reactive transport*, in XVIII Conference on Computational Methods in Water Resources CMWR XVIII, J. Carrera, P. Binning, and G. F. Pinder, eds., 2010. http://congress.cimne.upc.es/cmwr2010/Proceedings/docs/p30.pdf.

[35] A. KLAWONN AND G. STARKE, *Block triangular preconditioners for nonsymmetric saddle point problems: field-of-values analysis*, Numer. Math., 81 (1999), pp. 577–594.

[36] D. A. KNOLL AND D. E. KEYES, *Jacobian-free Newton-Krylov methods: a survey of approaches and applications*, J. Comput. Phys., 193 (2004), pp. 357–397.

[37] S. KRÄUTLE AND P. KNABNER, *A new numerical reduction scheme for fully coupled multicomponent transport-reaction problems in porous media*, Water Resour. Res., 41 (2005).

[38] ———, *A reduction scheme for coupled multicomponent transport-reaction problems in porous media: Generalization to problems with heterogeneous equilibrium reactions*, Water Resour. Res., 43 (2007).

[39] V. LAGNEAU AND J. VAN DER LEE, *HYTEC results of the MoMas reactive transport benchmark*, Computat. Geosci., 14 (2010), pp. 435–449.

[40] H. MACHAT AND J. CARRAYROU, *Comparison of linear solvers for equilibrium geochemistry computations*, Computational Geosciences, 21 (2017), pp. 131–150.

[41] M. MARINONI, J. CARRAYROU, Y. LUCAS, AND P. ACKERER, *Thermodynamic equilibrium solutions*

through a modified newton raphson method, AIChE Journal, 63 (2017), pp. 1246–1262.

[42] N. C. MARTY, C. TOURNASSAT, A. BURNOL, E. GIFFAUT, AND E. C. GAUCHER, *Influence of reaction kinetics and mesh refinement on the numerical modelling of concrete/clay interactions*, Journal of Hydrology, 364 (2009), pp. 58–72.

[43] N. C. M. MARTY, O. BILDSTEIN, P. BLANC, F. CLARET, B. COCHEPIN, E. C. GAUCHER, D. JACQUES, J.-E. LARTIGUE, S. LIU, K. U. MAYER, J. C. L. MEEUSSEN, I. MUNIER, I. POINTEAU, D. SU, AND C. I. STEEFEL, *Benchmarks for multicomponent reactive transport across a cement/clay interface*, Computat. Geosci., 19 (2015), pp. 635–653.

[44] P. MARZAL, A. SECO, J. FERRER, AND C. GABALÓN, *Modeling multiple reactive solute transport with adsorption under equilibrium and nonequilibrium conditions*, Adv. Water Resour., 17 (1994), pp. 363–374.

[45] K. U. MAYER, E. O. FRIND, AND D. W. BLOWES, *Multicomponent reactive transport modeling in variably saturated porous media using a generalized formulation for kinetically controlled reactions*, Water Resour. Res., 38 (2002), pp. 13–1–13–21. 1174.

[46] C. D. MEYER, *Matrix Analysis and Applied Linear Algebra*, SIAM, 2000.

[47] F. M. M. MOREL AND J. G. HERING, *Principles and Applications of Aquatic Chemistry*, Wiley, New-York, 1993.

[48] M. F. MURPHY, G. H. GOLUB, AND A. J. WATHEN, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput., 21 (2000), pp. 1969–1972.

[49] J. M. NORDBOTTEN AND M. A. CELIA, *Geological Storage of CO2. Modeling Approaches for Large-Scale Simulation*, Wiley, 2012.

[50] J. RUBIN, *Transport of reacting solutes in porous media: Relation between mathematical nature of problem formulation and chemical nature of reactions*, Water Resour. Res., 19 (1983), pp. 1231–1252.

[51] Y. SAAD AND M. H. SCHULTZ, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.

[52] M. SAALTINK, C. AYORA, AND J.CARRERA, *A mathematical formulation for reactive transport that eliminates mineral concentrations*, Water Resour. Res., 34 (1998), pp. 1649–1656.

[53] M. SAALTINK, J. CARRERA, AND C. AYORA, *A comparison of two approaches for reactive transport modelling*, Journal of Geochemical Exploration, 69-70 (2000), pp. 97–101.

[54] M. SAALTINK, V. VILARRASA, F. DE GASPARI, O. SILVA, J. CARRERA, AND T. ROETTING, *A method for incorporating equilibrium chemical reactions into multiphase flow models for CO2 storage*, Adv. Water resour., 62 (2013), pp. 431–441.

[55] J. SAMPER, H. MA, J. L. CORMENZANA, C. LU, L. MONTENEGRO, AND M. . A. CUÑADO, *Testing Kd models of Cs+ in the near field of a HLW repository in granite with a reactive transport model*, Physics and Chemistry of the Earth, Parts A/B/C, 35 (2010), pp. 278–283. MIGRATION 2009, 12th International Conference on the Chemistry and Migration Behaviour of Actinides and Fission Products in the Geosphere.

[56] J. SAMPER, T. XU, AND C. YANG, *A sequential partly iterative approach for multicomponent reactive transport with CORE2D*, Computat. Geosci., 13 (2008), pp. 301–316.

[57] P. SIEGEL, R. MOSÉ, P. ACKERER, AND J. JAFFRÉ, *Solution of the advection-dispersion equation using a combination of discontinuous and mixed finite elements*, Int. J. Numer. Meth. Fl., 24 (1997), pp. 595–613.

[58] I. SIN, V. LAGNEAU, AND J. CORVISIER, *Integrating a compressible multicomponent two-phase flow into an existing reactive transport simulator*, Adv, Water Resour., 100 (2017), pp. 62 – 77.

[59] B. SOUTHWORTH, T. MANTEUFFEL, S. MCCORMICK, S. MUNZENMAIER, AND J. RUGE, *Reduction-based Algebraic Multigrid for Upwind Discretizations*, ArXiv e-prints, (2017).

[60] C. STEEFEL, D. DEPAOLO, AND P. LICHTNER, *Reactive transport modeling: An essential tool and a new research approach for the earth sciences*, Earth and Planetary Science Letters, 240 (2005), pp. 539–558.

[61] C. I. STEEFEL, C. A. J. APPELO, B. ARORA, D. JACQUES, T. KALBACHER, O. KOLDITZ, V. LAGNEAU, P. C. LICHTNER, K. U. MAYER, J. C. L. MEEUSSEN, S. MOLINS, D. MOULTON, H. SHAO, J. ŠIMŮNEK, N. SPYCHER, S. B. YABUSAKI, AND G. T. YEH, *Reactive transport codes for subsurface environmental simulation*, Computat. Geosci., 19 (2015), pp. 445–478.

[62] G. STRANG, *Introduction to Linear Algebra*, Wellesley-Cambridge Press, 4th ed., 2009.

[63] L. TRENTY, A. MICHEL, E. TILLIER, AND Y. L. GALLO, *A sequential splitting strategy for CO2 storage modelling*, in ECMOR X - 10th European Conference on the Mathematics of Oil Recovery, EAGE, 2006.

[64] A. J. VALOCCHI AND M. MALMSTEAD, *Accuracy of operator splitting for advection-dispersion-reaction problems*, Water Resour. Res., 28 (1992), pp. 1471–1476.

[65] J. A. WHITE AND R. I. BORJA, *Block-preconditioned Newton–Krylov solvers for fully coupled flow and geomechanics*, Computat. Geosci., 15 (2011), pp. 647–659.

[66] T. G. WRIGHT, *Eigtool.* http://www.comlab.ox.ac.uk/pseudospectra/eigtool/, 2002.

[67] T. G. WRIGHT AND L. N. TREFETHEN, *Large-scale computation of pseudospectra using ARPACK and Eigs*, SIAM J. Sci. Comput., 23 (2001), pp. 591–605.

[68] T. XU, E. SONNENTHAL, N. SPYCHER, AND K. PRUESS, *Toughreact – a simulation program for non-*

isothermal multiphase reactive geochemical transport in variably saturated geologic media: Applications to geothermal injectivity and CO2 geological sequestration, Comput. Geosci., 32 (2006), pp. 145 – 165.

[69] A. YAPPAROVA, T. GABELLONE, F. WHITAKER, D. A. KULIK, AND S. K. MATTHÄI, *Reactive transport modelling of dolomitisation using the new csmp++gem coupled code: Governing equations, solution method and benchmarking results*, Transport Porous Med., 117 (2017), pp. 385–413.

[70] G. T. YEH AND V. S. TRIPATHI, *A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components*, Water Resour. Res., 25 (1989), pp. 93–108.

[71] G.-T. YEH AND V. S. TRIPATHI, *A model for simulating transport of reactive multispecies components: Model development and demonstration*, Water Resour. Res., 27 (1991), pp. 3075–3094.

[72] F. ZHANG, G. YEH, AND J. PARKER, eds., *Groundwater Reactive Transport Models*, Behtham Publishers, 2012.