Nonlinear methods for reactive transport simulations

J. Erhel - INRIA-RENNES - team SAGE

INTERNATIONAL CONFERENCE

ON APPROXIMATION AND ITERATIVE METHODS

IN HONOR OF PROFESSOR CLAUDE BREZINSKI

Outline

- Chemistry model
- Transport model and coupled model
- Discrete nonlinear coupled model
- SOR-Newton iterative method
- Newton-type iterative methods
- Preliminary results

Nuclear waste disposal

grants with ANDRA and GDR MOMAS

Joint work with M. Kern, INRIA-ROCQUENCOURT

and C. de Dieuleveult, INRIA-RENNES

Chemistry model

Mass action laws (no precipitation)

$$\log x = S \log c + \log K_c$$

$$\log y = A \log c + B \log s + \log K_s$$

- $c \in \mathbb{R}^{N_c}$: aqueous components
- $s \in \mathbb{R}^{N_s}$: sorbed components
- $x \in \mathbb{R}^{N_x}$: secondary aqueous species
- $y \in \mathbb{R}^{N_y}$: secondary sorbed species
- $K_c \in \mathbb{R}^{N_x}$ and $K_s \in \mathbb{R}^{N_y}$: equilibrium constants
- $\overline{S} = \begin{pmatrix} S & 0 \\ A & B \end{pmatrix} \in \mathbb{R}^{N_x+N_y, N_c+N_s}$: stoechiometric coefficients

Chemistry model

Mass conservation

$$T = c + S^T x + A^T y$$
$$W = s + B^T y$$

- W fixed and given
- T given or coupled with transport model
- aqueous total $C = c + S^T x$
- fixed total $F = A^T y$

Chemistry model

Nonlinear equations

$$\Phi(a) - \begin{pmatrix} T \\ W \end{pmatrix} = 0$$

- $\Phi(a) = \exp(a) + \overline{S}^T exp(K + \overline{S}a)$
- $a = (\log c, \log s)^T$
- $K = (\log K_c, \log K_s)^T$

Jacobian matrix

$$J_c(a) = \mathrm{diag}(\exp(a)) + \overline{S}^T \, \mathrm{diag}(\exp(K + \overline{S}a))\overline{S}$$

Newton iterative method

Global method with line search or trust region

Chemistry model with precipitation

Mass Conservation

$$F = A^T y + D^T p$$

$p \in \mathbb{R}^{N_p}$: precipitated species

Mass action laws

Π : saturation index

$$\Pi = \log K_p + D \log c$$

$$\begin{cases} p_i = 0 & \text{if } \Pi_i < 1 \\ \Pi_i = 1 & \text{otherwise .} \end{cases}$$

Coupled transport and chemistry models

Advection-dispersion

$$\mathcal{L}(C) = \nabla.\left(C\vec{V}\right) - \nabla.\left(D\nabla C\right)$$

Transport of each chemical component

$$\omega\frac{\partial T_j}{\partial t} + \mathcal{L}(C_j) = 0, \quad j = 1, \ldots, N_c$$

Chemistry equations (no precipitation)

$$a = (\log c, \log s)^T$$
$$C = c + S^T \exp(\log K_c + S \log c) = C(a)$$
$$\Phi(a) - \begin{pmatrix} T \\ W \end{pmatrix} = 0$$

Discrete coupled model

Space discretization with a Finite Volume method with N_m points.

Discrete transport operator.

Variables T, C, F of order $N_m N_c$ and a of order $N_m(N_c + N_s)$.

TC formulation with explicit chemistry

$$\begin{cases} \frac{dT}{dt} + (L \otimes I)C = 0, \\ \Phi(a) - \begin{pmatrix} T \\ W \end{pmatrix} = 0, \\ C - C(a) = 0. \end{cases}$$

Discrete coupled model

TC formulation with implicit chemistry

$$\begin{cases} \frac{dT}{dt} + (L \otimes I)C = 0, \\ C - C(\Psi(T)) = 0, \\ a = \Psi(T). \end{cases}$$

CC formulation

$$\begin{cases} \frac{dC}{dt} + \frac{dF}{dt} + (L \otimes I)C = 0, \\ \Phi(a) - \begin{pmatrix} C + F \\ W \end{pmatrix} = 0, \\ C - C(a) = 0 \text{ or } F - F(a) = 0. \end{cases}$$

ODE and DAE frameworks

ODE framework with TC formulation and implicit chemistry

$$\frac{dT}{dt} + (L \otimes I)C(\Psi(T)) = 0.$$

- chemistry solver for each function evaluation
- No explicit Jacobian
- Chemistry solver as a black box
- Precipitation can be included

ODE and DAE frameworks

DAE framework with CC formulation

$$M\frac{dy}{dt} + f(y) = 0.$$

$$y = \begin{pmatrix} C \\ a \\ F \end{pmatrix}, M = \begin{pmatrix} I & 0 & I \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, f(y) = \begin{pmatrix} (L \otimes I)C \\ \Phi(a) - \begin{pmatrix} C + F \\ W \end{pmatrix} \\ F - F(a) \end{pmatrix}$$

- precipitation not easy to include
- chemistry functions
- Explicit Jacobian

ODE and DAE frameworks

Implicit time discretization

Euler implicit scheme and DAE : $My_{n+1} + \Delta t f(y_{n+1}) = My_n$

Nonlinear equations at each timestep

Nonlinear solvers

- block-SOR-Newton : Sequential Iterative Approach
- Newton-type : Global Approach
  - ⋆ Newton-block-SOR (inexact Newton)
  - ⋆ Newton-LU
  - ⋆ Newton-Krylov (inexact Newton)

Sequential Iterative Approach

Implicit Euler : $My_{n+1} + \Delta t f(y_{n+1}) = My_n$.

block-SOR-Newton

$y^1 = y_n$
For $k = 1, \ldots$until convergence
$(I + \Delta t(L \otimes I))C^{k+1} + F^k = C_n + F_n$
$\Phi(a) - \begin{pmatrix} C^{k+1} + F^k \\ W \end{pmatrix} = 0$
$F^{k+1} = F(a)$
End

- slow convergence
- chemistry solver at each iteration
- decoupled transport and chemistry

Global approach

Implicit Euler : $My_{n+1} + \Delta t f(y_{n+1}) = My_n$.

Newton-type method

$y^1 = y_n$

For $k = 1, \ldots$until convergence

$(M + \Delta t J^k)(y^{k+1} - y^k) = -(M(y^k - y_n) + \Delta t f(y^k))$

End

$$J^k = \begin{pmatrix} L \otimes I & 0 & 0 \\ -\begin{pmatrix} I \\ 0 \end{pmatrix} & \text{diag}(J_c(a^k)) & -\begin{pmatrix} I \\ 0 \end{pmatrix} \\ 0 & -\frac{dF}{da}(a^k) & I \end{pmatrix}$$

- Linearised chemistry at each Newton iteration
- fast convergence
- Coupled large sparse linear system

Global approach

Newton linearized equations :$(M + \Delta t J^k)(\Delta y) = z^k$.

Newton-block-SOR method

For $j = 1, \ldots$ until convergence

$$(I + \Delta t(L \otimes I))\Delta C^{j+1} + \Delta F^j = z_c^k$$

$$\text{diag}(J_c(a^k))\Delta a - \begin{pmatrix} I \\ 0 \end{pmatrix}(\Delta C^{j+1} + \Delta F^j) = z_a^k$$

$$\Delta F^{j+1} - \frac{dF}{da}(a^k)\Delta a = z_f^k$$

End

- decoupled transport and linearized chemistry
- slow convergence

Global approach

Newton linearized equations :$(M + \Delta t J^k)(\Delta y) = z^k$.

Newton-LU method

- efficient sparse linear solvers
- large matrix

Newton-Krylov method

- decoupled linearized chemistry and transport
- requires an efficient preconditioner

Preliminary results

Pyrite test case

4 components, 39 aqueous species and 13 fixed species

Somehow artificial with no precipitation

Comparison between SOR-Newton and Newton-LU

CPU times on a PC using Matlab

- SOR-Newton, timestep 5.510^{-4} : 4212 CPU seconds
- SOR-Newton, timestep 10^{-3} : 2465 CPU seconds
- Newton-LU, timestep 5.510^{-4} : 409 CPU seconds

Reactive transport - open questions

- DAE system with precipitation-dissolution ?
- Large 2D and 3D problems ?
- Reduced Coupled Model ?