ADVERTIMENT. La consulta d’aquesta tesi queda condicionada a l’acceptació de les següents condicions d'ús: La difusió d’aquesta tesi per mitjà del servei TDX (www.tesisenxarxa.net) ha estat autoritzada pels titulars dels drets de propietat intel·lectual únicament per a usos privats emmarcats en activitats d’investigació i docència. No s’autoritza la seva reproducció amb finalitats de lucre ni la seva difusió i posada a disposició des d’un lloc aliè al servei TDX. No s’autoritza la presentació del seu contingut en una finestra o marc aliè a TDX (framing). Aquesta reserva de drets afecta tant al resum de presentació de la tesi com als seus continguts. En la utilització o cita de parts de la tesi és obligat indicar el nom de la persona autora.

ADVERTENCIA. La consulta de esta tesis queda condicionada a la aceptación de las siguientes condiciones de uso: La difusión de esta tesis por medio del servicio TDR (www.tesisenred.net) ha sido autorizada por los titulares de los derechos de propiedad intelectual únicamente para usos privados enmarcados en actividades de investigación y docencia. No se autoriza su reproducción con finalidades de lucro ni su difusión y puesta a disposición desde un sitio ajeno al servicio TDR. No se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR (framing). Esta reserva de derechos afecta tanto al resumen de presentación de la tesis como a sus contenidos. En la utilización o cita de partes de la tesis es obligado indicar el nombre de la persona autora.

WARNING. On having consulted this thesis you’re accepting the following use conditions: Spreading this thesis by the TDX (www.tesisenxarxa.net) service has been authorized by the titular of the intellectual property rights only for private uses placed in investigation and teaching activities. Reproduction with lucrative aims is not authorized neither its spreading and availability from a site foreign to the TDX service. Introducing its content in a window or frame foreign to the TDX service is not authorized (framing). This rights affect to the presentation summary of the thesis as well as to its contents. In the using or citation of parts of the thesis it’s obliged to indicate the name of the author

Ph.D. Thesis

Mixing and speciation algorithms for geochemical and reactive transport problems

Author:
Francesca De Gaspari

Supervisors:
Dr. Jesús Carrera
Dr. Maarten W. Saaltink

GHS, Institute of Enviromental Assessment and Water Research (IDÆA), CSIC, Barcelona, Spain

GHS, Department of Geotechnical Engineering and Geosciences, Universitat Politècnica de Catalunya, UPC-BarcelonaTech, Barcelona, Spain

April 2015

This thesis was funded by the Universitat Politècnica de Catalunya - Escola Tècnica Superior d'Engiyners de Camins, Canals i Ports de Barcelona through the FPU-UPC program. The thesis was developed in the framework of the PANACEA project (Seventh Framework Programme FP7/2007-2013 under Grant agreements no. 282900).

Abstract

Geochemical and reactive transport modelling are essential tools in hydrogeology. They help to identify and assess geochemical processes occurring in applications such as groundwater contamination, water-rock interactions and geologic carbon sequestration. In this thesis we present methods for mixing and speciation calculations to be used for both interpretation of hydrochemical data and numerical modelling.

The first method presented allows solving geochemical speciation using redundant information, while acknowledging errors in data. Traditional speciation algorithms use a fixed number of data and equilibrium assumptions to calculate the concentration of the species present in a chemical system. We demonstrate that using redundant data (i.e., data or assumptions that exceed the minimum required and therefore are not strictly necessary) can improve speciation results by reducing estimation errors. In fact, we show that speciation errors decrease when increasing the number of redundant data.

The second method presented allows calculating mixing proportions of a number of end-members in a water sample from uncertain chemical data. Traditional methods for evaluating mixing ratios require the use of conservative tracers, which severely limits their applicability. The novelty of the method lies on the possibility of imposing equilibrium conditions on the mixture, while acknowledging kinetic reactions, which naturally leads to quantification of reactions. We applied the method to a freshwater-saltwater mixing problem in a set of samples collected by Sanz (2007), where we also characterized carbonate dissolution/precipitation and the production/consumption of CO_2.

These methods have been implemented in an Object-Oriented library called "CHEPROO++". This library can be used for hydrogeochemical calculations such as mixing waters linked to mass balance programs, which allows extending conservative transport simulators to solve reactive transport. One peculiarity of CHEPROO++ is the possibility of defining components decoupling constant activity species (CAS) such as, for example, pure equilibrium minerals or water (if the solution is sufficiently diluted). CHEPROO++ treats CAS as primary species. Decoupling CAS can be useful for speciation calculations because it allows reducing the system to be solved iteratively.

To check if decoupling CAS is advantageous, we applied the speciation algorithm that decouples CAS to a reactive transport application. In particular we used this speciation algorithm for the chemical step of the Sequential Iteration Approach for reactive transport modelling. We compared the proposed algorithm with the traditional method, which does not decouple CAS, on a one-dimensional domain where calcite is dissolving in equilibrium. Results show that decoupling CAS can decrease the number of iterations necessary for transport and chemistry calculations in case of equilibrium dissolution.

Resumen

El transporte reactivo y la modelación geoquímica son herramientas necesarias en hidrogeología, porque ayudan a identificar y cuantificar procesos geoquímicos que ocurren en diferentes aplicaciones como contaminación de aguas subterráneas, interacción agua-roca y secuestro geológico de CO_2. En estas tesis presentamos métodos para cálculos de mezcla y especiación para interpretación de datos hidrogeoquímicos y modelación numérica.

El primer método que presentamos permite resolver la especiación geoquímica con información redundante, y permite definir errores en los datos. Los algoritmos tradicionales de especiación utilizan un número fijo de datos y condiciones de equilibrio para calcular las concentraciones de las especies de un sistema químico. Con este método, demostramos que el uso de datos redundantes (p.ej. datos o hipótesis que exceden el mínimo requerido, y por tanto no son estrictamente necesarios) puede mejorar los resultados de la especiación reduciendo los errores de estimación. De hecho, mostramos como los errores de especiación disminuyen aumentando el número de datos redundantes.

El segundo método que presentamos permite calcular proporciones de mezcla de miembros extremo en una muestra a partir de datos químicos inciertos. Los métodos de mezcla tradicionales utilizan trazadores conservativos, y esto limita mucho sus aplicabilidad. La novedad de este método está en la posibilidad de imponer condiciones de equilibrio y considerar reacciones cinéticas, que naturalmente permiten cuantificar reacciones. Hemos aplicado el método a unas muestras representativas de mezcla entre agua dulce y agua salada tomadas por Sanz (2007), en las que también hemos caracterizado disolución/precipitación de carbonatos y producción/consumo de CO_2.

Estos métodos han sido implementados en una librería ("CHEPROO++") usando el paradigma de programación orientada a objetos. Este módulo se puede utilizar para cálculos hidrogeoquímicos como mezclas de agua si se acopla a programas de balance de masa, y también puede extender programas de transporte conservativo para resolver transporte reactivo. Una peculiaridad de CHEPROO++ es la posibilidad de definir componentes desacoplando las especies de actividad constante (CAS) como, por ejemplo, minerales puros en equilibrio o agua (si la solución está suficientemente diluida). CHEPROO++ considera CAS como especies primarias. El desacople de CAS puede ser útil para la especiación porque permite reducir el tamaño del sistema que se tiene que resolver iterativamente.

Para comprobar que desacoplar CAS es una ventaja, hemos aplicado el algoritmo de especiación que desacopla CAS a una aplicación de transporte reactivo. En particular, lo hemos utilizado para el paso químico del Sequential Iteration Approach, uno de los algoritmos para transporte reactivo. Hemos comparado el algoritmo propuesto con el tradicional, que no desacopla CAS, en un dominio unidimensional donde la calcita está disolviendo en equilibrio. Los resultados muestran que desacoplar CAS puede ayudar a reducir el número de iteraciones necesarias para el transporte y para la química en el caso de disolución en equilibrio.

Resum

El transport reactiu i la modelació geoquímica són eines necessàries en hidrogeologia, perquè ajuden a identificar i quantificar processos geoquímics que es donen en diferents aplicacions com contaminació d'aigües subterrànies, interacció aigua-roca i segrest geològic de CO_2. En aquesta tesi presentem mètodes per a càlculs de mescla i especiació per interpretació de dades hidrogeoquímiques i modelació numèrica.

El primer mètode que presentem permet resoldre la especiació geoquímica amb informació redundant, i permet definir errors en les dades. Els algoritmes tradicionals d'especiació utilitzen un nombre fix de dades i condicions d'equilibri per calcular les concentracions de les espècies d'un sistema químic. Amb aquest mètode, demostrem que l'ús de dades redundants (p. e. dades o hipòtesis que excedeixen el mínim requerit, i per tant no són estrictament necessàries) pot millorar els resultats de l'especiació reduint els errors d'estimació. De fet, ensenyem com els errors d'especiació disminueixen augmentant el nombre de dades redundants.

El segon mètode que presentem permet calcular proporcions de mescla de membres extrem en una mostra a partir de dades químiques incertes. Els mètodes de mescla tradicionals utilitzen traçadors conservatius, i això limita molt la seva aplicabilitat. La novetat d'aquest mètode està en la possibilitat d'imposar condicions d'equilibri i considerar reaccions cinètiques, que naturalment permeten quantificar reaccions. Hem aplicat el mètode a unes mostres representatives de mescla entre aigua dolça i aigua salada preses per Sanz (2007), en què també hem caracteritzat dissolució / precipitació de carbonats i producció/consum de CO_2.

Aquests mètodes han estat implementats en una llibreria ("CHEPROO ++") utilitzant el paradigma de programació orientada a objectes. Aquest mòdul es pot utilitzar per càlculs hidrogeoquímics com mescles d'aigua si s'acobla a programes de balanç de massa, i també pot estendre programes de transport conservatiu per resoldre transport reactiu. Una peculiaritat de CHEPROO ++ és la possibilitat de definir components desacoblant les espècies d'activitat constant (CAS) com, per exemple, minerals purs en equilibri o aigua (si la solució és prou diluïda). CHEPROO ++ considera CAS com a espècies primàries. El desacoblament de CAS pot ser útil per a la especiació perquè permet reduir la mida del sistema que s'ha de resoldre iterativament.

Per comprovar si desacoblar CAS és un avantatge, hem aplicat l'algoritme d'especiació que desacobla CAS a una aplicació de transport reactiu. En particular, l'hem utilitzat per al pas químic del Sequential Iteration Approach, un dels algoritmes per a transport reactiu. Hem comparat l'algoritme proposat amb el tradicional, que no desacobla CAS, en un domini unidimensional on la calcita està dissolent en equilibri. Els resultats mostren que desacoblar CAS pot ajudar a reduir el nombre d'iteracions necessàries per al transport i per a la química en el cas de dissolució en equilibri.

Riassunto

Il trasporto reattivo e la modellazione geochimica sono strumenti necessari in idrogeologia, perchè aiutano a identificare e quantificare processi geochimici che si verificano in diverse applicazioni come la contaminazione di acque sotterranee, nel caso di interazioni tra acqua e rocce o sequestro geologico della CO_2. In questa tesi presentiamo metodi per calcoli di miscele e speciazione, per interpretare dati geochimici e per modellazione numerica.

Il primo metodo che presentiamo permette di risolvere la speciazione geochimica utilizzando dati ridondanti che possono presentare errori. Gli algoritmi tradizionali di speciazione si avvalgono di un numero fisso di dati e condizioni di equilibrio termodinamico per calcolare la concentrazione delle specie presenti in un sistema chimico. Con questo metodo dimostriamo che l'uso di dati ridondanti (ovvero, dati o ipotesi che eccedono il minimo richiesto, e per tanto non sono strettamente necessari) può migliorare i risultati della speciazione riducendo gli errori di stima. Oltretutto, mostriamo come gli errori di speciazione diminuiscono aumentando il numero di dati ridondanti.

Il secondo metodo che presentiamo permette di calcolare le proporzioni di un numero di acque estremo in una miscela a partire da dati che presentano errori. I metodi tradizionali utilizzano traccianti conservativi, e ciò limita molto la loro applicabilità. La novità di questo metodo risiede nella possibilità di imporre condizioni di equilibrio e considerare anche reazioni cinetiche, il che permette di quantificare anche le reazioni. Abbiamo applicato il metodo a dei campioni rappresentativi di una miscela di acqua dolce e salata raccolti da Sanz (2007), nelle quali abbiamo anche caratterizzato dissoluzione/precipitazione di rocce carbonatiche e produzione/consumo di CO_2.

Questi metodi sono stati implementati in una libreria ("CHEPROO++") usando la programmazione a oggetti. Questo modulo può essere utilizzato per calcoli idrogeochimici come mescolare acque, accoppiato a programmi di bilanci di massa, o può ampliare le funzionalità di codici per il trasporto conservativo per risolvere trasporto reattivo. Una peculiarità di CHEPROO++ è la possibilità di definire componenti disaccoppiando le specie di attività costante (CAS) come, per esempio, minerali puri in equilibrio o acqua (se la soluzione è sufficientemente diluita). CHEPROO++ considera CAS come specie primarie. Disaccoppiarle può essere utile per la speciazione perchè permette ridurre la dimensione del sistema che dev'essere risolto iterativamente.

Per verificare se disaccoppiare CAS è vantaggioso, abbiamo applicato l'algoritmo di speciazione che disaccoppia CAS al trasporto reattivo. In particolare, l'abbiamo utilizzato per il passo chimico del Sequential Iteration Approach, uno degli algoritmi per il trasporto reattivo. Abbiamo comparato l'algoritmo proposto con quello tradizionale, che non disaccoppia CAS, in un dominio unidimensionale in cui la calcite dissolve in equilibrio. I risultati indicano che disaccoppiare CAS può aiutare a ridurre il numero di iterazioni necessarie per il trasporto e per la chimica nel caso della dissoluzione in equilibrio.

Acknowledgments

Firstly I would like to thank my advisors, Jesús Carrera and Maarten Saaltink. They have been a huge inspiration over the years. They have helped me and encouraged me even in the most difficult moments. I have learned so much by their side.

I want to thank Andrés, who gave me the chance to come to Barcelona (and a flat to stay in when I got here!). Thanks to him, Luit, Luis Manuel, and their combined colossal patience I learned how to program and the joy of a beer after finding a well hidden error in the code.

I shared worries, hopes and a lot of laughter with colleagues who became wonderful friends during my PhD years: Dani, Anna, Estanis, Marco, Edu, Anna, Violeta, Chris, Sona, Silvia, Jairo, Tanushree. It has been inspirational to have great researchers in the department; people such as Victor, Diogo, Felipe, Simonetta, Sergio and Pablo.

The GHS is full of lovely people, always ready to help and listen: Meritxell, Ester, Tere, Silvia Aranda, Albert Carles, and the guys in the lab who greet me with "Hola Francesca" as I walk down the corridor.

Of course I wouldn't have survived all these years without the unconditional support of my family: Gianfry, Susi, Stefy. Family who always listen to my stories, provide useful advice and most importantly, make me laugh when I need it.

Thanks to my friends who cheered me up throughout my thesis: Laura, Giuly, Vale, Ronan, Robin, Natalia, Andrew, Mario, Judy, Ian, Cat, Rich.

Finally, I am extremely grateful to my Steven for his infinite patience and moral support. I look forward to our next adventure!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Geochemical and reactive transport modelling in natural and artificial porous media are important for a variety of problems (Lichtner, 1996; Van Der Lee and De Windt, 2001; Steefel et al., 2005; Kulik et al., 2015) including geological sequestration of CO_2 (Xu et al., 2003; Knauss et al., 2005; Saaltink et al., 2013), remediation of contaminated groundwater (Xu et al., 2000; Mayer et al., 2002; Steefel et al., 2003), nuclear waste disposal (Dearlove et al., 1991; Metz et al., 2003; De Windt et al., 2004; Soler and Maeder, 2005), fresh-saltwater mixing zone of coastal aquifers (Rezaei et al., 2005; Walraevens and Van Camp, 2005; Han et al., 2014), alteration of geologic materials as a result of water-rock interactions (Steefel and Lasaga, 1994; Le Gallo et al., 1998; Blum et al., 2002) or evolution of evaporation fronts under extreme conditions (Gamazo et al., 2011).

Geochemical and reactive transport modelling have benefited from the developments of numerical codes. In fact, the complexity of geochemical problems due to the non-linearity of the equations, the coupling between different processes and the need for more realistic chemical systems accounting for more species and reactions simultaneously necessitate the use of numerical methods. In the last decades many codes have been released to deal with both geochemical and reactive transport calculations. Visual MINTEQ (Gustafsson, 2011), MIN3P (Mayer et al., 2002) and EQ3NR (Wolery, 1992), for example, are codes to deal with geochemical calculations, although PHREEQC (Parkhurst et al., 1999) is by far the most used. As for reactive transport TOUGHREACT (Xu et al., 2006), PFLOTRAN (Hammond et al., 2014) and RETRASO (Saaltink et al., 2004b) are only some of the many codes available.

Bea et al. (2009) developed a code named CHEPROO (CHemical PRocesses Object-Oriented) to simulate complex hydrobiogeochemical processes and to provide the functionalities required for coupling to any transport simulator in order to solve reactive transport. The novelty of CHEPROO was its Object-Oriented structure, in which every class represents a well defined geochemical entity (e.g., a phase, a species or a reaction). Bea et al. (2009) wrote CHEPROO in Fortran 90, which is not an Object-Oriented language. Still, he could include the main features of Object-Oriented programming: encapsulation (i.e., to hide details of the implementation behind interfaces) and polymorfism (i.e., to use the same name for methods implemented in different classes). He could also emulate inheritance (i.e., to share common data and methods of existing classes under a "parent" class).

Despite the impressive effort of emulating the Object-Oriented paradigm, Fortran 90 remains a very rigid language to modify and expand. The fact that Fortran 90 is not made for Object-Oriented caused CHEPROO to have thousands of thousands of code lines which are difficult to both read and modify. For this reason, we developed a tool for hydrogeochemical calculations using a proper Object-Oriented language: C++. We named it CHEPROO++.

CHEPROO++ is conceived as a portable library that offers methods to external programs to perform geochemical calculations such as mixing or speciating waters, and also to solve reactive transport. Regarding reactive transport calculations, CHEPROO++ was designed to be coupled with another program developed by the Hydrogeology Group (GHS) called PROOST (Slooten et al., 2010). PROOST is an Object-Oriented code for multiphase flow and transport simulations written in C++. CHEPROO++ and PROOST share some common libraries such as Qt, the XML input file parsing (i.e., reading), or Eigen, a C++ template library for linear algebra. However, CHEPROO++ can also be coupled to other conservative transport simulators in order to solve reactive transport.

1.2 Thesis goal and outline

The object of this thesis is to present CHEPROO++ as a new tool for hydrogeochemical calculations and to introduce its novel methods. The manuscript is organized in six chapters as follows

- Chapter 2 introduces CHEPROO++ and the types of equations needed to solve hydrogeochemical calculations: mass action laws and mass balance equations. In Chapter 2 it is also presented one of the novelties of CHEPROO++: a definition of components that allows to decouple constant activity species (e.g., pure minerals in equilibrium, proton if pH is fixed, etc.). Finally, after a brief introduction about Object-Oriented philosophy, the structure of CHEPROO++ classes is described.

- Chapter 3 presents a novel functionality implemented in CHEPROO++: a method to solve geochemical speciation that uses redundant data while acknowledging errors in data. Speciation is the process of evaluating the concentration of all the species in a chemical system from equilibrium conditions and measured data such as total concentration of components, electrical conductivity, pH, redox potential or gas partial pressure. Redundant data are measurements or assumptions that exceed the minimum required. Therefore they are not strictly necessary to speciate a water sample. Yet, they can be used in speciation. In this chapter we demonstrate that estimation errors can be reduced in geochemical speciation by means of redundant data. Moreover, the larger the amount of redundant data, the better the speciation in terms of errors of the estimated concentrations.

- Chapter 4 presents another peculiarity of CHEPROO++: an algorithm to calculate mixing proportions of a number of end-members in a sample and to quantify reactions leading to the chemical composition of the sampled water. The novelty of the algorithm is the possibility of imposing equilibrium conditions with reacting phases, such as minerals or gases, on the mixture. In this chapter we apply the method to a set of geochemical data relative to a brackish spring in Mallorca, Spain, which is the result of freshwater and seawater reactive mixing in depth of a coastal carbonate aquifer. Mixing proportions of fresh and seawater are calculated and verified on the basis of measurements such as electrical conductivity and dissolution/precipitation of carbonates are quantified together with a small consumption/production of CO_2.

- Chapter 5 presents a method to solve reactive transport with CHEPROO++ by means of the Sequential Iteration Approach (SIA). In particular, we propose to use the component definition presented in Chapter 2, where constant activity species are decoupled, in the chemical step. This renders the system to solve the chemistry smaller and may also reduce the number of iterations needed in the reactive transport simulations. In this chapter we compare the proposed and the traditional SIA methods in terms of quality of the results and performances, and we discuss the results.

- Chapter 6 presents the general conclusions of the thesis deduced from the previous chapters.

Chapter 2

CHEPROO++

This chapter describes CHEPROO++, which is a tool for hydrogeochemical calculations. First we present the two types of equations to solve these kind of calculations: mass action laws and mass balance equations. Then we introduce one peculiarity of CHEPROO++: its ability to decouple constant activity species. Afterwards, the Object-Oriented philosophy used to design CHEPROO++ is described. Finally, its structure is illustrated.

2.1 Hydrogeochemical calculations

Hydrogeochemical calculations are required for reactive transport modeling (Bundschuh and Zilberbrand, 2011; Rezaei et al., 2005; Steefel et al., 2005; Saaltink et al., 1998) and other scientific calculations such as mixing or evaporation of waters, reaction paths or geochemical inverse modeling (De Gaspari et al.; Appelo and Postma, 2010; Bethke, 2008). These types of calculations are useful to gain understanding of the geochemical processes occurring in both natural and artificial systems.

The main objective of hydrogeochemical calculations is to evaluate the set of concentrations $\mathbf{c}$ of the species present in the chemical system. Once $\mathbf{c}$ is known, it is possible to fully characterize the system from a chemical point of view (e.g., calculate saturation indices of minerals or evaluate thermodynamic properties of the phases such as density or viscosity, etc.).

Two sets of equations need to be solved in hydrogeochemical calculations: mass balances of the reactive species and thermodynamic constraints such as mass action laws. Let's analyze the latter first.

2.1.1 Mass action laws

Mass action laws (MAL) relate activities of species at equilibrium. They consist of N_{re} equations (one for every equilibrium reaction). In vector-matrix form, MAL can be written as

$$\mathbf{f}_{MAL} = \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \tag{2.1}$$

where $\mathbf{a}$ is a vector containing the activities of the N_s species. $\mathbf{S}_e$ is a matrix ($N_{re} \times N_s$) with the stoichiometric coefficients of the equilibrium reactions and $\mathbf{k}$ is a vector (N_{re}) of equilibrium constants. Algorithms to deal with system (2.1) are described in detail in Appendix (D.3). In general, they all derive from the fact that it is always possible to divide the species in two sub-sets of ($N_1 = N_s - N_{re}$) primary and $N_2 = N_{re}$ secondary species and express the activities of the secondary species as a function of the activities of the primary species. In fact, splitting $\mathbf{S}_e$ and $\mathbf{a}$ in two such parts (i.e., relative to primary and secondary species)

$$\mathbf{S}_e = \begin{pmatrix} \mathbf{S}_{e1} & \mathbf{S}_{e2} \end{pmatrix} \tag{2.2}$$

$$\mathbf{a} = \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{pmatrix} \tag{2.3}$$

and substituting (2.2) and (2.3) into (2.1) we obtain

$$\mathbf{f}_{MAL} = \mathbf{S}_{e1} \log \mathbf{a}_1 + \mathbf{S}_{e2} \log \mathbf{a}_2 - \log \mathbf{k} = 0 \tag{2.4}$$

where $\mathbf{S}_{e1}$ and $\mathbf{S}_{e2}$ contain the stoichiometric coefficients and $\mathbf{a}_1$ and $\mathbf{a}_2$ store the activities for primary and secondary species, respectively. The only necessary condition for splitting between primary and secondary species is $\mathbf{S}_{e2}$ to be invertible. Multiplying then (2.4) times $\mathbf{S}_{e2}^{-1}$ yields

$$\log \mathbf{a}_2 = \mathbf{S}_{e1}^{*} \log \mathbf{a}_1 + \log \mathbf{k}^{*} = 0 \tag{2.5}$$

where $\mathbf{S}_{e1}^{*} = -\mathbf{S}_{e2}^{-1}\mathbf{S}_{e1}$ and $\log \mathbf{k}^{*} = \mathbf{S}_{e2}^{-1} \log \mathbf{k}$.

Thus, if the activities of N_1 primary species are known (e.g., from mass balance equations), it is possible to calculate the activities of N_2 secondary species by means of (2.5). Mass balance equations are defined in the following section.

2.1.2 Mass balance formulations

Mass balance equations of reacting species can be formulated as distributed or lumped, i.e., with or without transport. The lumped mass balance for a i-th reactive species results from spatial integration of distributed mass balance and reads

$$\frac{d(\theta_{\alpha,i} \cdot c_{i,\alpha})}{dt} = \sum_{j=1}^{N_{re}} S_{e,ji} \cdot r_{e,j} + \sum_{l=1}^{N_{rk}} S_{k,li} \cdot r_{k,l} + f_i \qquad i = 1, ..., N_s \tag{2.6}$$

where $c_{i,\alpha}$ is the species concentrations in the phase α, expressed in mol/V_α, and $\theta_{\alpha,i}$ (V_α/V_{tot}) is the volumetric fraction of the phase α to which the i-th species belongs. $S_{e,ji}$ is the stoichiometric coefficient of the i-th species in the j-th equilibrium reaction and $r_{e,j}$ is the equilibrium reaction rate of the j-th equilibrium reaction. $S_{k,li}$ and $r_{k,l}$ are equivalent to $S_{e,ji}$ and $r_{e,j}$ for kinetic reactions. f_i (moles per unit time per volume of porous medium) is a non-chemical sink source term.

Equation (2.6) can be modified for distributed formulation (i.e., to include transport) by adding a linear transport operator $L_\alpha(c_{i,\alpha})$ such that

$$\frac{\partial(\theta_{\alpha,i} \cdot c_{i,\alpha})}{\partial t} = L_\alpha(c_{i,\alpha}) + \sum_{j=1}^{N_{re}} S_{e,ji} \cdot r_{e,j} + \sum_{l=1}^{N_{rk}} S_{k,li} \cdot r_{k,l} + f_i \qquad i = 1, ..., N_s \tag{2.7}$$

where

$$L_\alpha(c_{i,\alpha}) = \nabla \left[\theta_{\alpha,i} \cdot (\mathbf{D}_\alpha \nabla c_{i,\alpha} - \mathbf{q}_\alpha c_{i,\alpha})\right] \qquad i = 1, ..., N_s \tag{2.8}$$

$\mathbf{q}_\alpha$ is the Darcy velocity and $\mathbf{D}_\alpha$ is the tensor accounting for molecular diffusion and for hydrodynamic dispersion. Notice that $\mathbf{q}_\alpha$ and $\mathbf{D}_\alpha$ are assumed to be the same for all the species belonging to phase α. Equation (2.7) is the reactive Advection-Dispersion Equation.

Note that (2.6) is equivalent to (2.7) integrated on the total volume. We can therefore explain (2.7) bearing in mind that (2.6) is a particular case of (2.7) in which $L_\alpha(c_{i,\alpha}) = 0$.

Mass balance equations (2.6) and (2.7) are formulated in terms of volumetric concentration, $c_{i,\alpha}$. However, geochemical codes usually deal with different concentration units, x_i, for species belonging to different phases (Gamazo et al., 2012). Each phase presents a different relationship between $c_{i,\alpha}$ and x_i. For aqueous phase, for example, x_i is expressed in molality (moles per mass of solvent, usually liquid water), which is also the unknown in the mass action laws, and its relationship with $c_{i,\alpha}$ can be expressed as

$$c_{i,aq} = \rho_l \omega_l^{H2O} x_i \tag{2.9}$$

where ρ_l is the liquid density and ω_l^{H2O} is the water mass fraction.

For gas phases, x_i is the gas partial pressure and its relationship with $c_{i,gas}$ can be deduced by means of the gas law

$$c_{i,gas} = \frac{x_i}{Z_i RT} \tag{2.10}$$

where Z_i values can be calculated from virial relationships or considered equal to one in case the gas is ideal. R and T are the universal gas constant and temperature, respectively.

For pure mineral phases molarity is constant and can be computed from the molar volume, constant property of the mineral phase. Therefore in hydrogeochemical calculations the unknown is considered to be the volumetric fraction, θ_{min}. However, if the mineral is not pure, mineral species must satisfy

$$\sum_{z=1}^{Nmin} \chi_{min}^{z} = 1 \tag{2.11}$$

where χ_{min}^{z} represents the molar fraction of the z-th mineral species in the mineral phase *min*, and *Nmin* is the number of mineral species in the mineral phase *min*.

To facilitate a compact matrix-vector notation of the mass balance equations, we define a concentration c_i' for a i-th species expressed in *mol* per unit volume of medium

$$c_i' = \theta_{\alpha,i} \cdot c_{i,\alpha} \qquad i = 1, ..., N_s \tag{2.12}$$

which allows to rewrite (2.7) for a set of N_s species as follows

$$\frac{\partial \mathbf{c}'}{\partial t} = L(\mathbf{c}') + \mathbf{S}_e^t \mathbf{r}_e + \mathbf{S}_k^t \mathbf{r}_k + \mathbf{f} \tag{2.13}$$

where $\mathbf{S}_e$ $(N_{re} \times N_s)$ and $\mathbf{S}_k$ $(N_{rk} \times N_s)$ are the equilibrium and kinetic stoichiometric matrices, respectively, and $\mathbf{r}_e$ (N_{re}) and $\mathbf{r}_k$ (N_{rk}) are the equilibrium and kinetic reaction rates.

Equation 2.13 can be differentiated by phases

$$\frac{\partial}{\partial t}\begin{pmatrix} \mathbf{c}'_{aq} \\ \mathbf{c}'_{gas} \\ \mathbf{c}'_{min} \\ \mathbf{c}'_{ads} \end{pmatrix} = \begin{pmatrix} L(\mathbf{c}'_{aq}) \\ L(\mathbf{c}'_{gas}) \\ 0 \\ 0 \end{pmatrix} + \mathbf{S}_e^t \mathbf{r}_e + \mathbf{S}_k^t \mathbf{r}_k + \mathbf{f} \tag{2.14}$$

Notice that solid species (mineral and adsorbed species) are considered immobile, therefore the transport operator is null for them.

Vectors $\mathbf{c}'$ and $\mathbf{r}_e$ are the (N_s+N_{re}) unknowns in equation (2.14), which can be a large number for chemical systems with many reactions and species. In particular, for reactive transport calculations on grids with many nodes (N), each node contains $(N_s + N_{re})$ unknowns (i.e., total number of unknowns $= N\cdot(N_s + N_{re})$). Moreover, no explicit expression is available to calculate $\mathbf{r}_e$. For these reasons, it is common practice, when dealing with these types of equations, to reduce the number of unknowns by multiplying them times a matrix $\mathbf{U}$ ($N_1 \times N_s$), orthogonal to $\mathbf{S}_e$ so that $\mathbf{U}\mathbf{S}_e^t = \mathbf{0}$ (Saaltink et al., 1998). This way, the source term relative to equilibrium reactions is eliminated and the unknowns of the problem are reduced to the concentrations of the $N_s - N_{re}$ species. The mass balance can then be rewritten as

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{U} \begin{pmatrix} L(\mathbf{c}'_{aq}) \\ L(\mathbf{c}'_{gas}) \\ 0 \\ 0 \end{pmatrix} + \mathbf{U}\mathbf{S}_k^t \mathbf{r}_k + \mathbf{U}\mathbf{f} \tag{2.15}$$

where $\mathbf{u} = \mathbf{U}\mathbf{c}'$ are called "components". The concept of "components" is fundamental in hydrogeochemical calculations, because the number of components is the number of independent contituents of chemical systems. Equation (2.15) can also be differentiated by phases as follows

$$\frac{\partial \mathbf{u}_{aq}}{\partial t} + \frac{\partial \mathbf{u}_{gas}}{\partial t} + \frac{\partial \mathbf{u}_{min}}{\partial t} + \frac{\partial \mathbf{u}_{ads}}{\partial t} = L(\mathbf{u}_{aq}) + L(\mathbf{u}_{gas}) + \mathbf{U}\mathbf{S}_k^t \mathbf{r}_k + \mathbf{U}\mathbf{f} \tag{2.16}$$

Regardless of the formulation, with or without transport, mass balance equations allow the modeller to calculate the vector of concentrations of components, $\mathbf{u}$. The goal of geochemical calculations, however, is to evaluate a vector of concentrations of species. This can be achieved by means of the definition of components ($\mathbf{u} = \mathbf{U}\mathbf{c}'$) and MAL. In essence, to calculate concentrations, a speciation must be solved (see Chapter 3 for definition of speciation). CHEPROO++ is able to perform speciation calculations. Thus, at its core there are several speciation algorithms.

Speciation and reactive transport algorithms depend strongly on the way the component matrix $\mathbf{U}$ is defined. Since the definition of $\mathbf{U}$ is one of the novelties of CHEPROO++, it is presented in the following section.

2.2 Components definition in CHEPROO++

Several methods exist to build the component matrix $\mathbf{U}$. A list and a detailed explanation can be found in Bundschuh and Zilberbrand (2011). The method proposed by Steefel and MacQuarrie (1996), based on the Gauss-Jordan elimination, is by far the most used because, as explained in Bundschuh and Zilberbrand (2011) and Saaltink et al. (1998), it can lead to components with a physical meaning such as total analytical concentrations of species, total inorganic carbon, etc. It makes use of matrix $\mathbf{S}_{e1}^*$ defined in equation (2.5) to build $\mathbf{U}$ as

$$\mathbf{U} = \begin{pmatrix} \mathbf{I} & \mathbf{S}_{e1}^{*t} \end{pmatrix} \tag{2.17}$$

Consider, for example, the following chemical system

$$\begin{aligned} &\text{R1}: CO_3^{2-} + H^+ \rightleftharpoons HCO_3^- \\ &\text{R2}: CO_{2(aq)} + H_2O \rightleftharpoons HCO_3^- + H^+ \\ &\text{R3}: OH^- + H^+ \rightleftharpoons H_2O \\ &\text{R4}: CaCO_{3(s)} + H^+ \rightleftharpoons Ca^{2+} + HCO_3^- \end{aligned} \tag{2.18}$$

If we choose Ca^{2+}, H^+, Cl^-, H_2O and HCO_3^- to be primary ($N_1 = 5$) and CO_3^{2-}, $CO_{2(aq)}$, OH^- and $CaCO_{3(s)}$ to be secondary ($N_2 = 4$), the component matrix reads

$$\mathbf{U} = \begin{array}{c} \begin{array}{ccccccccc} Ca^{2+} & H^{+} & Cl^{-} & H_2O & HCO_3^{-} & CO_3^{2-} & CO_{2(aq)} & OH^{-} & CaCO_{3(s)} \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & -1 & 1 & -1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array} \tag{2.19}$$

and the corresponding components $\mathbf{u} = \mathbf{U}\mathbf{c}'$

$$\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} c'_{Ca^{2+}} + c'_{CaCO_{3(s)}} \\ c'_{H^+} - c'_{CO_3^{2-}} + c'_{CO_{2(aq)}} - c'_{OH^-} - c'_{CaCO_{3(s)}} \\ c'_{Cl^-} \\ c'_{H_2O} - c'_{CO_{2(aq)}} + c'_{OH^-} \\ c'_{HCO_3^-} + c'_{CO_3^{2-}} + c'_{CO_{2(aq)}} + c'_{CaCO_{3(s)}} \end{pmatrix} \tag{2.20}$$

However, it has become common lately (Saaltink et al., 1998; De Simoni et al., 2005; Gamazo et al., 2012) to look for alternative definitions of $\mathbf{U}$ in order to reduce the number of components (i.e., the number of independent variables of the system) by eliminating constant activity species (CAS). Very diluted solutions, for example, can be seen almost as pure water, and therefore H_2O activity can be considered unity. Activity of gases as well can be equal to their total pressure if the phase is pure, and minerals have unity activity, if the mineral phase is pure. All these species whose activity is fixed and known a priori can be eliminated. Saaltink et al. (1998) and De Simoni et al. (2005) proposed two methods to eliminate CAS. The first consists in multiplying $\mathbf{U}$, previously built by means of definition (2.17), times an elimination matrix $\mathbf{E}$ to get rid of CAS. De Simoni et al. (2005) instead directly eliminates CAS from the vector of species and then builds $\mathbf{U}$ by means of (2.17).

CHEPROO++ builds the component matrix decoupling CAS, i.e., considering them as primary species. This means that it defines $\mathbf{S}_e$ and $\mathbf{a}$ as follows

$$\mathbf{S}_e = \begin{pmatrix} \mathbf{S}_{e1,nc} & \mathbf{S}_{e1,CAS} & \mathbf{S}_{e2} \end{pmatrix} \tag{2.21}$$

$$\mathbf{a} = \begin{pmatrix} \mathbf{a}_{1,nc} \\ \mathbf{a}_{1,CAS} \\ \mathbf{a}_2 \end{pmatrix} \tag{2.22}$$

where $\mathbf{a}_{1,nc}$ ($N_{1,nc} = N_s - N_{re} - N_{CAS}$) and $\mathbf{a}_{1,CAS}$ (N_{CAS}) are the vectors of reduced primary species and CAS, respectively. $\mathbf{S}_{e1,nc}$ and $\mathbf{S}_{e1,CAS}$ are the stoichiometric matrices of reduced primary species and CAS, of dimension ($N_{re} \times N_{1,nc}$) and ($N_{re} \times N_{CAS}$), respectively.

By means of (2.21) and (5.15) $\mathbf{f}_{MAL}$ can be re-defined to account explicitly for CAS:

$$\mathbf{f}_{MAL} = \mathbf{S}_{e1,nc} \log \mathbf{a}_{1,nc} + \mathbf{S}_{e1,CAS} \log \mathbf{a}_{1,CAS} + \mathbf{S}_{e2} \log \mathbf{a}_2 - \log \mathbf{k} = 0 \tag{2.23}$$

It is still possible to express $\mathbf{a}_2 = f(\mathbf{a}_1)$ as we did in (2.5) by multiplying (2.23) times $\mathbf{S}_{e2}^{-1}$ so that

$$\log \mathbf{a}_2 = \mathbf{S}^*_{e1,nc} \log \mathbf{a}_{1,nc} + \mathbf{S}^*_{e1,CAS} \log \mathbf{a}_{CAS} + \log \mathbf{k}^* = 0 \tag{2.24}$$

where $\mathbf{S}^*_{e1,nc} = -\mathbf{S}_{e2}^{-1}\mathbf{S}_{e1,nc}$ and $\mathbf{S}^*_{e1,CAS} = -\mathbf{S}_{e2}^{-1}\mathbf{S}_{e1,CAS}$.

The definition of the component matrix implemented in CHEPROO++ still uses the Gauss-Jordan elimination but accounts for definition (2.21). It reads

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_{nc} \\ \mathbf{U}_{CAS} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & 0 & \mathbf{S}^{*t}_{e1,nc} \\ 0 & \mathbf{I} & \mathbf{S}^{*t}_{e1,CAS} \end{pmatrix} \tag{2.25}$$

where $\mathbf{U}_{nc}$ ($N_{1,nc} \times N_s$) and $\mathbf{U}_{CAS}$ ($N_{CAS} \times N_s$) are the parts of the component matrix relative to reduced primary and constant activity species. System (2.18), for example, presents two CAS: H_2O and $CaCO_{3(s)}$ ($N_{CAS} = 2$). If we choose then Ca^{2+}, H^+, Cl^- as reduced primary species ($N_{1,nc} = 3$) and HCO_3^-, CO_3^{2-}, $CO_{2(aq)}$ and OH^- as secondary ($N_2 = 4$), the component matrix is the following

$$\mathbf{U} = \begin{array}{c} \begin{array}{ccccccccc} Ca^{2+} & H^+ & Cl^- & H_2O & CaCO_{3(s)} & HCO_3^- & CO_3^{2-} & CO_{2(aq)} & OH^- \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 2 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \end{array} \tag{2.26}$$

and the total concentrations of components read

$$\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} c'_{Ca^{2+}} - c'_{HCO_3^-} - c'_{CO_3^{2-}} - c'_{CO_{2(aq)}} \\ c'_{H^+} + c'_{HCO_3^-} + 2c'_{CO_{2(aq)}} - c'_{OH^-} \\ c'_{Cl^-} \\ c'_{H_2O} - c'_{CO_{2(aq)}} + c'_{OH^-} \\ c'_{CaCO_{3(s)}} + c'_{HCO_3^-} + c'_{CO_3^{2-}} + c'_{CO_{2(aq)}} \end{pmatrix} \tag{2.27}$$

The advantage of defining the CAS as primary species is that the first three components (u_1, u_2 and u_3 of definition 2.27) do not depend on the CAS. Neither do the mass action laws depend on the CAS, as their activity is constant. Therefore, CAS concentrations can be calculated explicitly once the concentrations of primary and secondary species are known. This concept will be further explained in Chapter 5.

Clearly definition (2.25) of components may vary spatially and in time, e.g., when some CAS such as equilibrium pure minerals are not present in all the domain or they appear/disappear due to precipitation/dissolution reactions. This is taken into account in CHEPROO++ through the possibility of defining more "components zones", which are characterized by a different sub-set of reduced $N_{1,nc}$ primary species. Each definition of "components zone" is stored in a different object of the class CLocalChemicalSystem, which is described together with the other classes in the following section.

2.3 CHEPROO++ and Object Oriented philosophy

In the previous sections we have described the concept of hydrogeochemical calculations. Along with it, we have introduced chemical entities that are common to every geochemical calculation: phases, reactions, reaction rates, species, etc. These entities invite to define specific categories of objects. For this reason, CHEPROO++ was developed following an Object-Oriented (OO) philosophy in which every class represents one geochemical entity.

An important requirement that has been taken in account to design CHEPROO++ is the flexibility of the modelling software that has to be guaranteed. As CHEPROO++ is conceived as a portable library, it can be easily coupled to different codes (e.g. conservative transport simulator) and modified without too much difficulty thanks to its OO modularity. The OO design methodology is widely credited for reducing coupling (i.e., interdependence of different parts of a body of software) and favouring code reuse (Bea et al., 2009; Slooten et al., 2010). At its core are the concepts of "class" and "object". The class is a definition of a category (e.g., reactions) and of the possible operations over this category (e.g., calculate the logK of the reaction given a temperature value). As such, a class definition contains both a storage structure and a collection of functions. The storage structure is defined as a list of elements of the class or of other classes, called "the attributes" (a reaction class could have as attributes, for example, the logK and the stoichiometric coefficients of the species participating to the reaction). An "object" is a particular elements of the set (for example, a reaction called "calcite dissolution/precipitation"). Therefore, classes can be used to represent a wide range of concepts in the context of a hydrogeochemical code. To understand how the use of classes can lead to code re-use it is important to know how they can be related. Classes can have two types of relationships with other classes. The first of these is the "has-a" relationship, or composition. This type of relationship is formed when a class has one or more attributes belonging to another class. The second is the "is-a" relationship or inheritance. This relationship allows specifying a class representing a specific concept (e.g. "aqueous phase") being a specialization of a more general class (e.g. "phase"). To define the specific class, only those methods and attributes need to be defined that are somehow different from those of the general class. Both relationships allow code re-use. The reduction in coupling is achieved by two mechanisms: first, common interfaces are defined for different specialization classes (e.g., "aqueous phase" and "mineral phase") so that the code that uses a class ("phase") becomes independent of the particular specialization that is being used. Second, it is discouraged to directly access class attributes from outside the class. Instead, an interface is defined to ask for information or give it to a class using methods. That way, a code that uses a particular class becomes independent of the storage structure used by the class. Besides, this structure of the code facilitates its expansion, since developers can implement new functions and specializations (inherited classes) without knowledge of the wider program structure.

2.4 Structure of CHEPROO++

This section contains a brief description of the main classes of CHEPROO++. The structure of the code is shown in Figure 2.1, while Figure 2.2 illustrates the structure of the specializations of the class "CPhase". This structure is based on the one presented by Bea et al. (2009). However, there are some differences, especially with respect to classes related to the definition of components and primary/secondary species (i.e., CGlobalChemicalSystem and CLocalChemicalSystem).

- All CHEPROO++ classes inherit from a CCheprooBase class, which in turn inherits from a classe called CBase. CBase is a class shared between CHEPROO++ and PROOST, an Object-Oriented code for multiphase flow and transport that has been developed in the GHS group in the past few years (Slooten et al., 2010). Its main functionality is to call an "Object Factory" to create instances of classes from a name and a type (i.e., specialization of the class) defined in an input file.

- The CCheprooPlusPlus class is a link between the programs using CHEPROO++ and the

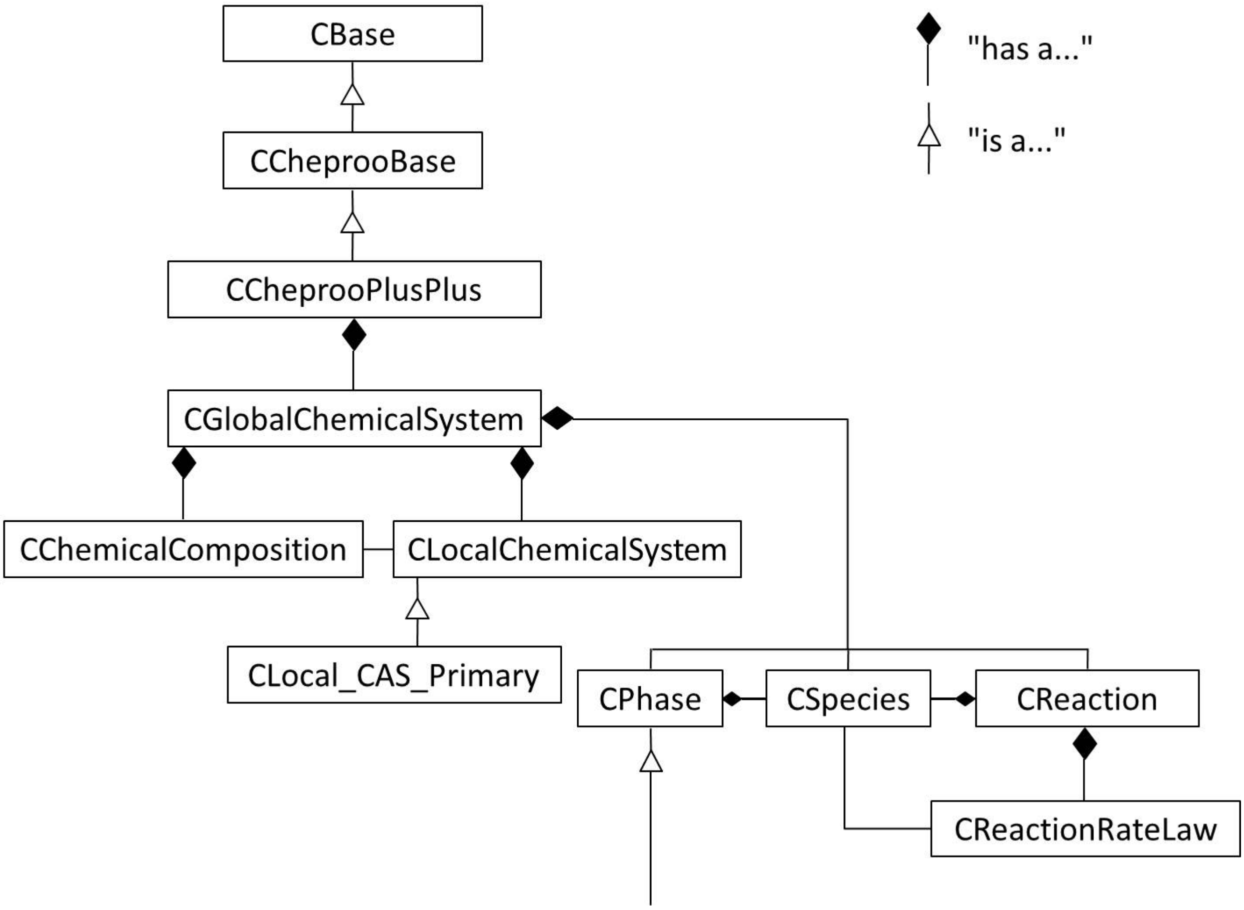


Figure 2.1: Structure of CHEPROO++ classes.

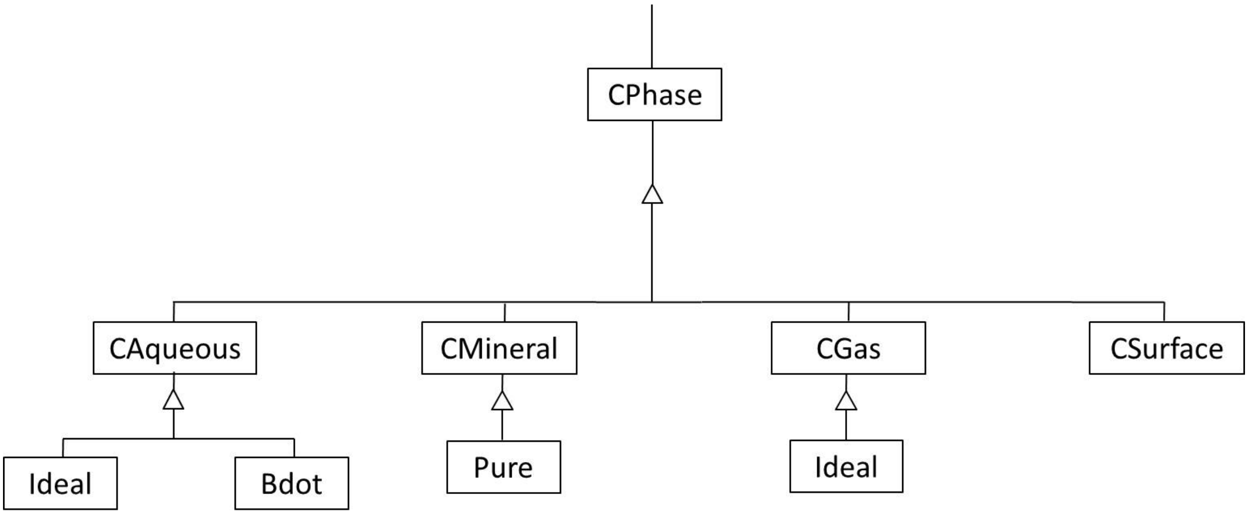


Figure 2.2: Specializations of the phase class.

CGlobalChemicalSystem class. As such, it offers functionalities (methods) that can be used by another program, directly calling the methods that are needed (method "MixWaters" for example). CCheprooPlusPlus in charge of starting the reading process which triggers the initializations of all the objects in the library ("ReadAndInitialize" method, which can be used both for reactive transport and geochemical modelling initializations).

- CChemicalComposition contains a definition of a local chemistry as a combination of phases (waters, minerals, gases, surfaces). Therefore, its attributes are the state variables that are used in geochemical calculations and reactive transport: concentrations, activity coefficients, temperature, gas/liquid pressure, nodes volume, volumetric fractions and reactive area of minerals etc. Besides methods to "set" and "get" state variables value (i.e., to assign or obtain a value of state variables), CChemicalComposition contains methods that can be used to calculate the chemistry of one water during reactive transport calculations, e.g., to solve the chemical step for one water in the Sequential Iteration Approach (see Chapter 5 for details).
- The class CGlobalChemicalSystem is in charge of creating all the objects defined by the user in the input file of CHEPROO++ (i.e., species, phases, reactions, chemical compositions, local chemical systems) and of storing them in lists. The association between local chemical systems and chemical compositions is also stored in this class. One local chemical system can be associated to one or more chemical compositions, but one chemical composition can be associated only to one local chemical system.
- The class CLocalChemicalSystem represents a sub-domain of the problem characterized by a unique definition of components. As explained in Section 2.2, CAS can be decoupled in hydrogeochemical calculations to reduce the number of in of independent variables of the system (i.e., primary species). Therefore, every instance of the class CLocalChemicalSystem is characterized by a different set of "reduced" primary species. The main function of CLocalChemicalSystem is to solve the speciation.
- The CReaction class represents a single reaction of the system. Therefore, its attributes are the stoichiometric coefficients and the coefficients necessary to calculate the equilibrium constant as a function of the temperature. CReaction is in charge of creating a pointer to the class CReactionRateLaw (also an attribute of CReaction) in case of kinetic reactions. It also contains a method to evaluate the ratio IAP/k (being IAP the ion activity product and k the equilibrium constant) for the reaction, given concentration and activity vectors.
- The kinetic reaction rate law expressions are evaluated by means of the class CReactionRateLaw, which presents two inherited classes (CReactionRateLawLasaga and CReactionRateLawMonod).
- The class CSpecies represents a single species present in the system. Depending on the phase they belong to, they can have different attributes (e.g. electrical charge, ion size and molecular weight for aqueous species; molar volume for mineral and gaseous species).
- The CPhase class represents a homogeneous part that constitutes a system (e.g., aqueous phase, mineral phase).This is a parent class, from which the four child phase classes (aqueous, mineral, gas and surface) derive. Its main function is to evaluate not only the activity coefficients for each species and their derivatives wrt secondary concentrations, but also properties such density, viscosity and their derivatives wrt pressure, temperature and composition.

Chapter 3

Use of redundant data to reduce estimation errors in geochemical speciation *

3.1 Introduction

Geochemical modeling is important in Earth Sciences. In particular, it is required to assess problems ranging from weathering to the characterization of the chemical composition of water and processes that could influence its quality (Appelo and Postma, 2010; Bethke, 2008). Geochemical speciation is a key step of geochemical modelling that consists of evaluating concentrations of all the species in a chemical system from measured data (e.g., total concentrations of components, pH, alkalinity, gas partial pressures, electrical conductivity, redox potential) and equilibrium constraints. For this reason, it is often termed thermodynamic speciation.

Speciation requires the solution of a non-linear system of equations and a lot of research has been focused on numerical issues that might arise when solving these equations. Several methods have been proposed to solve chemical equilibrium in a robust way in order to guarantee the convergence (Paz-García et al., 2013; Carrayrou et al., 2002; Brassard and Bodurtha, 2000) and many codes have also been released to deal with geochemical speciation calculations: GEMS3K (Kulik et al., 2013), Visual MINTEQ (Gustafsson, 2011), CHEPROO (Bea et al., 2009), ORCHESTRA (Meeussen, 2003), MIN3P (Mayer et al., 2002), PHREEQC (Parkhurst et al., 1999) and its interactive version, PHREEQCi (Charlton et al., 1997), EQ3NR (Wolery, 1983, 1992) and WATEQ4F (Ball and Nordstrom, 1991).

Speciation calculations are subject to sources of uncertainty which can derive from uncertainty in thermodynamic data, such as equilibrium constant values, or from errors in chemical analyses (i.e., analytical errors). These types of random errors can be referred to as "aleatory uncertainty". Misjudgment in the definition of the chemical system, such as failure to account for some reactions or discarding others, can also lead to errors in speciation. These can be defined as "epistemic uncertainty". They arise from an incomplete or inadequate characterization of the system (Gupta et al., 2012), such as assuming the neutrality of a solution when it is not electrically balanced, or imposing equilibrium with phases that are not. The effect of errors propagation in geochemical calculations has been extensively studied. In particular, the effect of aleatory errors has been investigated by Weber et al. (2006); Denison and Garnier-Laplace (2005); Ödegaard-Jensen et al. (2004); Nitzsche et al. (2000); Cabaniss (1999, 1997); Criscenti et al. (1996) and Merino (1979). Smith et al. (1999) examined the connection between aleatory and epistemic errors. Although the origin and propagation effects of both types of errors are different, they can be treated in the same way through probabilistic methods.

All these studies use a fixed number of data to solve the speciation. Geochemical speciation,

---

[*] This Chapter is based on the paper De Gaspari et al. (2015)

in fact, requires a fixed minimum number of data, including equilibrium assumptions, equal to the number of independent variables of the system (i.e., number of species). For example, a carbonate system is characterized by four degrees of freedom (see Section 3.2.1). Therefore, four data (e.g., total concentrations of inorganic carbon and calcium, pH) or hypotheses (e.g., water activity equal to 1) are needed. However, extra data might be available (e.g., alkalinity, electrical conductivity or redox potential) or extra assumptions about the system might be made (e.g., equilibrium with calcite or $CO_{2(g)}$ in equilibrium with the atmosphere). Chemical analyses of waters, for example, often provide extra data and also the analytical errors associated to each of them.

We term these extra data as redundant and we claim that speciation calculations can benefit from their use, while aknowledging analytical errors.

The aim of this paper is to present an algorithm to include redundant data in speciation calculations and to prove that their use can improve the results by reducing estimation errors. We also claim that increasing the number of redundant data helps decreasing even further the estimation errors.

3.2 Methodology

We start by analyzing a speciation example to clarify the differences between the traditional and the proposed method. This allows us to formalize the problem statement and to propose a solution algorithm.

3.2.1 Speciation of a carbonate system

We consider the problem of calculating the concentrations of dissolved species in a carbonate system. This system has received extensive attention from the scientific community, e.g. to study seawater intrusion in carbonate coastal aquifers (Werner et al., 2013; Bear, 1999; Back et al., 1979, amongst many others), including geochemical processes occurring in the mixing zone between freshwater and saltwater (Sanz et al., 2011; De Simoni et al., 2007; Rezaei et al., 2005), and to analyze the feasibility of CO_2 sequestration in deep aquifers (Saaltink et al., 2013; Duan and Li, 2008; Xu et al., 2006).

The most simple chemical system consists of 9 species ($N_s = 9$) and the following 5 equilibrium reactions ($N_{re} = 5$)

$$\begin{array}{ll}
OH^- + H^+ \rightleftharpoons H_2O & \log K_1 = 13.995 \\
CO_3^{2-} + H^+ \rightleftharpoons HCO_3^- & \log K_2 = 10.328 \\
CO_{2(aq)} + H_2O \rightleftharpoons HCO_3^- + H^+ & \log K_3 = -6.344 \\
CO_{2(g)} + H_2O \rightleftharpoons HCO_3^- + H^+ & \log K_4 = -7.813 \\
CaCO_{3(s)} + H^+ \rightleftharpoons Ca^{2+} + HCO_3^- & \log K_5 = 1.848
\end{array} \tag{3.1}$$

The number of degrees of freedom of this system is $N_s - N_{re} = 4$. This means that 4 data or assumptions are needed to solve the speciation problem. Speciation codes normally use this criterion. Optionally species with constant activity can be decoupled and eliminated, e.g., water if the system is sufficiently diluted ($\mathrm{a}_{H_2O} = 1$) or proton if the pH is fixed ($\mathrm{a}_{H^+} = 10^{-pH}$), to reduce the number of unknowns. Numerous methods have been proposed to eliminate constant activity species in reactive transport calculations (Kräutle, 2011; De Simoni et al., 2005; Kräutle and Knabner, 2005; Molins et al., 2004; Saaltink et al., 1998). Regardless of the decision to eliminate them, we refer generically to these methods as the traditional speciation methods, as they should all yield the same results.

Being the degrees of freedom for system (3.1) equal to 4, the concentrations of all species can be calculated from four known data: total concentration of calcium, alkalinity, activity of water and pH for example

$$\begin{cases} Ca_{tot} : [Ca^{2+}] - x_1 = 0 \\ Alkalinity : [HCO_3^-] + 2[CO_3^{2-}] + [OH^-] - [H^+] - x_2 = 0 \\ Water\ Activity : \mathrm{a}_{H_2O} - x_3 = 0 \\ pH : -\log \mathrm{a}_{H^+} + x_4 = 0 \end{cases} \tag{3.2}$$

where [ ] represents molal concentration (mol/kgw). x_1, x_2 and x_4 are actual measurements representing Ca_{tot}, $Alkalinity$ and pH, while x_3 is the value of water activity fixed to 1. We term these kind of equations "data equations". These must be solved together with the mass action laws deriving from system (3.1)

$$\mathbf{f}_{MAL} = \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \tag{3.3}$$

where $\mathbf{a}$ is a vector containing the activities of the N_s species, $\mathbf{S}_e$ is a matrix ($N_{re} \times N_s$) with the stoichiometric coefficients of the equilibrium reactions and $\mathbf{k}$ is a vector (N_{re}) of equilibrium constants.

Generalizing the traditional speciation method we can say that $N_1 = N_s - N_{re}$ data equations need to be solved together with $N_2 = N_{re}$ mass action laws, $\mathbf{f}_{MAL}$

$$\begin{cases} \mathbf{g}(\mathbf{c}) - \mathbf{x} = 0 \\ \mathbf{f}_{MAL}(\mathbf{c}) = 0 \end{cases} \tag{3.4}$$

where $\mathbf{c}$ is the vector of concentrations of the N_s species, $\mathbf{x}$ a vector of N_1 data and $\mathbf{g}(\mathbf{c})$ defines operations to be applied to $\mathbf{c}$ in order to compute what is measured (e.g., linear combinations of species concentrations to obtain measured components, or $-\log(\gamma_{H^+} \cdot [H^+])$ to obtain pH, where γ_{H^+} is the proton activity coefficient). Typically data equations contain balances of total concentrations, electrical charge, alkalinity, total dissolved inorganic carbon (TIC), pH values, redox potential or electrical conductivity.

The traditional algorithm to speciate consists of five steps: (1) dividing the species in two sets of $N_1 = N_s - N_{re}$ primary and $N_2 = N_{re}$ secondary species (Steefel and Yabusaki, 2000) with concentrations $\mathbf{c}_1$ and $\mathbf{c}_2$, respectively; (2) guess an initial value of primary concentrations; (3) use $\mathbf{f}_{MAL}$ to calculate $\mathbf{c}_2 = f(\mathbf{c}_1)$; (4) use data $\mathbf{x}$ to solve $g(\mathbf{c}_1, \mathbf{c}_2) - \mathbf{x} = 0$ for $\mathbf{c}_1$, (5) repeat steps (3) and (4) until convergence.

This work is focused on cases in which the number of available data is larger than N_1. In this case, the resulting data equations cannot be solved exactly. Instead, they need to aknowledge measurement errors.

For example, if measurements of total dissolved inorganic carbon (TIC) and pressure of gas ($P_{CO_{2(g)}}$) were available and we wanted to apply zero charge balance and equilibrium with calcite as well, the data equations could be rewritten as

$$\begin{cases} Ca_{tot} : [Ca^{2+}] - x_1 = \varepsilon_1 \\ Alkalinity : [HCO_3^-] + 2[CO_3^{2-}] + [OH^-] - [H^+] - x_2 = \varepsilon_2 \\ Water\ Activity : \mathrm{a}_{H_2O} - x_3 = \varepsilon_3 \\ pH : -\log \mathrm{a}_{H^+} + x_4 = \varepsilon_4 \\ TIC : [CO_{2(aq)}] + [HCO_3^-] + [CO_3^{2-}] - x_5 = \varepsilon_5 \\ P_{CO_{2(g)}} : \log \mathrm{a}_{CO_{2(g)}} - x_6 = \varepsilon_6 \\ Charge\ Balance : [H^+] + 2[Ca^{2+}] - [OH^-] - [HCO_3^-] - 2[CO_3^{2-}] = \varepsilon_7 \\ Calcite\ Eq. : \log \mathrm{a}_{\mathrm{Ca}^{2+}} + \log \mathrm{a}_{\mathrm{HCO}_3^-} - \log \mathrm{a}_{\mathrm{H}^+} - \log K_5 = \varepsilon_8 \end{cases} \tag{3.5}$$

where x_5 is the measured TIC, x_6 is $\log(P_{CO_{2(g)}})$ and x_7 and x_8 are equal to zero because of the zero charge balance and equilibrium constraints (x_7 corresponds to the saturation index of calcite, null at equilibrium). ε_i, $i = 1, ..., 8$, represent measurement errors that need to be taken in account since the system to be solved has become overdetermined (i.e., the number of data is larger than N_1). The data set (3.5) presents 4 redundant data.

The algorithm to solve data equations (3.5) together with mass action laws (3.3) to speciate is explained in the following section.

3.2.2 Speciation with redundant data: Problem statement

If redundant informations are used to solve a speciation problem, system (3.4) can be re-defined as follows

$$\begin{cases} \mathbf{g}(\mathbf{c}) - \mathbf{x} = \boldsymbol{\varepsilon} \\ \mathbf{f}_{MAL}(\mathbf{c}) = 0 \end{cases} \tag{3.6}$$

The differences of system (3.6) from the traditional speciation problem defined in (3.4) are the dimension of $\mathbf{g}$ and $\mathbf{x}$ ($dim(\mathbf{g}) = dim(\mathbf{x}) = N_d > N_1$) and errors in measurements $\boldsymbol{\varepsilon}$ which are included. $\boldsymbol{\varepsilon}$ can incorporate analytical errors in data, such as in data 1 to 4 in system (3.5), and uncertainty about the correct model, such as charge balance and equilibrium with calcite assumptions in system (3.5).

When solving speciation problems, it is common to use data equations which are either linear combinations of concentrations (e.g., TIC, alkalinity) or linear combinations of log-activities (e.g., equilibrium with minerals or gases). Moreover, the errors ($\boldsymbol{\varepsilon}$) of both types of data equations can have a normal or log-normal distribution. Therefore, the expressions of $\mathbf{g}(\mathbf{c})$ must be defined and calculated accordingly to the types of data equations (see Appendix A for details).

System (3.6) is overdetermined, therefore a non-linear least square fitting is required to minimize $\boldsymbol{\varepsilon}$, as described below.

3.2.3 Algorithm

We want to find the solution of (3.6) that minimizes the sum S of the weighted squares of the difference between measured and calculated data, defined as

$$S = \boldsymbol{\varepsilon}^t \mathbf{V}^{-1} \boldsymbol{\varepsilon} \tag{3.7}$$

where $\mathbf{V}$ is the covariance matrix ($N_d \times N_d$) of measurement errors. Without loss of generality, we will assume errors to be not correlated, so that $\mathbf{V}$ is a diagonal matrix, containing the variance of each i-th measurement, $\sigma^2_{e,i}$. The condition leading to the minimum value of S is that its derivative with respect to the unknowns, $\ln \mathbf{c}_1$, is zero

$$\frac{\partial S}{\partial \ln \mathbf{c}_1} = 2\boldsymbol{\varepsilon}^t \mathbf{V}^{-1} \frac{\partial \mathbf{g}}{\partial \ln \mathbf{c}_1} = 2\boldsymbol{\varepsilon}^t \mathbf{V}^{-1}\mathbf{J} = \mathbf{0} \tag{3.8}$$

where $\mathbf{J}$ is the jacobian matrix containing the derivatives of $\mathbf{g}(\mathbf{c})$ with respect to the unknowns, whose expression is derived and explained in Appendix B. We decided to work with $\ln \mathbf{c}_1$ as variable but it would be equally possible to express all the equations as function of $\mathbf{c}_1$.

Approximating the function $\boldsymbol{\varepsilon}$ linearly between two sequential iterations k and $k+1$

$$\boldsymbol{\varepsilon}^{k+1} = \boldsymbol{\varepsilon}^k + \mathbf{J}^k \Delta \ln \mathbf{c}_1^k \tag{3.9}$$

with

$$\Delta \ln \mathbf{c}_1^k = \ln \mathbf{c}_1^{k+1} - \ln \mathbf{c}_1^k \tag{3.10}$$

and substituting it in (3.8), the solution for a given iteration k is

$$\mathbf{J}^t \mathbf{V}^{-1} \mathbf{J} \Delta \ln \mathbf{c}_1 = -\mathbf{J}^t \mathbf{V}^{-1} \boldsymbol{\varepsilon} \tag{3.11}$$

After convergence, the covariance matrices of the estimation errors associated to $\ln \mathbf{c}_1$ ($\boldsymbol{\Sigma}_1$) and $\ln \mathbf{c}_2$ ($\boldsymbol{\Sigma}_2$) can be calculated. This can be useful to analyze the quality of the estimation (see Section 3.2.4). $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$, in fact, provide the uncertainty associated to the estimation of $\ln \mathbf{c}_1$ and $\ln \mathbf{c}_2$. $\boldsymbol{\Sigma}_1$ ($N_1 \times N_1$) can be calculated by means of the "real" covariance matrix of the data, $\mathbf{C}_d$ ($N_d \times N_d$)

$$\boldsymbol{\Sigma}_1 = Cov(\ln \mathbf{c}_1) = (\mathbf{J}^t \mathbf{C}_d^{-1} \mathbf{J})^{-1} \tag{3.12}$$

However, in reality $\mathbf{C}_d$ is not known. Therefore, it is necessary to make an hypothesis about its structure. A simple and reasonable assumption consists of assuming that the relative values of error variances, expressed by $\mathbf{V}$, are more reliable than their absolute values. This can be quantified by an unknown factor σ^2, so that the covariance matrix reads

$$\mathbf{C}_d = \sigma^2 \mathbf{V} \tag{3.13}$$

where σ^2 can be approximated as S/N_d (S was defined in equation 3.7). Substituting (3.13) into (3.12) we obtain

$$\boldsymbol{\Sigma}_1 = \sigma^2 (\mathbf{J}^t \mathbf{V}^{-1} \mathbf{J})^{-1} \tag{3.14}$$

$\boldsymbol{\Sigma}_2$ ($N_2 \times N_2$) can be calculated by taking into account the dependence of $\mathbf{c}_2$ on $\mathbf{c}_1$

$$\boldsymbol{\Sigma}_2 = Cov(\ln \mathbf{c}_2) = \left(\frac{\partial \ln \mathbf{c}_2}{\partial \ln \mathbf{c}_1}\right) \boldsymbol{\Sigma}_1 \left(\frac{\partial \ln \mathbf{c}_2}{\partial \ln \mathbf{c}_1}\right)^t \tag{3.15}$$

Details on the calculation of $(\partial \ln \mathbf{c}_2 / \partial \ln \mathbf{c}_1)$ are explained in Appendix B.

The steps of the proposed algorithm can be outlined as follows:

1. Set $\mathbf{x}$ and matrices of $\mathbf{g}(\mathbf{c})$ (see Appendix A)

2. Guess initial value of $\mathbf{c}_1^0$

3. Given $\mathbf{c}_1^k$, calculate secondary concentrations $\mathbf{c}_2^k = f(\mathbf{c}_1^k)$ and $\partial\mathbf{c}_2^k/\partial\mathbf{c}_1^k$ from $\mathbf{f}_{MAL} = 0$ (see Appendix B)

4. Calculate $\boldsymbol{\varepsilon}^k$, Jacobian matrix $\mathbf{J}^k$, and RHS and LHS of system (3.11). Solve system (3.11) and evaluate $\Delta \ln \mathbf{c}_1^k$

5. Update the solution $\ln \mathbf{c}_1^{k+1} = \ln \mathbf{c}_1^k + \Delta \ln \mathbf{c}_1^k$

6. Set k=k+1 and repeat steps 3. to 5. until convergence

7. After convergence, calculate $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$

As convergence criteria, we check the maximum relative error between two sequential iterations and the residual (RHS in system 3.11) at every iteration. The iterative process is stopped when both quantities are smaller than threshold values defined by the user.

3.2.4 Testing approach

The algorithm was tested by means of two synthetic examples: first a single reaction representing gypsum equilibrium at a temperature of 25 ^{o}C in ideal conditions ($\mathbf{a} = \mathbf{c}$)

$$CaSO_{4(s)} \rightleftharpoons Ca^{2+} + SO_4^{2-} \qquad \log K = -4.482 \tag{3.16}$$

We chose arbitrarily a known solution for Ca^{2+} and SO_4^{2-} and 5 possible measurements with errors of $\log[Ca^{2+}]$ and $\log[SO_4^{2-}]$, decimal logarithm of calcium and sulfate concentrations, respectively. Figure 3.1 shows the equilibrium line and the exact solution of the speciation together with the 5 measurement points that were used in this case.

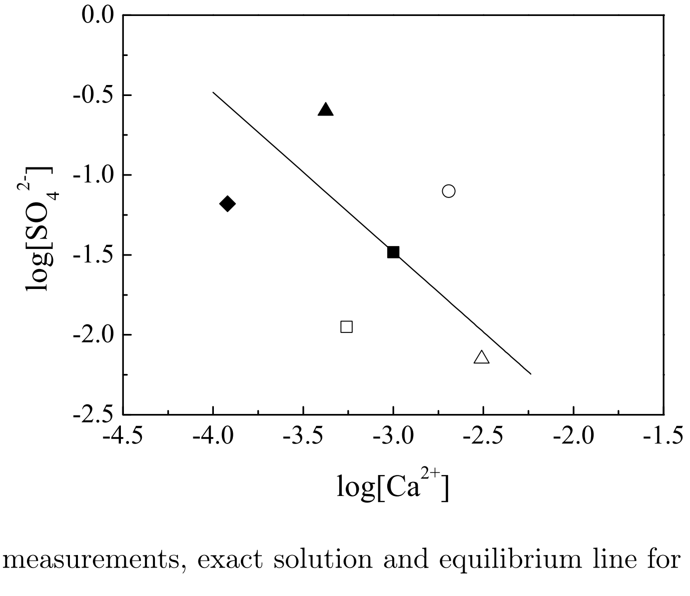


Figure 3.1: Five measurements, exact solution and equilibrium line for gypsum example.

Both the traditional and the proposed methods were applied to solve the speciation. Since the system is characterized by one degree of freedom, i.e. one datum is necessary to solve the speciation, first the traditional method was employed using only $\log[Ca^{2+}]$ data, and later the proposed method was employed using both $\log[Ca^{2+}]$ and $\log[SO_4^{2-}]$ measurements. The results were first compared in terms of logaritmic mean square error, MSE_{log}

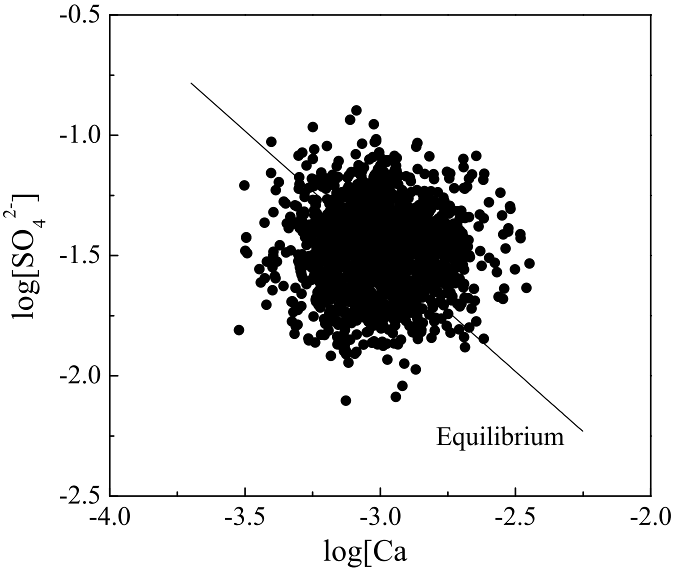


Figure 3.2: 1500 measurement points generated by means of a lognormal distribution for gypsum example.

$$MSE_{log} = \frac{1}{N}\frac{1}{N_s}\sum_{i=1}^{N}\sum_{j=1}^{N_s}\left[\log\left(\frac{c_{ij}}{c_j^*}\right)\right]^2 \tag{3.17}$$

where N is the number of measurements (5), N_s is the number of species (2), c_{ij} is the calculated concentration of the i-th measurement and j-th species, and c_j^* is the exact value of the j-th species. Comparing the two methods in terms of MSE_{log} was possible because the exact solution in this case is known. However, when a speciation is calculated using real data the exact solution is not known a priori. Therefore, we computed the covariance matrices of estimation errors $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ defined in (3.14) and (3.15), respectively, and compared the estimation errors of the traditional and the proposed methods also in terms of the variable Var^*

$$Var^* = \frac{1}{N}\frac{1}{N_s}\sum_{i=1}^{N}\left[\sum_{l=1}^{N_1}\left(\Sigma_{1,ll}\right)_i + \sum_{m=1}^{N_2}\left(\Sigma_{2,mm}\right)_i\right] \tag{3.18}$$

Var^* represents a mean value of the estimation errors variance of all the species. For comparison purposes, we fixed $\sigma^2 = 1$ for the two methods. This way, we guarantee the same real covariance matrix of data, $\mathbf{C}_d$ (see equation 3.13), for both methods.

An input variable of the method is the variance of measurement errors associated to the data, σ_e^2 (see equation 3.7). In this example we used the same value for both log$[Ca^{2+}]$ and log$[SO_4^{2-}]$ data ($\sigma_e = 0.17$), so that they have the same weight. In order to make the results of this example more general we tested again the two methods for 1500 measurements of log$[Ca^{2+}]$ and log$[SO_4^{2-}]$ with $\sigma_e = 0.17$. The exact solution was therefore perturbed 1500 times by means of a lognormal distribution with standard deviation $\sigma_g = 0.17$ to obtain the measurement points shown in figure 3.2. The use of a lognormal distribution allowed us to avoid possible negative values in concentrations in the measurements generation process.

The advantage of this simple problem is that it presents an analytical solution. In the case of using only log$[Ca^{2+}]$ data, the variance of log$[Ca^{2+}]$ error will be σ_e^2 because log$[Ca^{2+}]$ will remain unchanged. The same error will transfer to log$[SO_4^{2-}]$ = $\log K$ - log$[Ca^{2+}]$. Therefore, the expected value of Var^* is σ_e^2. If both log$[Ca^{2+}]$ and log$[SO_4^{2-}]$ measurements are used, it is easy to check that minimizing $\varepsilon^2_{Ca^{2+}} + \varepsilon^2_{SO_4^{2-}}$, where $\varepsilon_i = \log c_i - x_i$ $(i = Ca^{2+}, SO_4^{2-})$, subject to log$[Ca^{2+}]$+log$[SO_4^{2-}]$ = $\log K$, leads to log$[Ca^{2+}] = (\log K + x_{Ca^{2+}} - x_{SO_4^{2-}})/2$. Thus, the variance of both estimates is $\sigma_e^2/2$, which is the expected value of Var^*. This result will serve to test our approach and to illustrate the advantage of using redundant data.

As second example we chose the carbonate system defined in (3.1). The extended Debye-Hückell expression for activity coefficients was used in this example (Helgeson and Kirkham, 1974). As explained in Section 3.2.1 the system is characterized by four degrees of freedom. We used as exact solution a water in equilibrium with calcite, with partial pressure of $CO_{2(g)}$ equal to $10^{-3.5}$, a$_{H_2O} = 1$ and electrically balanced. Its chemical composition is shown in Table 3.1.

Table 3.1: Exact solution for carbonate example.

	Species	Values
c^*	Ca^{2+}	$4.92 \cdot 10^{-4}$
$[mol/l]$	H^+	$5.51 \cdot 10^{-9}$
	HCO_3^-	$9.63 \cdot 10^{-4}$
	$CO_{2(aq)}$	$1.07 \cdot 10^{-5}$
	CO_3^{2-}	$9.78 \cdot 10^{-6}$
	OH^-	$2.01 \cdot 10^{-6}$
a* $[bar]$	$CO_{2(g)}$	$3.16 \cdot 10^{-4}$
SI (Sat. Index)	$Calcite$	0.0

First we compared the traditional and the proposed methods to verify the accuracy of the two algorithms in terms of speciation results. For this purpose, six data were extracted from the exact solution to be used in the speciation calculations (Table 3.2). The traditional speciation method was employed using four data equations: $alkalinity$, Ca_{tot}, a$_{H^+}$ and a$_{H_2O} = 1$. Note that the proton activity presents a measurement error, therefore cannot be eliminated. We will refer to this case as 'solution 1'. Afterwards we tested the proposed method adding gradually redundant data to the previous three: TIC ('solution 2'), activity of the gas, a$_{CO_{2(g)}}$ ('solution 3') and equilibrium with calcite condition ('solution 4'). Afterwards we compared the solution obtained with the traditional speciation method to the solutions obtained using an increasing number of redundant data: from 1 in solution 2 to 3 in solution 4. As for the gypsum example, we perturbed the data to generate 1500 possible measured values and then we compared the speciation results in terms of MSE_{log} and Var^*, defined in (3.17) and (3.18), respectively.

The measured values were generated perturbing the logarithm of the exact value (μ) with a standard deviation, σ_g, of 0.17 by means of a log-normal distribution. The condition of calcite equilibrium was not perturbed ($\sigma_g = 0$), since zero is the reference value of the saturation index for minerals in equilibrium. The values of σ_g were used also to define the uncertainty associated to every datum presenting analytical errors ($\sigma_e = 0.17$, see Table 3.2).

In reality, however, it is difficult to know the correct value of uncertainty for each type of measurement. To analyze the effect of an incorrect measurement error we performed a second group of simulations in which we changed the σ_e values of all data one at a time and calculated

Table 3.2: Mean values and standard deviations adopted to generate 1500 realizations of data for the carbonate example.

Data	μ	σ_g	σ_e	ε equation
Ca_{tot}	$4.92 \cdot 10^{-4}$	0.17	0.17	$(A.3)$
a_{H^+}	$5.29 \cdot 10^{-9}$	0.17	0.17	$(A.4)$
Alkalinity	$9.85 \cdot 10^{-4}$	0.17	0.17	$(A.3)$
TIC	$9.84 \cdot 10^{-4}$	0.17	0.17	$(A.3)$
$\mathrm{a}_{CO_{2(g)}}$	$3.16 \cdot 10^{-4}$	0.17	0.17	$(A.4)$
Calcite Eq.	0	0	0.17	$(A.4)$

Units are in mol/l except for $\mathrm{a}_{CO_{2(g)}}$ which is expressed in bar.

MSE_{log} and Var^* as function of the standard deviation of measurement errors of every constraint. We chose two values of σ_e: the first larger than the one used to generate the perturbed measurements ($\sigma_e = 0.35 > \sigma_g$), to simulate a higher uncertainty associated to the data, and the second smaller ($\sigma_e = 0.09 < \sigma_g$), to simulate more certain data values.

3.3 Results

3.3.1 Gypsum example

Figure 3.3 shows the results of the traditional speciation method, i.e. using only $\log[Ca^{2+}]$ data. It can be noticed that the five points moved on the equilibrium line, since the equilibrium with gypsum was imposed as certain condition, along a line parallel to the y-axis which represents the imposed calcium concentration data. In this case MSE_{log}=0.28 and Var^*=0.154. Note that Var^* coincides with its expected value, σ_e^2, once ln is converted to $\log_{10}$. Afterwards, the proposed method was tested, i.e., using both $\log[Ca^{2+}]$ and $\log[SO_4^{2-}]$ data. The results are shown in figure 3.4. It can be observed that while some of the points moved further from the exact solution with respect to the classical speciation results (white and black triangles), the others moved closer to the exact solution. However, for all the points the proposed algorithm minimizes the distance between measured and calculated data. The calculated mean square error in this case was 0.23, smaller than 0.28 obtained with the traditional method. Moreover, Var^*=0.077, which coincides again with its expected value, $\sigma_e^2/2$.

The same methodology was employed to compare the two methods for the 1500 measurement points of figure 3.2 and the resulting mean square error decreased from 0.029 for the traditional method to 0.016 for the proposed methods. The value of Var^* also decreased by half, confirming its expected value: from 0.156 to 0.078 for traditional and proposed methods, respectively.

3.3.2 Carbonate example

The results of the comparison between the two methods in terms of MSE_{log} are shown in figure 3.5. It can be noticed that the value of the mean square error for the traditional method (solution 1) is barely larger than 0.05, while it is smaller for the solutions using redundant data (solutions 2, 3 and 4). Moreover, increasing the number of redundant data used in the speciation contributes to reduce more the MSE_{log} value: it decreases from 0.04 using only one redundant data (solution 2) to 0.016 using 3 redundant data (solution 4).

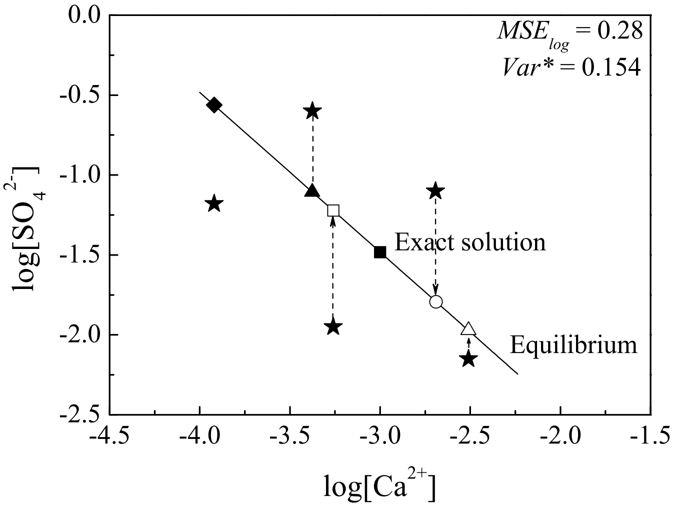


Figure 3.3: Speciation results of traditional speciation method, exact solution and equilibrium line for gypsum example. Dashed arrows show the movement of the five points from initial conditions, represented with stars.

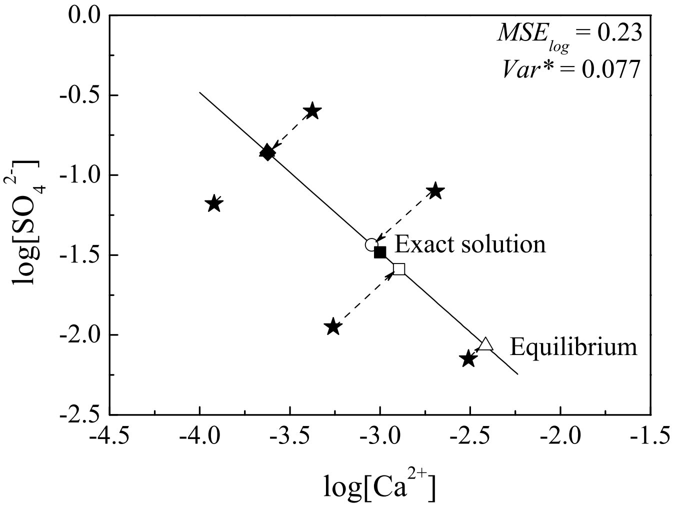


Figure 3.4: Speciation results of proposed method, exact solution and equilibrium line for gypsum example. Dashed arrows show the movement of the five points from initial conditions, represented with stars.

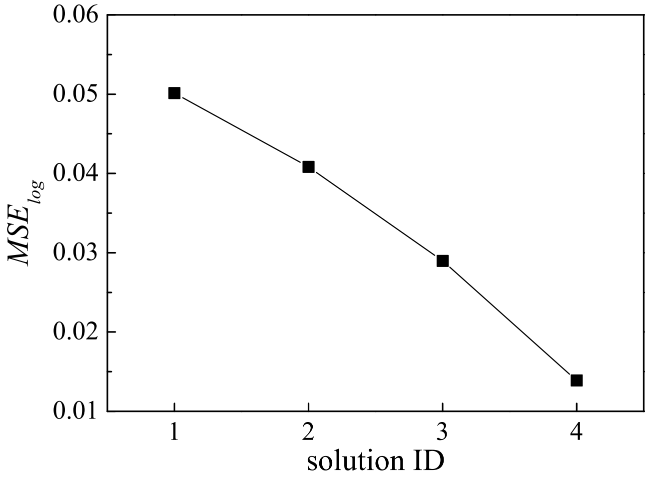


Figure 3.5: MSE_{log} for traditional speciation method (solution 1) and proposed method (solutions 2, 3 and 4) obtained with σ_e values of table 3.2.

Figure 3.6 shows the effect of changing the standard deviation associated to measurements (σ_e). Obviously its value does not affect solution 1, which is the result of a traditional speciation calculation. Neither does the value of σ_e for a$_{CO_{2(g)}}$ have an effect on solution 2 (figure 3.6e) because this solution does not use a$_{CO_{2(g)}}$ data. Nor does the σ_e for the assumption of equilibrium with calcite have an effect on solutions 2 and 3 (figure 3.6f), for the same reason. In general one can observe that the use of an incorrect standard deviation can worsen the solution with respect to the one obtained with the correct standard deviations. However, the quality of the solution in terms of MSE_{log} improves using redundant data with respect to the traditional speciation, despite a wrong choice about σ_e value. Decreasing the uncertainty relative to the equilibrium with calcite assumption (figure 3.6f) improves the solution even with respect to the one obtained with correct σ_e values. This is because the correct value of σ_g for this datum is 0, not 0.17 (see Table 3.2). The effect of σ_e relative to alkalinity and TIC (figure 3.6c, d) are very similar as their values are very close, due to the fact that in this pH range the concentration of HCO_3^- is predominant with respect to carbonate species or OH^- concentrations. Finally, it seems that changing the uncertainty relative to Ca_{tot} (figure 3.6a) does not affect the solution. Nevertheless, when a large number of redundant data is used, such as in solution 4, the standard deviation seems to have a minor effect on the estimation error, MSE_{log}.

The effect of an incorrect value of σ_e on Var^* was also analyzed. Only the results for a$_{H^+}$ and alkalinity are reported in figure 3.7 as the most representative. It can be noticed that the error variance can be big for the traditional speciation method (solution 1), while it slightly decreases when redundant data are used (solutions 2, 3 and 4). Moreover, the more redundant data are used, the more the variance of estimation error decreases, converging to a value corresponding to the true standard deviation of measurement errors (σ_e=0.17).

3.4 Conclusions

We proposed a speciation algorithm that uses redundant data and aknowledges measurement errors, on the assumption that redundant data will reduce estimation errors in geochemical calculations.

We compared the proposed method with the traditional speciation method in terms of logaritmic

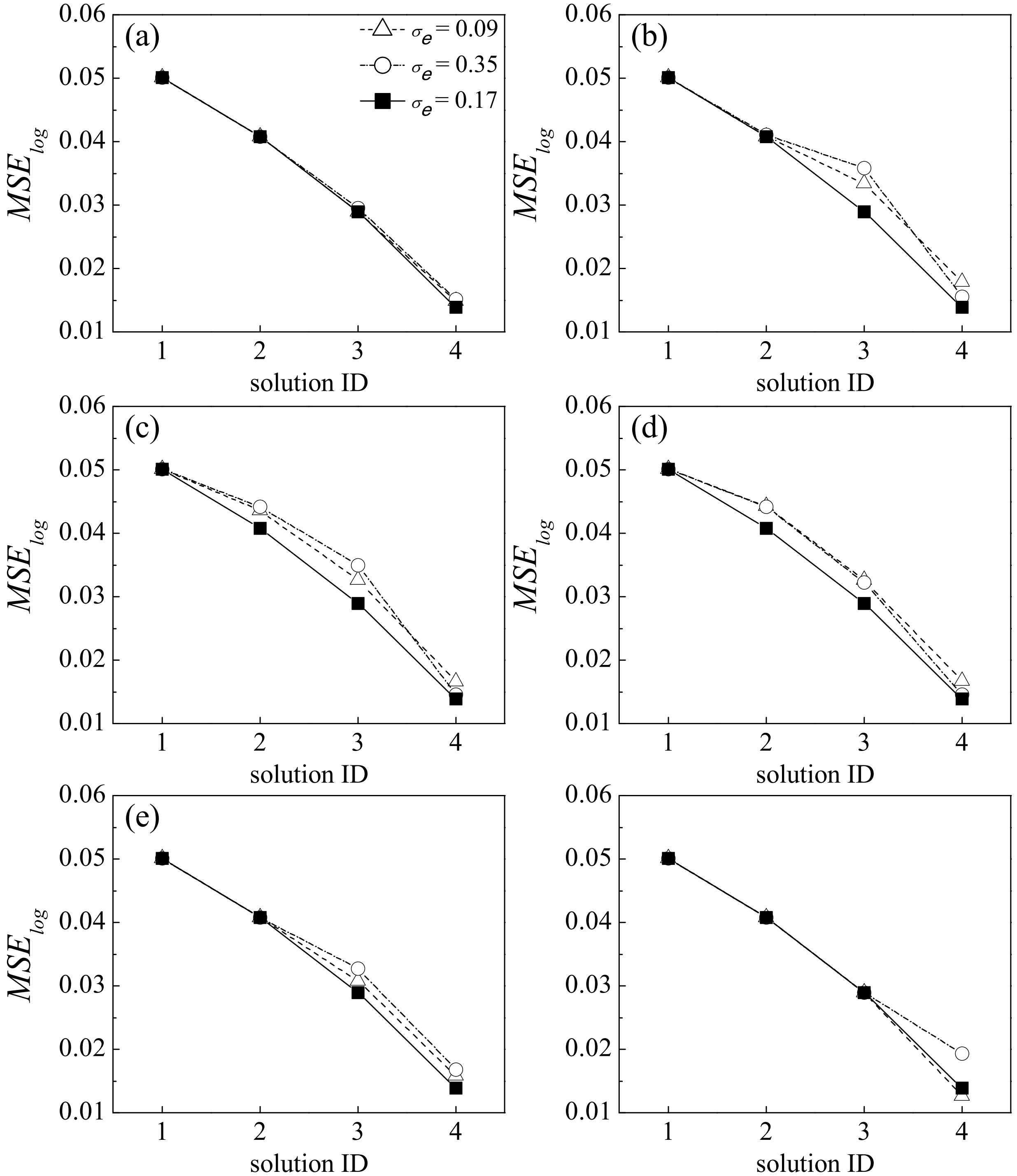


Figure 3.6: MSE_{log} obtained changing values of σ_e of each data: (a) Ca_{tot}; (b) a_{H^+}; (c) alkalinity; (d) TIC; (e) $a_{CO_{2(g)}}$; (f) calcite equilibrium.

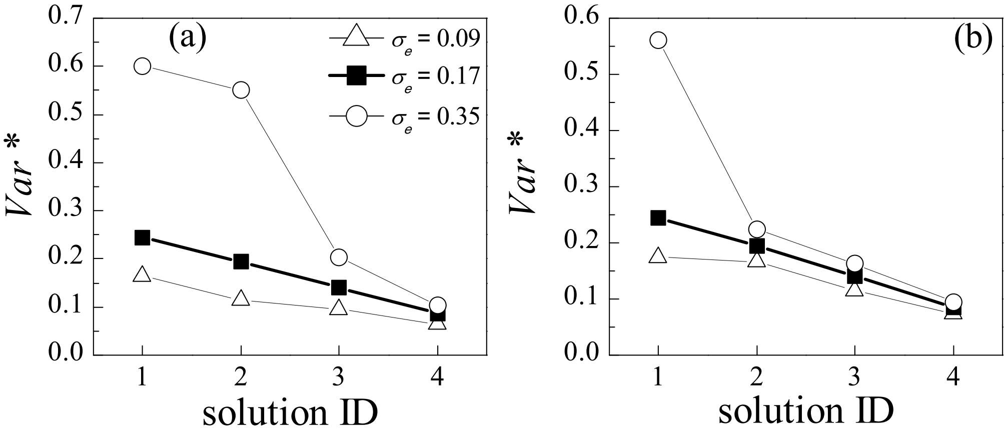


Figure 3.7: Var^* obtained for different values of σ_e of (a) a_{H^+} and (b) alkalinity.

mean square error, MSE_{log} and mean value of estimation error variance, Var^*. We tested both algorithms by means of two synthetic examples. Both MSE_{log} and Var^* using redundant data are consistently smaller than in the traditional method.

The effect of measurement errors was examined in a carbonate system example. The algorithm is sensitive to the variance of measurement errors. Also, a wrong value of these variances can worsen the results with respect to the ones obtained with the correct standard deviation. However, the effect of its value depends on the type of data associated to it. A wrong error associated to measurements can still improve the results in terms of mean square error and variance of estimation error with respect to a traditional speciation method, especially when a large number of redundant data are used.

Therefore we argue that the proposed method can improve the quality of the speciation results, reducing estimation errors.

Chapter 4

An algorithm to calculate mixing ratios and reactions applied to mixing in carbonate coastal aquifers [*]

4.1 Introduction

The mixing problem consists of identifying the proportions in which a number of known end-members are mixed in a sample. Mixing models have been used for decades in many branches of Earth Sciences, including mineralogy (Langmuir et al., 1978), air quality studies (De Gouw et al., 2005) and weathering processes study (Blum et al., 2002). They are also used in a variety of hydrogeological problems, e.g., for quantifying interactions between groundwaters and/or surface waters (Gómez et al., 2014; Morales-Casique, 2012; Kalbus et al., 2006), for determining the proportions of various pollution sources in waste streams (Phillips and Gregg, 2003), for identifying the sources of recharge (Vázquez-Suñé et al., 2010) or hydrochemical and microbiological reactions in urban groundwater (Jurado et al., 2015, 2013; Tubau et al., 2014), or for assessing water-rock interactions (Daniele et al., 2013; Van der Kemp et al., 2000) or seawater intrusion dynamics (Han et al., 2014; Currell and Cartwright, 2011; Saravana Kumar et al., 2009; Bouchaou et al., 2008; Jørgensen et al., 2008; Walraevens et al., 2007) in aquifers.

Traditionally, mixing calculations require the solution of systems containing mass balances of conservative species and a constraint imposing that the mixing proportions of all end-members must sum up to 1. For example, if the concentrations of a species A is known in a sample (c_A) and in two end-members, E-M1 (c_{A1}) and E-M2 (c_{A2}), the proportions of E-M1 and E-M2 in the sample (λ_1 and λ_2, respectively) can be calculated from the following system:

$$\begin{cases} c_A = \lambda_1 c_{A1} + \lambda_2 c_{A2} \\ \lambda_1 + \lambda_2 = 1 \end{cases} \tag{4.1}$$

The mass balance defined in (4.1) implies that the concentration of the sample c_A lies on a line segment ('conservative mixing line') whose end-points are c_{A1} and c_{A2}. If the concentrations of other conservative species (e.g., B) are also available, their mass balance can be added to system (4.1), which would become overdetermined. Ideally, if the data were error-free, c_B would fall on the conservative mixing line (figure 4.1a).

Several methods have been proposed (Sun and Gui, 2014; Valder et al., 2012; Long and Valder, 2011; Nakaya et al., 2007; Rueedi et al., 2005; Parkhurst, 1997) and a number of codes exist for calculating mixing proportions from systems like (4.1), acknowledging errors. In fact, Carrera et al. (2004) proposed a method to estimate not only mixing proportions, but also to reduce end-members uncertainty.

---

[*] This Chapter is based on the submitted paper De Gaspari et al. (submitted)

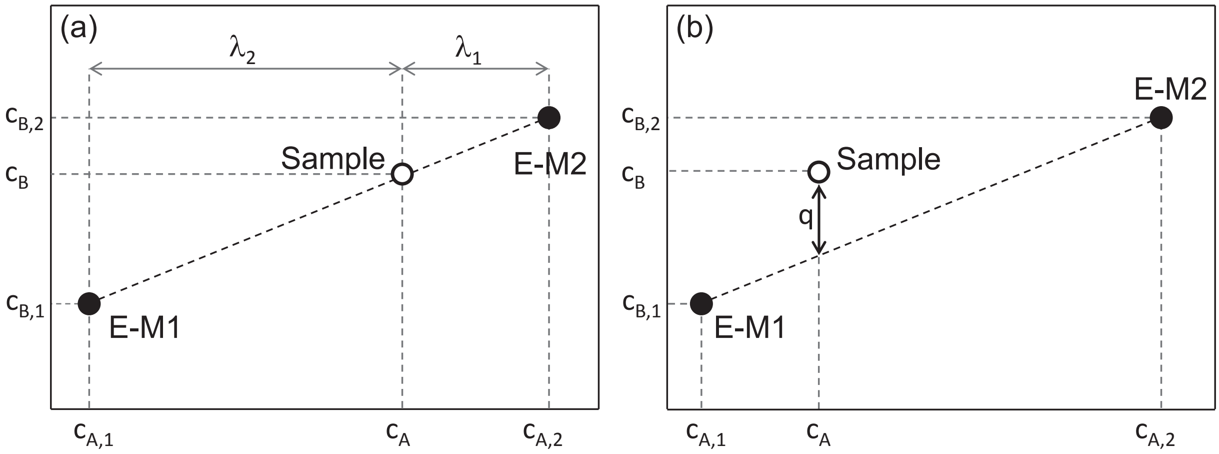


Figure 4.1: Illustration of (a) conservative mixing and (b) reactive mixing.

Lately it has become common to use also reactive species in mixing models. The mass balance defined in (4.1) can be extended to reactive species by including a source term (q) due to chemical reactions:

$$\begin{cases} c_A = \lambda_1 c_{A1} + \lambda_2 c_{A2} \\ c_B = \lambda_1 c_{B1} + \lambda_2 c_{B2} + q \\ \lambda_1 + \lambda_2 = 1 \end{cases} \tag{4.2}$$

q is the quantity of a reaction contributing to the production/depletion of the species B per unit volume of solution. As shown in figure 4.1b, system (4.2) expresses that the concentration c_B of the reactive species departs from the conservative mixing line (Pitkänen et al., 1999; Skov et al., 1997). Note that the unknowns of system (4.2) are the mixing ratios and the reacted quantity q. The use of reactive species concentrations is convenient in mixing calculations because conservative species are usually only a few, and also because the use of more data helps to improve the robustness of the solution (Carrera et al., 2004).

The most used codes for reactive mixing calculations are Netpath (Plummer et al., 1991), PHREEQC (Parkhurst et al., 1999) and M3 (Laaksoharju et al., 1999). Netpath and PHREEQC solve systems analogous to (4.2), but PHREEQC also accounts for uncertainty in the data to represent analytical or sampling errors. Uncertainty in the data is included by means of inequality constraints that allow each datum to vary within specific limits defined by the user. Both Netpath and PHREEQC need one mole-balance equation for each reacting phase present in the system. M3 is a useful code to solve mixing problems when a large data set is available for numerous samples (Gómez et al., 2008; Laaksoharju et al., 2008). In fact, it uses EMMA (End-Member Mixing Analysis, Christophersen and Hooper 1992) based on Principal Component Analysis (PCA) to identify the potential extreme waters before calculating the mixing proportions and reacted quantities.

These codes are useful in helping geochemists to analyze hydrochemical data and to gain insights into geochemical processes. However, they prevent the user from imposing equilibrium conditions on the mixture relative to phases that could react. Equilibrium conditions with minerals or gases can not be imposed on the mixture. This can be a limiting factor if, for example, gas pressure had been measured (e.g., Garing et al. 2013) or some assumptions about equilibrium minerals could be done in a mixing zone, such as the deep aquifer which is the subject of this work.

The objective of this work is to present a reactive mixing method that allows specifying equilibrium constraints, while acknowledging errors in the input data.

We applied the method to a set of geochemical data sampled by Sanz (2007) at a brackish spring, named 'S'Almadrava', in Mallorca island (Spain) which is the result of reactive mixing between fresh groundwater and saltwater intruding into a deep carbonate aquifer. Mixing calculations, in fact, have been extensively used for assessing geochemical processes occurring in the freshwater-saltwater mixing zone of different coastal carbonate aquifers in the world (Han et al., 2014; Panteleit et al., 2011; Walraevens and Van Camp, 2005; Lakshmanan et al., 2003; Vengosh et al., 2005; Jones et al., 1999, amongst others) and in Mallorca (Garing et al., 2013; Price and Herman, 1991; Herman et al., 1985). This zone has been demonstrated to be subject to complex geochemical mixing-driven processes such as: (i) carbonate dissolution/precipitation (Sanz et al., 2011; Rezaei et al., 2005; Corbella et al., 2003); (ii) ion exchange (Walraevens et al., 2007; Appelo, 1994; Howard and Lloyd, 1983); (iii) redox reactions (Panteleit et al., 2011; Slomp and Van Cappellen, 2004; Duncan and Shaw, 2003).

4.2 Methodology

4.2.1 Problem statement

Let us consider a water sample as a result of reactive mixing between N_e end-members and the interaction with the aquifer rock. In order to apply the reactive mixing method, the participating species (N_s) as well as equilibrium (N_{re}) and kinetic (N_{rk}) reactions that constitute the chemical system need to be defined. This is accomplished on the basis of the conceptual model of the modeler and the available data about the mixture. The objective is to calculate the proportions of each end-member in the mixture and the quantity of the reacted phases explaining the chemical composition of the sample.

Data are likely to contain errors. Therefore, we formulate the problem as the minimization of the following objective function, which expresses the error between sampled and calculated data:

$$F = (\mathbf{Ac} - \mathbf{a}^*)^t \mathbf{V}^{-1} (\mathbf{Ac} - \mathbf{a}^*) = min \tag{4.3}$$

where $\mathbf{c}$ is a vector (N_s) containing the unknown concentration of the aqueous species of the sample, H_2O excluded, $\mathbf{a}^*$ is a vector (N_d) containing the measured data and $\mathbf{A}$ is a matrix ($N_d \times N_s$) of coefficients such that, if muyltiplied by $\mathbf{c}$, it gives the calculated data. The matrix $\mathbf{V}$ ($N_d \times N_d$) contains the the covariance of measurement errors. Without loss of generality, we assume the data as not correlated and therefore $\mathbf{V}$ as a diagonal matrix containing the variance of each measurement error, σ_i^2, $i = 1, ..., N_d$.

The concentration vector $\mathbf{c}$ can be obtained from the mass balance defined in system (4.2), generalised in matrix form for N_s species:

$$\mathbf{c} = \mathbf{X}_c \boldsymbol{\lambda} + \mathbf{S}^t \mathbf{q} \tag{4.4}$$

where $\mathbf{X}_c$ is a matrix ($N_s \times N_e$) containing the concentration of the species in every end-member, $\boldsymbol{\lambda}$ is a vector containing the mixing proportions λ_m of every end-member contributing to the sample ($m = 1, ..., N_e$), $\mathbf{S}$ ($N_r \times N_s$) is the stoichiometric matrix and the vector $\mathbf{q}$ (N_r) contains the quantity of consumed/produced reactions. N_r represents the number of reactions. In evaluating $\mathbf{q}$, it is convenient to distinguish between equilibrium reactions (N_{re}) and kinetic reactions (N_{rk}):

$$\mathbf{c} = \mathbf{X}_c\boldsymbol{\lambda} + \mathbf{S}_e^t\mathbf{q}_e + \mathbf{S}_k^t\mathbf{q}_k \tag{4.5}$$

Being $\mathbf{S}_e$ $(N_{re} \times N_s)$ the part of the stoichiometric matrix relative to equilibrium reactions and $\mathbf{S}_k$ $(N_{rk} \times N_s)$ the part of the stoichiometric matrix relative to kinetic reactions. Vectors $\mathbf{q}_e$ and $\mathbf{q}_k$, of dimensions N_{re} and N_{rk}, contain the quantity of reacted species due to equilibrium and kinetic reactions, respectively.

Moreover, equilibrium reactions must meet the mass action law:

$$\mathbf{S}_e(\log \mathbf{c} + \log \boldsymbol{\gamma}) = \log \mathbf{K} \tag{4.6}$$

where $\boldsymbol{\gamma}$ is a vector (N_s) containing the activity coefficients for every species and $\mathbf{K}$ is a vector (N_{re}) of equilibrium constants.

In system (4.6), and therefore $\mathbf{q}_e$, are included homogeneous aqueous reactions and heterogeneous reactions that involve equilibrium phases (i.e., minerals or gases).

Accounting for equilibrium reactions complicates the equations. Therefore, following Saaltink et al. (1998), we eliminate them by multiplying system (4.5) by a component matrix $\mathbf{U}$ of dimensions $(N_c \times N_s)$ such that $\mathbf{U} \cdot \mathbf{S}_e^t = 0$. This allows to change from concentrations to components $(\mathbf{u})$:

$$\mathbf{u} = \mathbf{X}_u\boldsymbol{\lambda} + \mathbf{U}\mathbf{S}_k^t\mathbf{q}_k \tag{4.7}$$

where $\mathbf{u} = \mathbf{U} \cdot \mathbf{c}$, $\mathbf{X}_u = \mathbf{U} \cdot \mathbf{X}_c$ and $N_c = N_s - N_{re}$ is the number of components. Working with components instead of concentrations allows us to reduce the number of unknowns, eliminating the vector $\mathbf{q}_e$.

The objective of the proposed mixing algorithm is to minimize (4.3) and impose the constraint that the mixing proportions must sum 1

$$\mathbf{1}_{N_e}^t \cdot \boldsymbol{\lambda} = 1 \tag{4.8}$$

For this purpose, we build the following Lagrangian function

$$\mathcal{L} = F + \mu \left(\mathbf{1}_{N_e}^t \boldsymbol{\lambda} - 1\right) \tag{4.9}$$

where μ is an unknown Lagrange multiplier and $\mathbf{1}_{N_e}$ is a vector of 1 of dimension N_e.

4.2.2 Minimization algorithm

For (4.9) to be minimum, the derivatives of $\mathcal{L}$ with respect to the unknowns needs to be equal to zero

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = \frac{\partial \mathcal{L}}{\partial \mathbf{q}_k} = \frac{\partial \mathcal{L}}{\partial \mu} = \mathbf{0} \tag{4.10}$$

which leads to

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = \dfrac{\partial F}{\partial \boldsymbol{\lambda}} + \mu \mathbf{1}_{N_e}^t = 2\left(\mathbf{A}\mathbf{c} - \mathbf{a}^*\right)^t \mathbf{V}^{-1}\mathbf{A}\mathbf{J}_\lambda + \mu \mathbf{1}_{N_e}^t = \mathbf{0} \\ \dfrac{\partial \mathcal{L}}{\partial \mathbf{q}_k} = \dfrac{\partial F}{\partial \mathbf{q}_k} = 2\left(\mathbf{A}\mathbf{c} - \mathbf{a}^*\right)^t \mathbf{V}^{-1}\mathbf{A}\mathbf{J}_{qk} = \mathbf{0} \\ \dfrac{\partial \mathcal{L}}{\partial \mu} = \left(\mathbf{1}_{N_e}^t \boldsymbol{\lambda} - 1\right) = 0 \end{cases} \tag{4.11}$$

$\mathbf{J}_\lambda$ ($N_s \times N_{re}$) and $\mathbf{J}_{qk}$ ($N_s \times N_{rk}$) are the following Jacobian matrices

$$\mathbf{J}_\lambda = \frac{\partial \mathbf{c}}{\partial \boldsymbol{\lambda}} = \frac{\partial \mathbf{c}}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{u}}{\partial \boldsymbol{\lambda}} = \frac{\partial \mathbf{c}}{\partial \mathbf{u}} \mathbf{X}_u \tag{4.12}$$

$$\mathbf{J}_{qk} = \frac{\partial \mathbf{c}}{\partial \mathbf{q}_k} = \frac{\partial \mathbf{c}}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{u}}{\partial \mathbf{q}_k} = \frac{\partial \mathbf{c}}{\partial \mathbf{u}} \mathbf{U}\mathbf{S}_\mathbf{k}^\mathbf{t} \tag{4.13}$$

The derivatives $\partial \mathbf{c}/\partial \mathbf{u}$ can be calculated from the results of a standard speciation calculation (De Gaspari et al., 2015; Paz-García et al., 2013; Bea et al., 2009; Parkhurst et al., 1999, amongst others), i.e., after calculating the concentrations $\mathbf{c}$ from the total concentrations of components ($\mathbf{u} = \mathbf{U} \cdot \mathbf{c}$) and mass action laws (see Appendix A for details). As these equations are non linear, we also need to linearize $\mathbf{c}$ with respect to the unknowns

$$\mathbf{c} = \mathbf{c}^i + \frac{\partial \mathbf{c}^i}{\partial \boldsymbol{\lambda}} (\boldsymbol{\lambda}^{i+1} - \boldsymbol{\lambda}^i) + \frac{\partial \mathbf{c}^i}{\partial \mathbf{q}_k} (\mathbf{q}_k^{i+1} - \mathbf{q}_k^i) \tag{4.14}$$

where i is the iteration number. Substituting (4.14) into (4.11) we obtain the following system that needs to be solved iteratively in order to find a solution

$$\begin{pmatrix} (\mathbf{A}\mathbf{J}_\lambda)^t \mathbf{V}^{-1} (\mathbf{A}\mathbf{J}_\lambda) & (\mathbf{A}\mathbf{J}_\lambda)^t \mathbf{V}^{-1} (\mathbf{A}\mathbf{J}_{qk}) & 1 \\ (\mathbf{A}\mathbf{J}_{qk})^t \mathbf{V}^{-1} (\mathbf{A}\mathbf{J}_\lambda) & (\mathbf{A}\mathbf{J}_{qk})^t \mathbf{V}^{-1} (\mathbf{A}\mathbf{J}_{qk}) & 0 \\ \mathbf{1}_{N_e}^t & 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{\lambda}^i \\ \Delta \mathbf{q}_k^i \\ \Delta \mu^i \end{pmatrix} = - \begin{pmatrix} (\mathbf{A}\mathbf{J}_\lambda)^t \mathbf{V}^{-1} \boldsymbol{\varepsilon}^i \\ (\mathbf{A}\mathbf{J}_{qk})^t \mathbf{V}^{-1} \boldsymbol{\varepsilon}^i \\ 1 - \mathbf{1}_{N_e}^t \boldsymbol{\lambda}^i \end{pmatrix} \tag{4.15}$$

where $\Delta \boldsymbol{\lambda}^i = \boldsymbol{\lambda}^{i+1} - \boldsymbol{\lambda}^i$, $\Delta \mathbf{q}_k^i = \mathbf{q}_k^{i+1} - \mathbf{q}_k^i$ and $\Delta \mu^i = \mu^{i+1} - \mu^i$. $\boldsymbol{\varepsilon}^i$ is a vector of dimensions N_d containing the estimation errors ($\mathbf{A}\mathbf{c}^i - \mathbf{a}^*$) at every iteration, i.e., the difference between calculated and measured data. Moreover, we can calculate $\mathbf{q}_e$ at the end of the iterative process by substituting $\mathbf{c}$ into Equation (4.5).

It must be noted that to solve the optimization problem (4.9) a number of data equal to or larger than the number of unknowns ($N_d \geq N_e + N_{rk} - 1$) is required.

The steps of the algorithm can be outlined as follows:

1. Initialize $\mathbf{X}_u$, $\mathbf{A}$, $\mathbf{V}$ and $\mathbf{a}^*$
2. Guess initial values for $\boldsymbol{\lambda}^0$, $\mathbf{q}_k^0$ and μ^0
3. Calculate $\mathbf{u}^i$ from (4.7) and speciate the sample to calculate $\mathbf{c}^i$ and $\partial \mathbf{c}^i/\partial \mathbf{u}^i$
4. Calculate $\mathbf{J}_\lambda^i$, $\mathbf{J}_{qk}^i$ and $\boldsymbol{\varepsilon}^i$
5. Solve system (4.15) to evaluate $\Delta \boldsymbol{\lambda}^i$, $\Delta \mathbf{q}_k^i$ and $\Delta \mu^i$
6. Update the solution $\boldsymbol{\lambda}^{i+1} = \Delta \boldsymbol{\lambda}^i + \boldsymbol{\lambda}^i$, $\mathbf{q}_k^{i+1} = \Delta \mathbf{q}_k^i + \mathbf{q}_k^i$ and $\mu^{i+1} = \Delta \mu^i + \mu^i$
7. Set $i = i + 1$ and repeat steps 3. to 6. until convergence
8. Once $\boldsymbol{\lambda}$ and $\mathbf{q}_k$ are known, calculate $\mathbf{q}_e$ from Equation (4.5)

As convergence criteria the maximum relative error of the unknowns between two sequential iterations is checked. Alternatively, the iterative loop is interrupted if the objective function (4.3) is constant for ten consecutive iterations.

4.3 Application to a mixing zone of a coastal carbonate aquifer in Mallorca, Spain: S'Almadrava spring

S'Almadrava spring is located at an elevation of 8 m.a.s.l. in the Pollença plain, in the northeastern area of Mallorca island between the Serra Tramuntana mountains and the Pollença bay. The geology of the system has been explained by Sanz (2007). The spring area is fed by a deep permeable aquifer formed by fractured calcites, dolomites and carbonate breccias. Clay minerals have been identified in deep cores drilled in proximity of the spring.

Rainfall data and flow discharge measurements of the spring (Sanz, 2007, Chapter 3) show that the response of S'Almadrava spring to rainfall events is very fast, which reflects the karstic nature of the system. Sanz (2007) proposed a dual permeability model in which freshwater and seawater mix at a depth of approximately 540 m.b.s.l. and the mixed water flows to the spring mouth as a result of two flow regimes: a fast flow through karst conduits and a slow flow through the fractured matrix of the aquifer.

The chemical composition of the spring water results not only from mixing between fresh groundwater and seawater, but also from the water-rock interaction processes occurring in the deep carbonate formations. To characterize the discharged water, Sanz (2007) performed a high-frequency sampling campaign during a period of approximately 40 days between November and December 2004, covering two rainfall events that caused the spring reactivation. Samples were collected at the spring mouth every 1-8 hours. Electrical conductivity (EC), pH, alkalinity and total concentrations of major elements were measured. 110 samples were taken, but only 23 were selected for analysis to cover the whole range of measured EC values. Results of the analyses are listed in Table 4.1 and shown in figure 4.2 as a function of time. Table 4.1 reports also the analysis of the local seawater, which was collected at the Pollença bay (at 0 m.a.s.l.). Saturation indices for calcite (SI_{Cc}), magnesite (SI_{Mg}) and $\log \mathrm{P}_{\mathrm{CO}_2}$ corresponding to the measured values were calculated and are listed in Table 4.1 as well. It can be noticed that all samples are oversaturated with respect to both calcite and magnesite. Moreover, the values of $\log \mathrm{P}_{\mathrm{CO}_2}$ are very close to the CO_2 atmospheric pressure (-3.5 bar). Both EC (figure 4.2a) and the total concentrations of the major elements (figure 4.2c, d) display two peaks, corresponding to the reactivation of the spring after two rainfall events, that generally coincide with two minimum values of pH (figure 4.2b).

The aim of this example is two-fold. First we want to verify the ability of the algorithm to calculate the mixing proportions of freshwater and seawater by reproducing the measuremens of Sanz (2007). Then we want to demonstrate that the code allows to identify and quantify the processes leading to the chemical composition of the water discharged by the spring.

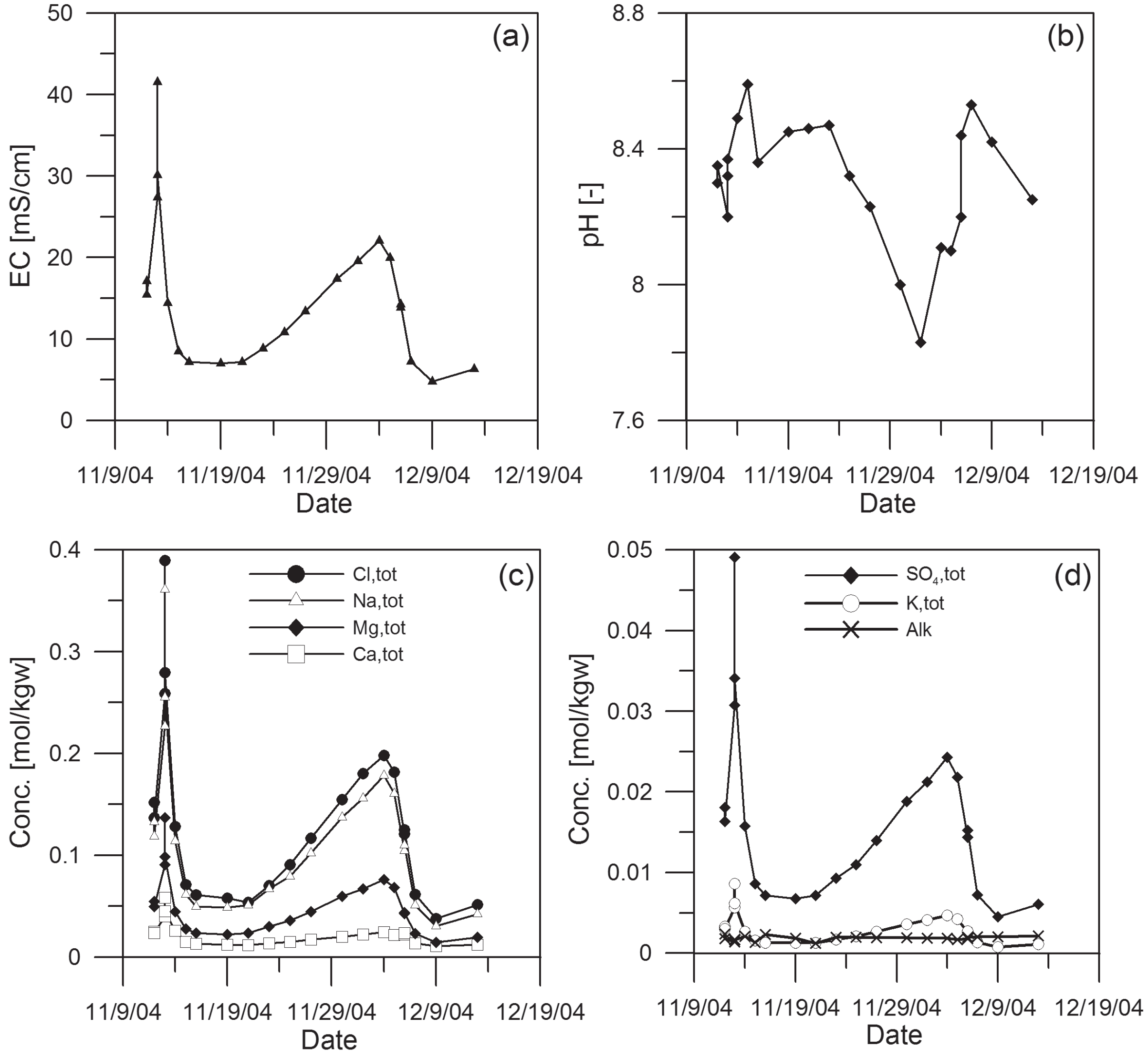


Figure 4.2: Results of chemical analyses relative to S'Almadrava spring as a function of time, performed by Sanz (2007): (a) Electrical Conductivity; (b) pH; (c,d) Total concentrations of major elements.

Table 4.1: Measured values in S'Almadrava spring (Sanz, 2007), corresponding saturation indices for calcite (SI_{Cc}) and magnesite (SI_{Mg}) and logarithm of P_{CO_2} in bar.

Sample	Date	EC	pH	Alkalinity	Ca, tot	Cl, tot	Na, tot	K, tot	Mg, tot	Sr, tot	SO_4, tot	SI_{Cc}	SI_{Mg}	$\log P_{CO_2}$
ID	(mm/dd/yy hh : mm)	(mS/cm)	(−)	(ppm $CaCO_3$)				(mmol/kgw)				(−)	(−)	(bar)
M − 1	11/12/04 14 : 10	17.10	8.3	236.4	25.24	151.9	133.0	3.29	54.44	0.12	18.06	1.43	1.41	-3.37
M − 2	11/12/04 22 : 10	15.43	8.35	183.4	23.46	136.8	118.7	3.00	49.74	0.11	16.32	1.35	1.31	-3.53
M − 3	11/13/04 02 : 10	27.40	8.20	139.9	40.04	259.0	226.9	5.68	90.76	0.2	30.76	1.23	1.25	-3.51
M − 4	11/13/04 16 : 10	41.53	8.32	168.9	58.46	389.6	361.1	8.60	137.0	0.296	49.08	1.50	1.54	-3.61
M − 5	11/13/04 23 : 10	30.11	8.37	164.1	45.1	279.2	255.3	6.18	98.64	0.216	34.08	1.46	1.47	-3.66
M − 6	11/14/04 13 : 10	14.45	8.49	207.5	25.92	128.3	113.9	2.70	44.84	0.116	15.72	1.54	1.42	-3.66
M − 7	11/15/04 03 : 10	8.52	8.59	135.1	14.96	70.99	61.69	1.58	27.64	0.062	8.62	1.23	1.13	-3.93
M − 8	11/16/04 07 : 10	7.20	8.36	231.6	13.14	61.26	49.49	1.30	23.74	0.05	7.18	1.34	1.21	-3.39
M − 9	11/19/04 11 : 30	7.03	8.45	188.2	12.24	58.03	48.68	1.28	21.98	0.046	6.74	1.31	1.18	-3.58
M − 10	11/21/04 11 : 30	7.18	8.46	120.6	11.8	53.92	51.11	1.30	23.36	0.046	7.14	1.11	1.02	-3.79
M − 11	11/23/04 19 : 30	8.86	8.47	197.8	13.52	70.11	66.85	1.70	29.94	0.058	9.28	1.35	1.31	-3.60
M − 12	11/25/04 16 : 52	10.82	8.32	197.8	15.0	90.81	79.54	2.10	35.72	0.07	10.98	1.24	1.24	-3.43
M − 13	11/27/04 17 : 52	13.38	8.23	193.0	17.28	116.9	102.0	2.68	44.82	0.086	13.94	1.18	1.22	-3.35
M − 14	11/30/04 08 : 52	17.39	8.00	193.0	19.94	154.8	137.3	3.62	59.80	0.112	18.78	0.99	1.11	-3.10
M − 15	12/02/04 09 : 52	19.55	7.83	189.1	22.04	180.2	156.2	4.09	67.06	0.126	21.20	0.85	0.98	-2.93
M − 16	12/04/04 17 : 52	22.05	8.11	188.2	24.52	198.2	178.4	4.68	76.32	0.144	24.26	1.13	1.27	-3.25
M − 17	12/05/04 00 : 52	19.96	8.10	1.69.8	21.82	181.4	160.8	4.25	68.18	0.0128	21.82	1.04	1.18	-3.28
M − 18	12/06/04 04 : 52	13.86	8.20	177.6	22.18	121.0	104.3	2.63	42.66	0.098	14.36	1.21	1.12	-3.36
M − 19	12/06/04 18 : 52	14.27	8.44	197.8	23.56	125.2	110.1	2.73	43.56	0.102	15.24	1.46	1.36	-3.61
M − 20	12/07/04 08 : 52	7.26	8.53	204.6	13.64	61.33	51.0	1.33	22.98	0.052	7.20	1.44	1.28	-3.65
M − 21	12/09/04 11 : 00	3.84	8.30	222.0	10.4	30.11	24.01	0.62	12.26	0.028	3.76	1.26	0.94	-3.32
M − 22	21/13/04 12 : 30	4.78	8.42	204.6	10.84	37.66	30.32	0.8	14.56	0.03	4.50	1.32	1.06	-3.50
M − 23	12/22/04 07 : 50	6.32	8.25	212.3	12.32	51.46	42.49	1.11	19.76	0.042	6.08	1.21	1.02	-3.30
Local sea	November 2004	53.65	7.95	127.4	44.1	506.2	459.2	11.42	206.9	0.352	62.34	0.96	1.31	-3.27

4.3.1 Conceptual model

To apply the proposed mixing algorithm, samples M-21 and Local seawater were selected from Table 4.1 to represent freshwater and seawater end-members, respectively, based on their values of Cl concentration (lowest for M-21, highest for Local seawater). The remaining waters of Table 4.1 were considered as mixtures.

Measured values of total concentrations of Ca, Mg, Na and alkalinity are presented in figure 4.3 as a function of chloride concentration, considered to be conservative in the system. Dashed lines represent conservative mixing between the two end-members. Departures from the conservative mixing line are attributed to chemical reactions. It can be noticed that the measured values of Na (figure 4.3c) lay on the mixing line, meaning that Na can be considered conservative in the system. On the contrary, Ca, Mg and alkalinity present deviations from the conservative mixing line, suggesting that certain chemical reactions occurred and affected their values in the mixing process. Some samples are characterized by an excess of Ca (figure 4.3a) and a depletion of Mg (figure 4.3b). Concentrations of K, SO_4 and Sr are not displayed because K and SO_4 presents a conservative behaviour analogous to Na, and Sr values are a few orders of magnitudes lower than the other elements.

Ion exchange has been demonstrated to be important in salinization/freshening of coastal aquifers (Walraevens and Van Camp, 2005; Kim et al., 2003; Moore, 1999; Appelo, 1994; Howard and Lloyd, 1983). However, exposure to clay in the flow system is expected to be small due to the conservative behaviour of Na and K. Therefore, ion exchange was neglected in our calculations.

Figure 4.3 suggests that three processes might be relevant in the mixing zone of S'Almadrava spring:

1) Dissolution/precipitation of carbonates. Ca excess and Mg depletion (figure 4.3a,b) could indicate calcite dissolution and magnesite precipitation. It is well known that mixing of fresh- and saltwater leads to mixing driven dissolution of calcite even if both end-members are saturated with respect to calcite (Rezaei et al., 2005; Sanford and Konikow, 1989; Back et al., 1986; Plummer, 1975)

2) Degasification of CO_2. This is an expected process when deep water is brought to the surface (Price and Herman, 1991; Herman et al., 1985; Plummer et al., 1976) and it is reflected in the measured values of pH (figure 4.2b). Such high values indicate that CO_2 might have escaped from the solution while the water was flowing towards the surface and into the sample bottle, increasing pH and saturation with respect to calcite. Sanz (2007) measured the pH at the spring mouth at atmospheric pressure, which probably caused atmospheric CO_2 to tend to equilibrate with the atmosphere (see $\log \mathrm{P_{CO_2}}$ values of Table 4.1). The measurements of Sanz (2007) can be compared with the pH values measured by Garing et al. (2013) in a different mixing test site in a coastal aquifer of South Mallorca. Garing et al. (2013) measured all pH values at the formation pressure and they all range between 6.7 and 7 at depths corresponding to the mixing zone, while Sanz (2007) measured values from 7.83 to 8.59. Moreover, Garing et al. (2013) observed a higher CO_2 pressure in the mixing zone than towards the surface. We conjecture that the degasification of CO_2 might be relevant and we classified the pH measurements of Table 4.1 as not representative of deep groundwater.

3) CO_2 consumption/production in depth. As speculated by Garing et al. (2013) and Price and Herman (1991) for coastal aquifers in Mallorca, high values of Ca and Mg could be also due to other reactions driven by organic matter oxidation by microorganisms, that could lead to an enhanced dissolution of calcite and magnesite. Krause et al. (2014) and Engel and Randall (2011) recently demonstrated that microorganisms can induce carbonate dissolution by consuming

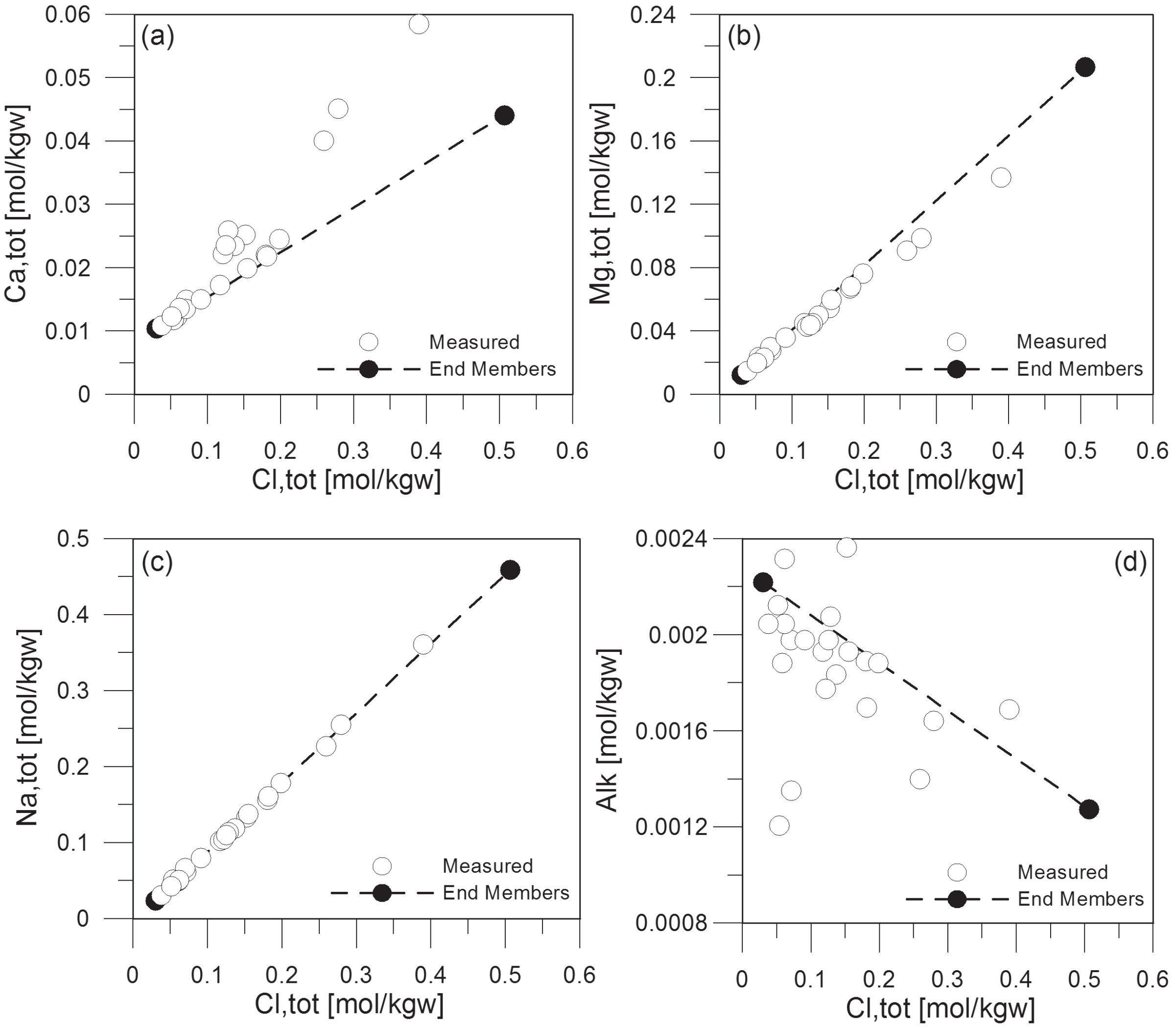


Figure 4.3: Results of (a) Ca, (b) Mg, (c) Na and (d) alkalinity as a function of Cl concentration.

oxygen and releasing CO_2. This can contribute to a decrease of pH values which can lead to further subsaturation with respect to calcite and magnesite and therefore increase in Ca and Mg concentrations.

4.3.2 Chemical System

Given the above conceptual model, the following species and reactions were selected to represent the chemical system at 25 ^{o}C:

$$\begin{aligned}
&\mathrm{R1}: \mathrm{OH^- + H^+ \rightleftharpoons H_2O} && \log \mathrm{K_1} = 13.995\\
&\mathrm{R2}: \mathrm{CO_3^{2-} + H^+ \rightleftharpoons HCO_3^-} && \log \mathrm{K_2} = 10.328\\
&\mathrm{R3}: \mathrm{CO_{2(aq)} + H_2O \rightleftharpoons HCO_3^- + H^+} && \log \mathrm{K_3} = -6.344\\
&\mathrm{R4}: \mathrm{KSO_4^- \rightleftharpoons K^+ + SO_4^{2-}} && \log \mathrm{K_4} = -0.8796\\
&\mathrm{R5}: \mathrm{CaCO_{3(aq)} + H^+ \rightleftharpoons Ca^{2+} + HCO_3^-} && \log \mathrm{K_5} = 7.002\\
&\mathrm{R6}: \mathrm{CaCO_{3(s)} + H^+ \rightleftharpoons Ca^{2+} + HCO_3^-} && \log \mathrm{K_6} = 1.848\\
&\mathrm{R7}: \mathrm{MgCO_{3(s)} + H^+ \rightleftharpoons Mg^{2+} + HCO_3^-} && \log \mathrm{K_7} = 2.294\\
&\mathrm{R8}: \mathrm{CO_{2(g)} \rightleftharpoons CO_{2(aq)}} && \log \mathrm{K_8} = -1.469
\end{aligned} \tag{4.16}$$

Cl^- and Na^+ were also added as conservative species.

The proposed mixing method requires to distinguish between equilibrium and kinetic reactions. Aqueous homogeneous reactions are normally considered in equilibrium, and we defined calcite reaction (R6) as equilibrium and magnesite and CO_2 reactions (R7 and R8, respectively) as kinetic. Saturation of seawater with respect to calcite, in fact, is known to reduce with depth in the Mediterranean sea (Millero et al., 1979). Besides, once freshwater and seawater enter a carbonate aquifer they might be affected by carbonates dissolution and organic matter degradation (Garing et al., 2013) which could produce CO_2, further decreasing SI_{Cc}. However, calcite dissolution is quite fast, so that end-members and mixtures will tend to equilibrate with calcite in the aquifer, which explains why we assumed that all of them are in equilibrium with calcite. CO_2 degasification oversaturates the water, which may lead to some calcite precipitation at shallow depths. However, no signs of travertine formation were identified, so that calcite precipitation is not expected.

The equilibrium and kinetic stoichiometric matrices ($\mathbf{S}_e$ and $\mathbf{S}_k$, respectively) of system (4.16) are

$$\mathbf{S}_e = \begin{array}{c} \\ \mathrm{R1}\\ \mathrm{R2}\\ \mathrm{R3}\\ \mathrm{R4}\\ \mathrm{R5}\\ \mathrm{R6} \end{array} \!\! \begin{array}{c} \begin{array}{ccccccccccccc} \mathrm{Ca^{2+}} & \mathrm{Cl^-} & \mathrm{Na^+} & \mathrm{SO_4^{2-}} & \mathrm{K^+} & \mathrm{CO_{2(aq)}} & \mathrm{Mg^{2+}} & \mathrm{HCO_3^-} & \mathrm{CaCO_{3(aq)}} & \mathrm{H^+} & \mathrm{CO_3^{2-}} & \mathrm{KSO_4^-} & OH^- \end{array} \\ \left(\begin{array}{ccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0\\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0\\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0
\end{array}\right) \end{array} \tag{4.17}$$

$$\mathbf{S}_k = \begin{array}{c} \mathrm{R7}\\ \mathrm{R8} \end{array} \left(\begin{array}{ccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right)$$

The component matrix $\mathbf{U}$ to be used in definition (4.7) can be calculated from $\mathbf{S}_e$ by means of a Gauss-Jordan elimination (Steefel and MacQuarrie, 1996):

$$\mathbf{U} = \begin{array}{c|ccccccccccccc} & Ca^{2+} & Cl^- & Na^+ & SO_4^{2-} & K^+ & CO_{2(aq)} & Mg^{2+} & HCO_3^- & CaCO_{3(aq)} & H^+ & CO_3^{2-} & KSO_4^- & OH^- \\ \hline u_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -0.5 & 0 & 0.5 & -1 & 0 & -0.5 \\ u_2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ u_3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ u_4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ u_5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ u_6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & -0.5 \\ u_7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \qquad (4.18)$$

Note that only aqueous species, without H_2O, were considered to build $\mathbf{S}_e$, $\mathbf{S}_k$ and $\mathbf{U}$ ($N_s = 13$).

The proposed method requires the chemical composition of the end-members to be known (i.e., concentrations of all species to be calculated), in order to fill matrix $\mathbf{X}_c$ of Equation (4.4). For this purpose, the end-members M-21 and Local seawater need to be speciated. Because we are assuming the solution to be degassed, which will increase the pH, we could not use the pH as variable for speciation. Instead, we used total concentrations of main elements and alkalinity:

$$\begin{cases} \mathrm{Ca,tot}: [Ca^{2+}] + [CaCO_{3(aq)}] = x_1 \\ \mathrm{Cl,tot}: [Cl^-] = x_2 \\ \mathrm{Na,tot}: [Na^+] = x_3 \\ \mathrm{S,tot}: [SO_4^{2-}] + [KSO_4^-] = x_4 \\ \mathrm{K,tot}: [K^+] + [KSO_4^-] = x_5 \\ \mathrm{Mg,tot}: [Mg^{2+}] = x_6 \\ \mathrm{Alk}: [HCO_3^-] + 2[CO_3^{2-}] + 2[CaCO_{3(aq)}] + [OH^-] - [H^+] = x_7 \end{cases} \qquad (4.19)$$

where [ ] represents molal concentration (mol/kgw). The data set defined in (4.19), together with the assumption of equilibrium with calcite, is sufficient to obtain to the concentrations of all species. Values $x_1, ..., x_7$ were chosen from Table 4.1. All samples from Table 4.1 were speciated according to (4.19). The resulting pH values at depth are shown in figure 4.4a as a function of chloride concentration. These will be referred to as "measured" values. It can be noticed that these pH values are lower than the ones measured by Sanz (2007) in the field (figure 4.2b), and closer to the pH values measured by Garing et al. (2013). Figure 4.4a indicates a non conservative pH behaviour, which can be expected since pH is an important factor in reaction processes such as carbonate dissolution/precipitation or degasification of solutions.

The presence of other reactions in chemical system (4.16) can be excluded by calculating a component matrix $\mathbf{U}^*$ that eliminates both $\mathbf{S}_e$ and $\mathbf{S}_k$, i.e., such that $\mathbf{U}^* \cdot \binom{\mathbf{S}_e}{\mathbf{S}_k}^t = 0$

$$\mathbf{U}^* = \begin{array}{c|ccccccccccccc} & Ca^{2+} & Cl^- & Na^+ & SO_4^{2-} & K^+ & CO_{2(aq)} & Mg^{2+} & HCO_3^- & CaCO_{3(aq)} & H^+ & CO_3^{2-} & KSO_4^- & OH^- \\ \hline u_1^* & 1 & 0 & 0 & 0 & 0 & 0 & 1 & -0.5 & 0 & 0.5 & -1 & 0 & -0.5 \\ u_2^* & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ u_3^* & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ u_4^* & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ u_5^* & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \qquad (4.20)$$

These components are not affected by the reactions considered in the chemical system (because these reactions have been eliminated by building $\mathbf{U}^*$). The first component $u_1^* = [Ca^{2+}] + [Mg^{2+}] - 0.5[HCO_3^-] - [CO_3^{2-}] - 0.5[OH^-] + 0.5[H^+]$=Ca,tot+Mg,tot-0.5Alk (in Equation 4.19) should only be affected by mixing. Indeed, the behaviour of u_1^*=Ca,tot+Mg,tot-0.5Alk, displayed for all the

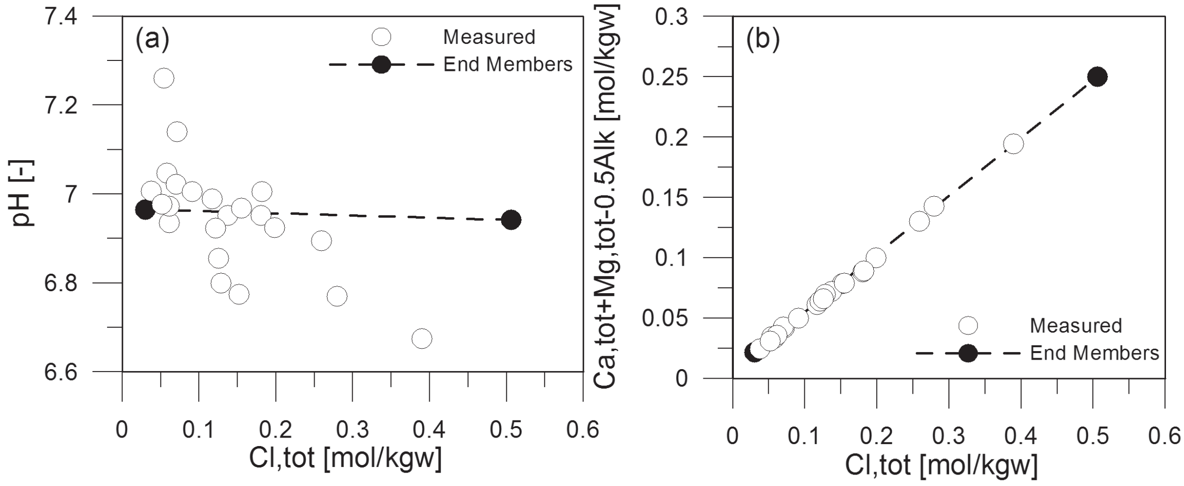


Figure 4.4: (a) pH measurements - as speciation output; (b) Ca,tot+Mg,tot-0.5Alk for end-members and samples as a function of Cl concentration.

samples in figure 4.4b as a function of Cl,tot, is conservative. This supports the choice of our chemical system.

4.3.3 Verification approach

We applied the proposed algorithm to two models:

1) Model 1. Only carbonate dissolution/precipitation process included (i.e., reactions R1-R7 of system 4.16 were considered in the chemical system);

2) Model 2. Both processes of carbonate dissolution/precipitation and CO_2 consumption/production in depth included (i.e., all reactions R1-R8 of system 4.16 were considered in the chemical system).

The process of CO_2 degasification introduced in Section 4.3.2 has been implicitly taken into account by imposing the condition of equilibrium with calcite. That is, Model 1 will yield mixing proportions ($\boldsymbol{\lambda}$) and the quantity of reacted phases ($\mathbf{q}$ of equation 4.4) corresponding only to calcite and magnesite, while Model 2 will also yield the quantity of CO_2 added to the mixture by, presumably, organic matter degradation. The minimum number of data necessary to apply the proposed algorithm (N_d) is 2 for Model 1 and 3 for Model 2.

The steps followed for each model were:

(i) Matrix $\mathbf{X}_c$ of Equation (4.4) was filled with the results of the speciation explained in Section 4.3.2.

(ii) Matrix $\mathbf{A}$ of Equation (4.3) was built using the definition (4.19). Vector $\mathbf{a}^*$ was then filled for every sample by means of the values of Table 4.1. An uncertainty $\sigma = 0.1 \cdot \mu$, where μ is the measured value contained in vector $\mathbf{a}^*$, was assigned to all data to fill matrix $\mathbf{V}$ in equation (4.3).

(iii) The mixing algorithm explained in Section 4.2.2 was applied.

4.4 Results

4.4.1 Model 1: without CO_2 consumption/production process

Figure 4.5 shows the results of total concentrations, pH and alkalinity vs chloride concentration. Computed values of total concentrations of Mg, Na and Ca,tot+Mg,tot-0.5Alk reproduce well the measurements. The lowest values of Ca are in good agreement with measurements, while the highest values deviate a little from the measured values. The algorithm couldn't compute correctly the measured values of pH and alkalinity. pH values are underestimated, while calculated values of alkalinity seem to follow almost a conservative behaviour. For this reason, this model was descarted and mixing ratios and the quantities of reacted species ($q_{calcite}$ and $q_{magnesite}$), which are similar to the results for Model 2 in the following section, are not shown.

4.4.2 Model 2: with CO_2 consumption/production in depth

Figure 4.6 shows the calculated values of total concentrations, pH and alkalinity for the model that includes CO_2 consumption/production process. In this case excellent fits were otained for all the species.

Calculated mixing proportions relative to seawater are shown in figure 4.7a as a function of time. It can be noticed that the proportions of seawater follow the trends of EC and Cl in figure 4.2a and c, respectively. This is a consistent result, considering that EC and Cl concentrations are expression of the presence of seawater. The calculated quantities of reacted phases ($q_{calcite}$, $q_{magnesite}$ and q_{CO_2}) are displayed in figure 4.7b, c and d respectively. Positive values of $q_{calcite}$ and $q_{magnesite}$ indicate dissolution, while negative correspond to precipitation. CO_2 consumption is represented by $q_{CO_2} < 0$, while positive values of q_{CO_2} indicate CO_2 production . Model results indicate that calcite has dissolved, while magnesite has precipitated. This is in good agreement with measurements of figure 4.3a,b and with calculations of other authors (Garing et al., 2013; Sanz, 2007; Rezaei et al., 2005). The maximum values of dissolved calcite and precipitated magnesite correspond to the maximum proportion of seawater, which is consistent with the results of Sanz et al. (2011), who found that calcite dissolution by mixing fresh and seawater is maximum for a 0.5 mixing ratio. The tendency of q_{CO_2} is not as clear as for $q_{calcite}$ or $q_{magnesite}$ (see figure 4.7d). The values are quite scattered around the mean ($q_{CO_2,av} = -0.0001$), however most of the samples indicate that CO_2 has been consumed ($q_{CO_2} < 0$). We cannot think of any process extracting CO_2 from the mixture. Therefore we conjecture that this result may be caused by limitations in the conceptual model, possibly the assumption of fresh and seawater in equilibrium with calcite. Interestingly $q_{CO_2} > 0$, which might be caused by degradation of organic matter, for the samples with largest EC values (i.e., seawater proportion). These are the samples corresponding to the reactivation of the spring after two rainfall events. This means that they have the longest residence time, during which the water could react with the aquifer matrix producing organic matter degradation. q_{CO_2} values are a couple of orders of magnitudes smaller than $q_{calcite}$ or $q_{magnesite}$, therefore the process of the CO_2 consumption/production at depth might seem secondary with respect to carbonate dissolution/precipitation. However, notice the small amount of CO_2 necessary to reproduce the alkalinity measurements in this model (figure 4.5d) with respect to Model 1 (figure 4.6d). For this reason we can deduce that its presence as process is justified. Moreover, as also the high values of Ca are reproduced well in this model, we can infer that the microbiological activity might be responsible for an enhancement of the calcite dissolution process, by producing CO_2 and decreasing the pH even more.

4.5 Conclusions

We presented an algorithm to calculate mixing proportions and reacted quantities when the chemical composition of a number of end-members is available. The proposed method allows the use of an indefinite number of data and the definition of the uncertainty associated to measured data. Unlike other mixing algorithms available, it allows to impose equilibrium conditions relative to reacting phases on the mixture.

The method was verified by means of a set of geochemical data sampled by Sanz (2007) at a brackish spring of Mallorca island (Spain), which is the product of mixing between freshwater and seawater intruding in depth of a carbonate coastal aquifer. Measurements indicate that, besides conservative mixing, chemical reactions affect the chemical composition of the water discharged by the spring, in particular total concentrations of Ca, Mg, alkalinity and pH.

From the data it was inferred that three processes might be relevant for S'Almadrava spring chemical composition: carbonates dissolution/precipitation, CO_2 consumption/production due to microbiological activity and CO_2 degasification of the water flowing from depth towards the surface.

Two models were chosen to test the proposed algorithm: Model 1, in which only dissolution/precipitation of calcite and magnesite is considered, and Model 2, in which CO_2 consumption/production in depth is added to the carbonates dissolution/precipitation. CO_2 degassing due to the mixture flowing towards the surface was implicitly taken into account in both models by imposing the equilibrium condition with calcite for end-members and samples.

The algorithm could calculate mixing proportions of freshwater and seawater which are in good agreement with measured values of EC and Cl concentration for both models.

Measured values of total concentrations are reproduced well in both models, even if with higher accuracy of Model 2 with respect to Model 1, while good fits for alkalinity and pH are obtained only for Model 2. This suggests that the CO_2 consumption/production in depth might be relevant in the chemical composition of the spring. Both models indicate that calcite dissolution and magnesite precipitation are the main geochemical processes, while Model 2 suggests that both CO_2 consumption and production need to occur in order to explain the measurements. Another process that might produce CO_2, besides CO_2 degasification, does not occur to us. Therefore we conjecture that CO_2 production results are due to limitations in the conceptual model. Probably the equilibrium with calcite condition increased too much the CO_2 pressure. Degradation of organic matter can explain the results of Model 2 corresponding to CO_2 consumption and occurring when EC (i.e., seawater proportion) is maximum. These peaks correspond to longest residence times for the water, which had more time to interact with the aquifer matrix and, therefore, more time to be exposed to microorganisms activity. Finally, since the highest measurements of Ca, corresponding to the lowest pH values, are reproduced well only in Model 2, it can be inferred that the activity of microorganisms enhances the dissolution of calcite by means of the production of CO_2 which decreases the pH and subsaturates the water with respect to calcite.

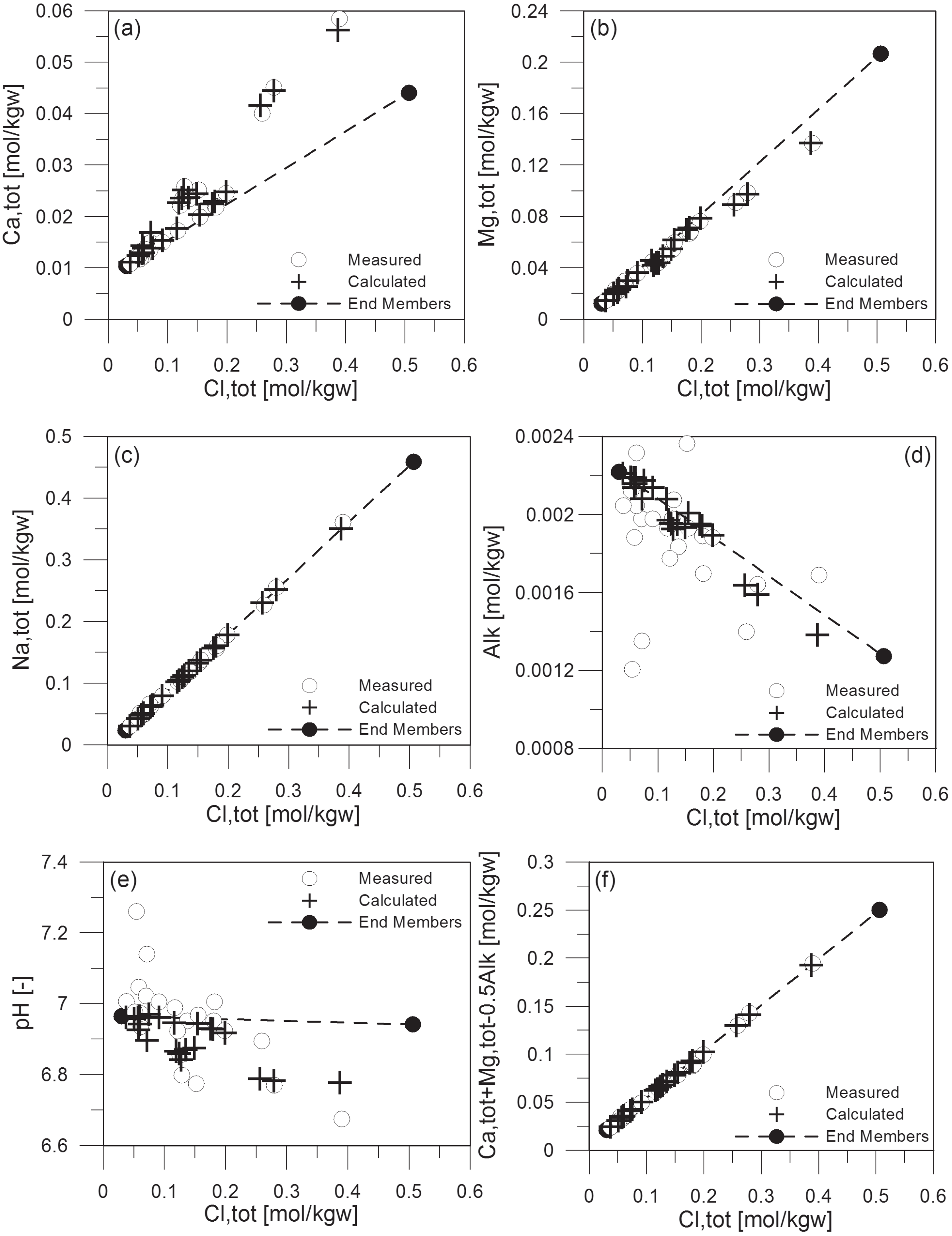


Figure 4.5: Results of Model 1 (without $CO_{2(g)}$ consumption/production): total concentrations of (a) Ca; (b) Mg; (c) Na; (d) Alkalinity; (e) pH and (f) Ca,tot+Mg,tot-0.5Alk as a function of Cl concentration.

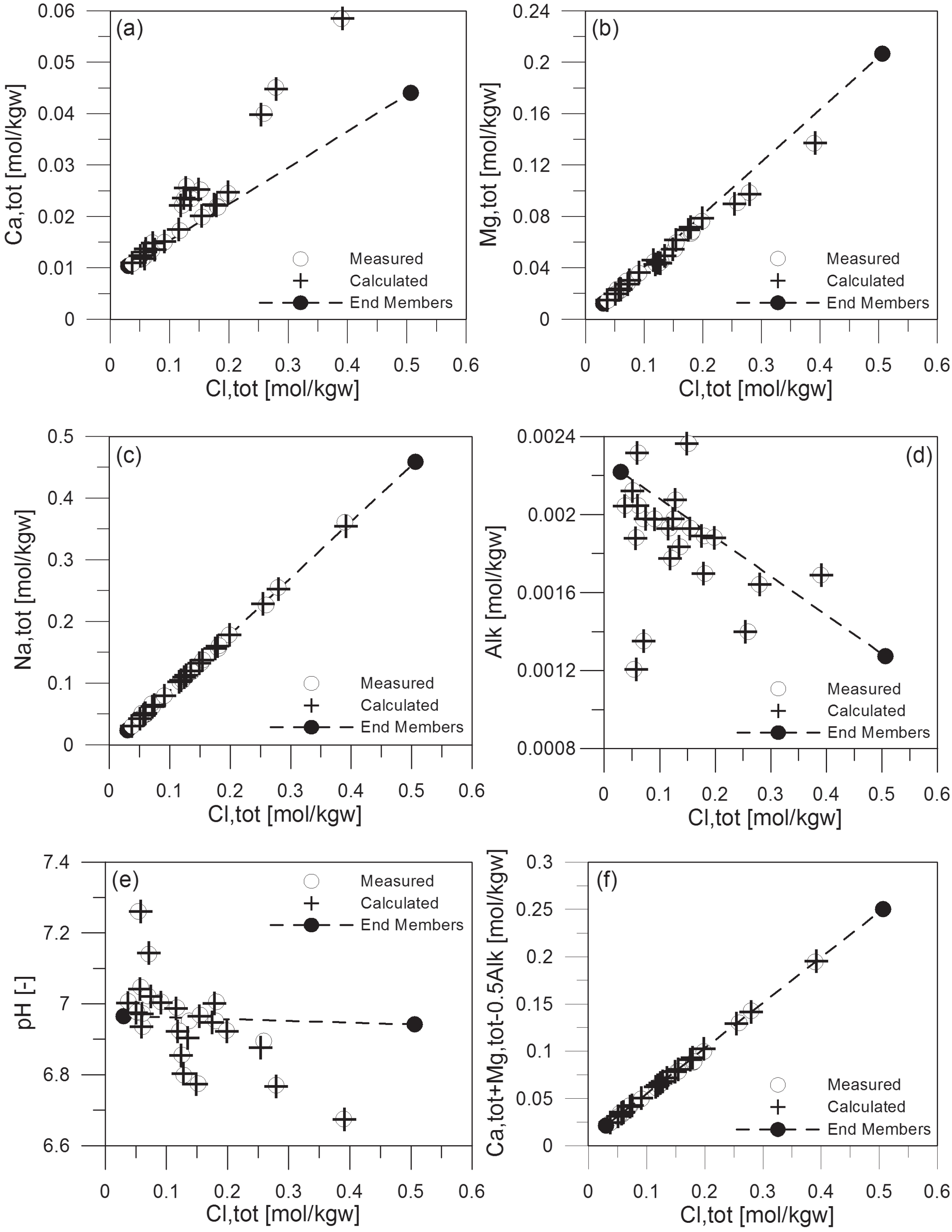


Figure 4.6: Results of Model 2 (with $CO_{2(g)}$ consumption/production): total concentrations of (a) Ca; (b) Mg; (c) Na; (d) Alkalinity; (e) pH and (f) Ca,tot+Mg,tot-0.5Alk as a function of Cl concentration.

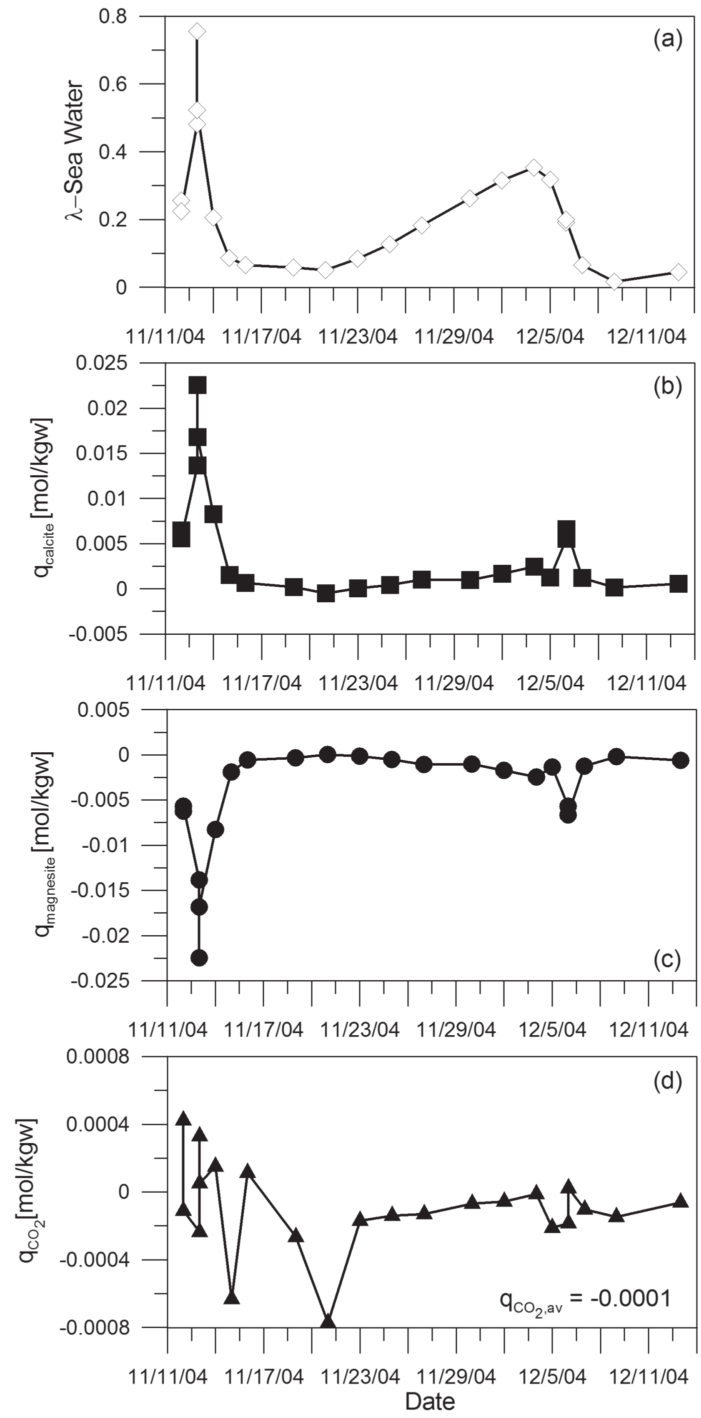


Figure 4.7: Results of Model 2 (with $CO_{2(g)}$ consumption/production): time evolution of (a) Mixing proportions of seawater, (b) Calcite dissolution ($q_{calcite} > 0$), (c) Magnesite precipitation ($q_{magnesite} < 0$), (d) $CO_{2(g)}$ production ($q_{CO2(g)} < 0$) and consumption ($q_{CO2(g)} > 0$) together with average value of $q_{CO2(g),av} = -0.0001$. Notice the small amount of $CO_{2(g)}$ required to the significant improvement of alkalinity and pH in figure 4.6 compared to figure 4.5.

Chapter 5

Decoupling constant activity species in reactive transport algorithm

5.1 Introduction

The importance of reactive transport is evident from the large number of papers about this topic (Saaltink et al., 2001; Van Der Lee and De Windt, 2001; Mayer et al., 2002; De Simoni et al., 2005; Steefel et al., 2005; Gamazo et al., 2012; Huo et al., 2014; Steefel et al., 2014, amongst many others) and the amount of codes recently developed to solve its cumbersome equations (Parkhurst et al., 1999; Steefel and Yabusaki, 2000; Saaltink et al., 2004b; Xu et al., 2006; Bea et al., 2009; Steefel, 2009; Hammond et al., 2014).

Reactive transport requires the solution of partial differential equations expressing the mass balance of each chemical species, introduced in Chapter 2, Section 2.1.2, together with the mass action laws representing thermodynamic equilibrium, also presented in Chapter 2, Section 2.1.1. Due to the complexity and to the non-linearity of geochemical processes, multicomponent reactive transport requires a significant computational effort.

Two families of numerical methods are available to solve reactive transport: one based on Picard and the other based on Newton-Raphson methods. The latter consists of substituting the chemical equations in the transport system and solving the whole non-linear system at every time step. For this reason it is called "fully coupled" or "Direct Substitution Approach" (DSA). The other method consists of decoupling the solution of transport from the chemistry. This is why it is called "operator splitting" or "Sequential Iteration Approach" (SIA). Both methods have been studied and compared from theoretical (Yeh and Tripathi, 1989) and applied (Saaltink et al., 2001; Carrayrou et al., 2010) points of view. In general, the DSA method is more robust than the SIA (Saaltink et al., 2001), and it performs better for complex cases with strong gradients of concentrations (Carrayrou et al., 2010). However, the implementation of the fully coupled method is more demanding than the one for the operator splitting, as it requires the calculation of the derivatives of the transport equations at every node with respect to every unknown. Moreover, operator splitting techniques allow the use of different specific methods for solving both transport and chemical sets of equations. Overall, the advantage of one method with respect to the other depends on the level of complexity of the chemistry to be solved, grid size and computational capabilities. It is generally believed that SIA is more suitable for chemically simple examples on large grids (Saaltink et al., 2001).

Regardless of the way chemistry and transport are coupled, authors have searched for techniques to simplify this type of calculations. One such possibility consists of decoupling constant activity species (CAS) such as pure minerals in equilibrium, proton if pH is fixed or gases whose partial pressure is constant. Saaltink et al. (1998) proposed a formulation to solve reactive transport decoupling CAS and applied to DSA and SIA. They concluded that decoupling CAS is especially advantageous for the DSA. De Simoni et al. (2005) present another procedure to eliminate CAS and solve reactive transport in case of homogeneous and heterogeneous equilibrium reactions, while Molins et al. (2004) allow to decouple components also in case of kinetic reactions. Lately, Huo et al. (2014) proposed a methodology to define different "component domains" cheracterized by different sets of primary species connected at the boundaries.

The decoupling of CAS is assumed to be particularly advantageous for DSA more than SIA (Saaltink et al., 1998). The coupling of all the equations in DSA, in fact, can benefit from a reduction of the number of variables. For SIA, however, as transport is solved independently for every component, the definition of different component zones does not seem so favorable. Still, decoupling CAS can be useful for the SIA chemical step. Eliminating CAS from the speciation reduces the system to be solved iteratively and might lead to some advantages in terms of chemical iterations.

We propose a formulation to decouple CAS from the SIA chemical step. We implemented the proposed algorithm in CHEPROO++ and PROOST, an Object-Oriented code for multiphase flow and transport developed by the GHS group (Slooten et al., 2010). We compared the proposed and the traditional methods in terms of number of iterations (for Picard and for the chemical loop) and in terms of CPU time on a synthetic example.

5.2 Methodology

The SIA requires two steps, "transport" and "chemical", to be solved sequentially for every time step and to be repeated until convergence is reached. The transport and chemical steps are solved in an iterative Picard loop contained in the more external time loop. The first step consists of solving the N_c transport equations (2.16) of Chapter 2 for $\mathbf{u}_{aq}$, total aqueous concentrations of every component, being $N_c = N_1$ the number of components and N_1 the number of primary species.

5.2.1 Transport step

The transport equation (2.16) of Chapter 2 needs to be discretized in space and time to solve for $\mathbf{u}_{aq}$. Using, for example, finite elements and finite differences for spatial and temporal discretization, respectively, equation (2.16) of Chapter 2 can be written as follows

$$\begin{aligned}\mathbf{F}_{aq}\frac{\mathbf{u}_{aq,j}^{k+1,i+1}-\mathbf{u}_{aq,j}^{k}}{\Delta t}+\mathbf{F}_{gas}\frac{\mathbf{u}_{gas,j}^{k+1,i}-\mathbf{u}_{gas,j}^{k}}{\Delta t}+\mathbf{F}_{s}\frac{\mathbf{u}_{min,j}^{k+1,i}-\mathbf{u}_{min,j}^{k}}{\Delta t}+\mathbf{F}_{s}\frac{\mathbf{u}_{ads,j}^{k+1,i}-\mathbf{u}_{ads,j}^{k}}{\Delta t}=\\ \mathbf{E}_{aq}\mathbf{u}_{aq,j}^{k+\theta,i+1}+\mathbf{E}_{gas}\mathbf{u}_{gas,j}^{k+\theta,i}+\mathbf{U}\mathbf{S}_{k}^{t}\mathbf{r}_{k}^{k+\theta,i}+\mathbf{g}_{j}\qquad j=1,...,N_c\end{aligned}\tag{5.1}$$

Superscripts k and i refer to time step and Picard iteration, respectively, and θ is the time weighting factor ranging between 0 and 1. $\mathbf{F}$ is a diagonal matrix containing volumetric fluid fractions (θ_{aq}) and $\mathbf{E}$ is a non diagonal matrix containing advection and dispersion terms. $\mathbf{g}_j = Q_c(\mathbf{u}_{aq,j}^{ext} - \mathbf{u}_{aq,j}^{k+\theta})$ is the sink-source term of the convective form of the transport equation (Saaltink et al., 2004a; Diersch, 1998), where $Q_c\mathbf{u}_{aq}^{ext}$ is the mass flow of the external sink/source. $\mathbf{F}_{min}$ and $\mathbf{F}_{ads}$ have been grouped under the solid volumetric fraction, $\mathbf{F}_s$. For further details about how to calculate these matrices and computational methods to solve reactive transport see Huyakorn (2012); Bundschuh and Zilberbrand (2011) or Zheng and Bennett (1995), amongst many others. Note that the concentrations of components of phases other than aqueous and kinetic terms are computed at the previous Picard iteration. This, together with the following time interpolation

$$\mathbf{u}_{aq}^{k+\theta} = \theta\mathbf{u}_{aq}^{k+1} + (1-\theta)\mathbf{u}_{aq}^{k}\tag{5.2}$$

allows to re-write equation (5.1) as a set of linear N_c equations

$$\mathbf{A}\mathbf{u}_{aq,j}^{k+1,i+1} = \mathbf{b}_j^i \qquad j = 1, ..., N_c \tag{5.3}$$

where matrix $\mathbf{A}$ is identical to the one used for solving conservative transport

$$\mathbf{A} = \left(\frac{\mathbf{F}_{aq}}{\Delta t} - \theta \mathbf{E}_{aq} \right) + Q_c \tag{5.4}$$

and vector $\mathbf{b}_j^i$ is the sum of two vectors

$$\mathbf{b}_j^{i+1} = \mathbf{b}_{cons,j} + \mathbf{b}_{reac,j}^{k,i+1} \qquad j = 1, ..., N_c \tag{5.5}$$

The first is the same of the conservative transport and the second represents chemical reactions, to which phases other than aqueous and kinetic terms contribute

$$\mathbf{b}_{cons,j} = \left(\frac{\mathbf{F}_{aq}}{\Delta t} - (1-\theta)\mathbf{E}_{aq} \right) \mathbf{u}_{aq,j}^k + Q_c \mathbf{u}_{aq,j}^{ext} \qquad j = 1, ..., N_c \tag{5.6}$$

$$\begin{aligned} \mathbf{b}_{reac,j}^i = \mathbf{E}_{gas}\mathbf{u}_{gas,j}^{k+\theta,i} + \mathbf{U}\mathbf{S}_k^t\mathbf{r}_k^{k+\theta,i} - \mathbf{F}_{gas}\frac{\mathbf{u}_{gas,j}^{k+1,i} - \mathbf{u}_{gas,j}^k}{\Delta t} - \mathbf{F}_s\frac{\mathbf{u}_{min,j}^{k+1,i} - \mathbf{u}_{min,j}^k}{\Delta t} + \\ -\mathbf{F}_s\frac{\mathbf{u}_{ads,j}^{k+1,i} - \mathbf{u}_{ads,j}^k}{\Delta t} \qquad j = 1, ..., N_c \end{aligned} \tag{5.7}$$

The solution of system (5.3) leads to the vector of aqueous components at every iteration $i+1$ of every time step $k+1$, $\mathbf{u}_{aq,j}^{k+1,i+1}$. Note that matrix $\mathbf{A}$, unlike vector b_j is the same for every component. Moreover, the set of equations for each component is independent from the others, unless the components definition (i.e., the component matrix $\mathbf{U}$) is variable in space. Thus, system (5.3) can be solved iteratively for each component separately. This is the main computational advantage of SIA with respect to other reactive transport algorithms. The adopted convergence criteria for system (5.3) is the following

$$\left| \frac{\mathbf{u}_{aq,j}^{k+1,i+1} - \mathbf{u}_{aq,j}^{k+1,i}}{\mathbf{u}_{aq,j}^{k+1,i+1}} \right|_{max} < \varepsilon_{tr} \qquad j = 1, ..., N_c \tag{5.8}$$

where ε_{tr} is a threshold value defined by the user.

The transport step was implemented in PROOST.

5.2.2 Chemical step with traditional speciation

The chemical step consists of updating the reactive independent term, $\mathbf{b}_{reac,j}$, once $\mathbf{u}_{aq,j}^{k+1,i+1}$ is calculated. This requires the concentrations of the species in all phases to be calculated. For this scope, a speciation is solved in CHEPROO++ by means of the mass action laws introduced in Section 2.1.1 of Chapter 2 and of the following mass balance equation

$$\begin{aligned} \mathbf{u}_{tot} = F_{aq}\mathbf{u}_{aq}^{k+1,i+1} + F_s \sum_{\alpha,ads} \hat{\theta}_{\alpha,ads}\mathbf{U}_{\alpha,ads}\mathbf{c}_{\alpha,ads}^i + F_g \sum_{\alpha,gas} \hat{\theta}_{\alpha,gas}\mathbf{U}_{\alpha,gas}\mathbf{c}_{\alpha,gas}^i \\ + F_s \sum_{\alpha,min} \hat{\theta}_{\alpha,min}\mathbf{U}_{\alpha,min}\mathbf{c}_{\alpha,min}^i \end{aligned} \tag{5.9}$$

$\mathbf{u}_{tot}$ ($N_c = N_1$) is the vector of total concentrations in all phases and $\hat{\theta}_{\alpha,ads}$, $\hat{\theta}_{\alpha,min}$ and $\hat{\theta}_{\alpha,gas}$ are the volumetric fractions of the adsorbed, mineral and gas α-phases, respectively. For simplicity we assume that the mineral phases are pure. Therefore, as explained in chapter 2, the concentration $\mathbf{c}_{\alpha,min}$ is constant, while $\hat{\theta}_{\alpha,min}$ is the variable.

In Section 3.2.1 we have presented the system of equations (3.4) to be solved for a traditional speciation. A similar system needs to be solved for the speciation of the chemical step, in which the set of N_1 data equations, $\mathbf{f}_t$, is the definition of components

$$\begin{cases} \mathbf{f}_t = \mathbf{g}(\mathbf{c}) - \mathbf{x} = 0 \\ \mathbf{f}_{MAL}(\mathbf{c}) = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{U}\mathbf{c} - \mathbf{u}_{tot} = 0 \\ \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \end{cases} \tag{5.10}$$

where the vector $\mathbf{u}_{tot}$ has been defined in (5.9) and $\mathbf{U}$ is the component matrix defined in Chapter 2, equation (2.17). The set of N_2 mass action laws has also been introduced in Chapter 2, Section 2.1.1. N_2 is the number of secondary species. The number of equations of system (5.10), $N_1 + N_2$, coincides with the number of unknowns, the concentrations of the N_s species.

The traditional algorithm to solve system (5.10) requires expressing the concentrations of secondary species, $\mathbf{c}_2$, as a function of $\mathbf{c}_1$ by means of $\mathbf{f}_{MAL}$ (as explained in Section 2.1.1), and then substitute them in the N_1 data equations, $\mathbf{f}_t$, to calculate $\mathbf{c}_1$. The substitution of $\mathbf{f}_{MAL}$ into $\mathbf{f}_t$ renders the system (5.10) non-linear. It is common practice, then, to linearize it with respect to the unknowns, $\mathbf{c}_1$, and solve it by means of an iterative system.

$$\mathbf{f}_t = 0 \Leftrightarrow \mathbf{J}^i(\mathbf{c}_1^{i+1} - \mathbf{c}_1^i) = -\mathbf{f}_t^i \tag{5.11}$$

$\mathbf{J}^i$ ($N_1 \times N_1$) is the Jacobian matrix at iteration i containing the derivatives of $\mathbf{f}_t$ with respect to $\mathbf{c}_1$.

$$\mathbf{J}^i = \frac{d\mathbf{f}_t^i}{d\mathbf{c}_1^i} = \mathbf{U}_1 + \mathbf{U}_2 \frac{\partial \mathbf{c}_2^i}{\partial \mathbf{c}_1^i} \tag{5.12}$$

Matrices $\mathbf{U}_1$ and $\mathbf{U}_2$ are the two blocks of the component matrix relative to primary and secondary species, respectively, defined in Chapter 2, equation (2.17). For details on the calculation of $\partial \mathbf{c}_2^i / \partial \mathbf{c}_1^i$ see Section D.3 of Appendix D.

System (5.11) needs to be solved iteratively until convergence, which is usually verified by means of two criteria: the first relative to the maximum relative increment of the primary species concentrations

$$\left| \frac{\mathbf{c}_1^{i+1} - \mathbf{c}_1^i}{\mathbf{c}_1^{i+1}} \right|_{max} < \varepsilon_{pr} \tag{5.13}$$

and the second relative to the residual of equation (5.11)

$$|\mathbf{f}_t|_{max} < \varepsilon_t \tag{5.14}$$

Both ε_{pr} and ε_t are threshold values defined by the user.

The traditional speciation method includes as unknowns also the constant activity species (CAS), whose concentration is calculated at every iteration even if their activity is fixed. This is not very efficient. For this reason CHEPROO++ uses a definition of the component matrix in which CAS are defined as primary species and can be decoupled from the speciation algorithm.

5.2.3 Speciation decoupling constant activity species

Differentiating between reduced primary species ($\mathbf{c}_{1,nc}$, $N_{1,nc} = N_1 - N_{CAS}$), CAS ($\mathbf{c}_{1,CAS}$, N_{CAS}), N_{CAS} being the number of CAS, and secondary species, ($\mathbf{c}_2$, N_2)

$$\mathbf{c} = \begin{pmatrix} \mathbf{c}_{1,nc} \\ \mathbf{c}_{1,CAS} \\ \mathbf{c}_2 \end{pmatrix} \tag{5.15}$$

and using the definition (2.25) of the component matrix allows us to decouple the vector of concentrations $\mathbf{c}_{1,CAS}$ from the speciation. In fact, neither $\mathbf{c}_{1,nc}$ depend on $\mathbf{c}_{1,CAS}$ through mass balance equations, because of definition (2.25), nor $\mathbf{c}_2$ depend on $\mathbf{c}_{1,CAS}$ through mass action laws, because the activity of CAS is constant. This way, $\mathbf{c}_2$ and $\mathbf{c}_{1,nc}$ can be calculated in the speciation iterative process, prior to obtaining $\mathbf{c}_{1,CAS}$. Thereafter, when convergence has been reached, $\mathbf{c}_{1,CAS} = f(\mathbf{c}_2)$ can be calculated from the values of components, which result from mass balance (i.e., transport) equations. As a consequence, the speciation system to be solved in the iterative process is smaller because the number of equations has reduced from N_s to $N_s - N_{CAS}$

$$\begin{cases} \mathbf{f}_t = \mathbf{U}_{nc}\mathbf{c} - \mathbf{u}_{tot}^{red} = 0 \\ \mathbf{f}_{MAL} = \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \end{cases} \tag{5.16}$$

Note that the dimension of the first set of equations in system (5.16) has been reduced to ($N_{1,nc}$). The same linearization defined in (5.11) can be applied to (5.16), with the difference that the Jacobian matrix is smaller ($N_{1,nc} \times N_{1,nc}$)

$$\mathbf{f}_t = 0 \Leftrightarrow \mathbf{J}^i(\mathbf{c}_{1,nc}^{i+1} - \mathbf{c}_{1,nc}^i) = -\mathbf{f}_t^i \tag{5.17}$$

The convergence criteria defined in (5.13) and (5.14) can be adopted for this iterative process, with the difference that criterion (5.13) will verify the maximum relative increment of $\mathbf{c}_{1,nc}$.

This speciation algorithm can be used in the SIA chemical step. Notice that this component definition is not convenient for the transport step because it can vary in space according to the CAS definition. Therefore, CAS need to be eliminated from $\mathbf{u}_{tot}$ defined in (5.9), i.e., a correspondence between $\mathbf{u}_{tot}$ (N_1) (recall 2.17) and $\mathbf{u}_{tot}^{red}$ ($N_{1,nc}$) (defined in 2.25) needs to be defined. For this purpose, an elimination matrix $\mathbf{E}$ ($N_{1,nc} \times N_1$) can be defined such that

$$\mathbf{u}_{tot}^{red} = \mathbf{E}\mathbf{u}_{tot} \tag{5.18}$$

where $\mathbf{E}$ is calculated as follows

$$\begin{aligned} &\mathbf{E}\mathbf{U} = \mathbf{U}_{nc} \\ &\Rightarrow \mathbf{E}\mathbf{U}\mathbf{U}^t = \mathbf{U}_{nc}\mathbf{U}^t \\ &\Rightarrow \mathbf{E} = \mathbf{U}_{nc}\mathbf{U}^t(\mathbf{U}\mathbf{U}^t)^{-1} \end{aligned} \tag{5.19}$$

Besides of the fact that a smaller system needs to be solved iteratively, another possible advantage of decoupling CAS is that $\mathbf{u}_{tot}^{red}$ are conservative with respect to CAS. This means that, if the set of CAS does not change, no adsorption/gas reactions are included and only equilibrium reactions are defined, it is necessary to solve the speciation only once at the first i-th Picard iteration for every time step and every node. This might be a benefit for reactive transport simulations, possibly leading to reduce the number of chemical iterations. In practice, it is necessary to check the residual of the first set of equations in system (5.16)

$$\left|\mathbf{U}_{nc}\mathbf{c} - \mathbf{u}_{tot}^{red}\right|_{max} < \varepsilon_r \tag{5.20}$$

where ε_r is a threshold value for the residual defined by the user. If condition (5.20) is met the speciation is not necessary.

The proposed algorithm to solve the SIA chemical step decoupling CAS is included in the description of the reactive transport algorithm in the following section.

5.2.4 Algorithm

The steps of the SIA algorithm can be outlined as follows

1. k=0: Set initial conditions for flow (pressure/heads) and transport (components)
2. Start time loop: k=k+1
   (a) Solve flow
   (b) Compute conservative transport matrices $\mathbf{E}_{aq}$ and $\mathbf{F}_{aq}$
   (c) Get $\mathbf{u}_{aq,j}^{k} = \mathbf{U}_{aq,j}\mathbf{c}_{aq,j} \qquad j=1,...,N_c$
   (d) Compute system matrix $\mathbf{A}$ and $\mathbf{b}_{cons,j} \qquad j=1,...,N_c$
   (e) Set $\mathbf{b}_j^{i=0} = \mathbf{b}_{cons,j}$ $(\mathbf{b}_{reac,j}^{k,i=0} = \mathbf{0}) \qquad$ j=1,...,N_c
   (f) Start Picard loop
      i. Transport step: solve $\mathbf{A}\mathbf{u}_{aq,j}^{k+1,i+1} = \mathbf{b}_j^i$ to calculate $\mathbf{u}_{aq,j}^{k+1,i+1} \qquad$ j=1,...,N_c
      ii. Chemical step: given $\mathbf{u}_{aq}^{k+1,i+1}$, F_{aq}, F_s, F_{tot} speciate in every node of the domain from $\mathbf{u}_{tot}$ defined in (5.9)
      iii. Update $\mathbf{b}_{reac,j}^{k,i+1} \qquad$ j=1,...,N_c
      iv. Compute $\mathbf{b}_j^{i+1} = \mathbf{b}_{cons,j} + \mathbf{b}_{reac,j}^{k,i+1} \qquad$ j=1,...,N_c
      v. Set i=i+1 and repeat steps i. to iv. until convergence

All the transport steps described above were implemented in PROOST.

The details of the chemical step implemented in CHEPROO++ are explained below. For simplicity we consider only one mineral, gas and adsorbed phases, and we assume that the mineral is a pure phase in equilibrium (i.e., it is a CAS).

1. Build $\mathbf{u}_{tot} = \theta_{aq}\mathbf{u}_{aq}^{k+1,i+1} + \theta_{ads}\mathbf{U}_{ads}\mathbf{c}_{ads}^{i} + \theta_{gas}\mathbf{U}_{gas}\mathbf{c}_{gas}^{i} + \theta_{min}\mathbf{U}_{min}\mathbf{c}_{min}^{i}$
   where $\theta_{aq} = F_{aq}/F_{tot}$, $\theta_{ads} = \hat{\theta}_{ads}F_s/F_{tot}$, $\theta_{gas} = \hat{\theta}_{gas}F_g/F_{tot}$, $\theta_{min} = \hat{\theta}_{min}F_s/F_{tot}$
2. Calculate $\mathbf{u}_{tot}^{red} = \mathbf{E}\mathbf{u}_{tot}$
3. Check condition (5.20). If it's verified go to step 5., otherwise continue.
4. Speciate from $\mathbf{u}_{tot}^{red} = \theta_{aq}\mathbf{E}\mathbf{u}_{aq}^{k+1,i+1} + \theta_{ads}\mathbf{E}\mathbf{U}_{ads}\mathbf{c}_{ads}^{i} + \theta_{gas}\mathbf{E}\mathbf{U}_{gas}\mathbf{c}_{gas}^{i}$
5. Calculate $\Delta\theta_{min}^{i+1}$ from $\Delta\theta_{min}^{i+1} + \mathbf{S}_{e1,CAS,aq}^{*t}\theta_{aq}\mathbf{c}_{2,aq}^{i+1} + \mathbf{S}_{e1,CAS,ads}^{*t}\theta_{ads}\mathbf{c}_{2,ads}^{i+1} + \mathbf{S}_{e1,CAS,gas}^{*t}\theta_{gas}\mathbf{c}_{2,gas}^{i+1} = 0$
6. Update $\theta_{min}^{i+1} = \theta_{min}^{i} + \Delta\theta_{min}^{i+1}$

Steps 2. to 6. are the novelties of the proposed algorithm. The traditional chemical step requires a speciation to be solved directly after step 1.

We applied the proposed algorithm to a synthetic case described in the following section. The proposed algorithm was compared to the traditional method in terms of quality of results and performances.

5.3 Application

A test case presented by Saaltink et al. (2001) was used to compare the traditional and proposed methods for reactive transport. The example consists of a one-dimensional domain (see figure 5.1) in which the resident water is initially in equilibrium with calcite. A subsaturated water is injected for 5 years in the domain (i.e, until pore volume has been flushed), causing calcite to dissolve. Chemical compositions of initial and boundary waters are listed in Table 5.1.

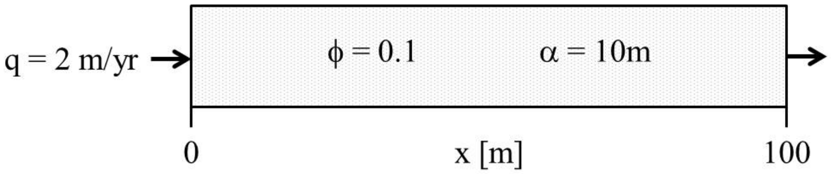


Figure 5.1: Domain geometry and parameters used for CAL simulations.

Table 5.1: Chemical characeristics of initial and boundary waters. c represents total aqueous concentration.

	Species	Initial water	Boundary water
c	Ca^{2+}	$9.75 \cdot 10^{-4}$	$4.0 \cdot 10^{-5}$
$[mol/l]$	HCO_3^-	$1.0 \cdot 10^{-3}$	$3.0 \cdot 10^{-5}$
	Cl^-	$1.08 \cdot 10^{-4}$	$1.96 \cdot 10^{-4}$
-loga $[mol/l]$	H^+	8.0	5.5

Following Saaltink et al. (2001), we simulated different scenarios: one in which calcite is allowed to dissolve in equilibrium (CAL-Eq.), and three in which calcite is dissolving in kinetic (CAL-1, CAL-2, CAL-3) with variable kinetic rate: from the slowest (CAL-1, corresponding to the smallest reactive surface) to the fastest (CAL-3, with highest reactive surface). A simplified expression of the kinetic reaction rate of Steefel and Lasaga (1994) was used

$$r_k = k \cdot \sigma \cdot (1 - \Omega) \tag{5.21}$$

where k is the kinetic rate constant and σ is the reactive surface. The values used in the example are listed in Table 5.2 for the different test cases.

The reactions considered in the chemical system at 25 ^oC are the following, together with the conservative species Cl^-

Table 5.2: Chemical parameters for kinetic rate calculations.

Model	Rate constant (k) $[mol \cdot m^{-2} \cdot s^{-1}]$	Initial reactive surface (σ) $[m^{-1}]$
CAL-1	$4.64 \cdot 10^{-7}$	$6.8 \cdot 10^{-5}$
CAL-2	$4.64 \cdot 10^{-7}$	$6.8 \cdot 10^{-4}$
CAL-3	$4.64 \cdot 10^{-7}$	$6.8 \cdot 10^{-3}$

$$\begin{aligned}
&\mathrm{R1: OH^- + H^+ \rightleftharpoons H_2O} && \mathrm{\log K_1 = 13.995}\\
&\mathrm{R2: CO_3^{2-} + H^+ \rightleftharpoons HCO_3^-} && \mathrm{\log K_2 = 10.328}\\
&\mathrm{R3: CO_{2(aq)} + H_2O \rightleftharpoons HCO_3^- + H^+} && \mathrm{\log K_3 = -6.344}\\
&\mathrm{R4: CaHCO_3^+ \rightleftharpoons Ca^{2+} + HCO_3^-} && \mathrm{\log K_4 = -1.0467}\\
&\mathrm{R5: CaCO_{3(s)} + H^+ \rightleftharpoons Ca^{2+} + HCO_3^-} && \mathrm{\log K_5 = 1.848}
\end{aligned} \tag{5.22}$$

The CAL-Eq. test case considers reaction R5 in equilibrium (N_s=10, N_{re}=5), while in the others it is defined as kinetic (N_s=9, N_{re}=4). N_{re} is the number of equilibrium reactions. In kinetic cases calcite concentration is not calculated in the speciation. The primary species of the traditional method (i.e., following definition 2.17) in both equilibrium and kinetic cases are 5 ($N_1 = N_s - N_{re} = 5$): Ca^{2+}, H^+, Cl^-, H_2O and HCO_3^-. These species coincide also with the transported components in the traditional method. For diluted solutions the activity of water is constant and equal to 1, therefore H_2O can be decoupled as a CAS. Thus, the proposed method transports only Ca^{2+}, H^+, Cl^-, and HCO_3^-. These four components are further reduced in the chemical step of CAL-Eq by the proposed method to 3 ($N_{1,nc} = 3$), to account for calcite as CAS. In the kinetic test cases, however, $N_{1,nc} = 4$. In CAL-Eq. the reduced primary species are Ca^{2+}, H^+ and Cl^-, while in the kinetic CAL cases Ca^{2+}, H^+, Cl^- and HCO_3^-.

The results of the proposed method were compared with the ones of the traditional method obtained with CHEPROO (Bea et al., 2009) coupled to TRACONF, a program to solve flow and transport in confined aquifers (Carrera et al., 1989). CHEPROO, in fact, solves SIA using the traditional speciation algorithm in the chemical step, without decoupling CAS.

The same convergence critera were adopted for both codes: $\varepsilon_{pr} = 10^{-7}$ and $\varepsilon_t = 10^{-10}$ for criteria (5.13) and (5.14), respectively, and $\varepsilon_{tr} = 10^{-3}$ for criterion (5.8).

First the quality of the results of the two methods was compared, to verify that the proposed method yields the correct results. Then we compared the performances of the two methods by means of the following three parameters

1. $\bar{p} = \dfrac{N_{PI}}{N_{\Delta t}}$: average number of Picard iterations per time step
2. $\bar{c} = \dfrac{N_{CI}}{N_{\Delta t} \cdot N_n}$: average number of chemical iterations per time step, per node
3. $\bar{t} = \dfrac{t_{chem}}{N_{\Delta t} \cdot N_n \cdot N_{PI}}$: average CPU time for a chemical step per time step, per node and Picard iteration

where N_{PI} and N_{CI} are the total number of Picard and chemical iterations in each simulation, respectively, and $N_{\Delta t}$ is the total number of time steps, 365 in the example. N_n is the number of nodes, 21 in the test case, and t_{chem} is the total CPU time spent in chemical steps in one simulation.

5.4 Results

Figure 5.2 show the pH results for the different test cases after 5 years. It can be noticed that the results of the proposed and the traditional method coincide.

In Table 5.3 the performance results are listed. The proposed method performs better in the CAL-Eq. case in terms of average Picard and chemical iterations, $\bar{p}$ and $\bar{c}$, respectively. This is due to the fact that the speciation of the chemical step is avoided after the first Picard iteration because the set of CAS does not change (i.e., calcite is always present in the system). In the CAL-1 case the values of $\bar{p}$ and $\bar{c}$ of traditional and proposed method are very similar, because calcite is not in equilibrium and the speciation in chemical step is needed at every Picard iteration. Number of iterations needed in the Picard loop are the same for CAL-2, but the number of iterations to solve the chemical step is higher for the proposed than the traditional method. In the last case, CAL-3, the proposed method performs worse in terms of both $\bar{p}$ and $\bar{c}$. For kinetic cases both methods become identical. Therefore, small differences in implementation and in programming language might be the cause of differences in performance of the two codes.

Finally, the comparison in terms of CPU time spent on the chemical steps is clear: the traditional method is much faster than the proposed method: from 4 (CAL-1) to almost 6 times (CAL-3). This could be due more to the way CHEPROO-TRACONF have been programmed than to a real advantage of the traditional method over the proposed method. In fact, both CHEPROO and TRACONF are very optimized codes.

Table 5.3: Performance results for traditional and proposed method for the different models. $\bar{t}$ is expressed in seconds.

	$\bar{p}$		$\bar{c}$		$\bar{t}$	
	Trad.	Prop.	Trad.	Prop.	Trad.	Prop.
CAL-Eq.	4.2	3.0	8.3	7.0	$1.00 \cdot 10^{-6}$	$5.01 \cdot 10^{-6}$
CAL-1	3.0	3.1	10.0	10.4	$1.53 \cdot 10^{-6}$	$6.18 \cdot 10^{-6}$
CAL-2	3.6	3.6	10.4	13.3	$1.23 \cdot 10^{-6}$	$5.25 \cdot 10^{-6}$
CAL-3	12.9	13.1	29.3	40.0	$9.28 \cdot 10^{-7}$	$5.55 \cdot 10^{-6}$

5.5 Conclusions

We proposed a method to decouple constant activity species, CAS, from the speciation of the chemical step in the sequential iteration approach to solve reactive transport. In order to decouple CAS it is necessary to calculate a vector of "reduced" components, $\mathbf{u}_{tot}^{red}$, from the vector of components, $\mathbf{u}_{tot}$, by means of an elimination matrix. Decoupling CAS from the speciation is advantageous for two reasons: first the system to be solved iteratively is smaller, and then the vector $\mathbf{u}_{tot}^{red}$, unlike $\mathbf{u}_{tot}$, does not change in one time step if the set of CAS is fixed. This means that the speciation needs to be solved only once per time step.

We applied the proposed method, implemented in CHEPROO++ and PROOST, and the traditional method, which does not decouple CAS, by means of a synthetic example representing calcite dissolution in a one-dimensional domain. First we verified that the two methods lead to the same results. Then we compared both methods in terms of average number of Picard and chemical iterations, and CPU time in the chemical steps.

Results show that when CAS are decoupled (i.e., in CAL-Eq.) the proposed method performs better in terms of average Picard and chemical iterations. On the contrary, in the kinetic cases the proposed method performs worse than the traditional method for the fastest kinetic cases. Since both methods are identical for kinetic cases, these differences might be due to differences in implementation and programming language. From the CPU time comparison it is clear that CHEPROO++ and PROOST perform worse than CHEPROO-TRACONF. Even though the proposed method performs better when CAS are decoupled, it would be more fair to compare a traditional method implemented in the same programming language (C++). As well, to demonstrate the advantages of the proposed method it is necessary to compare the two methods on more test cases, with more CAS, and maybe also with examples in which minerals appear/disappear from the domain.

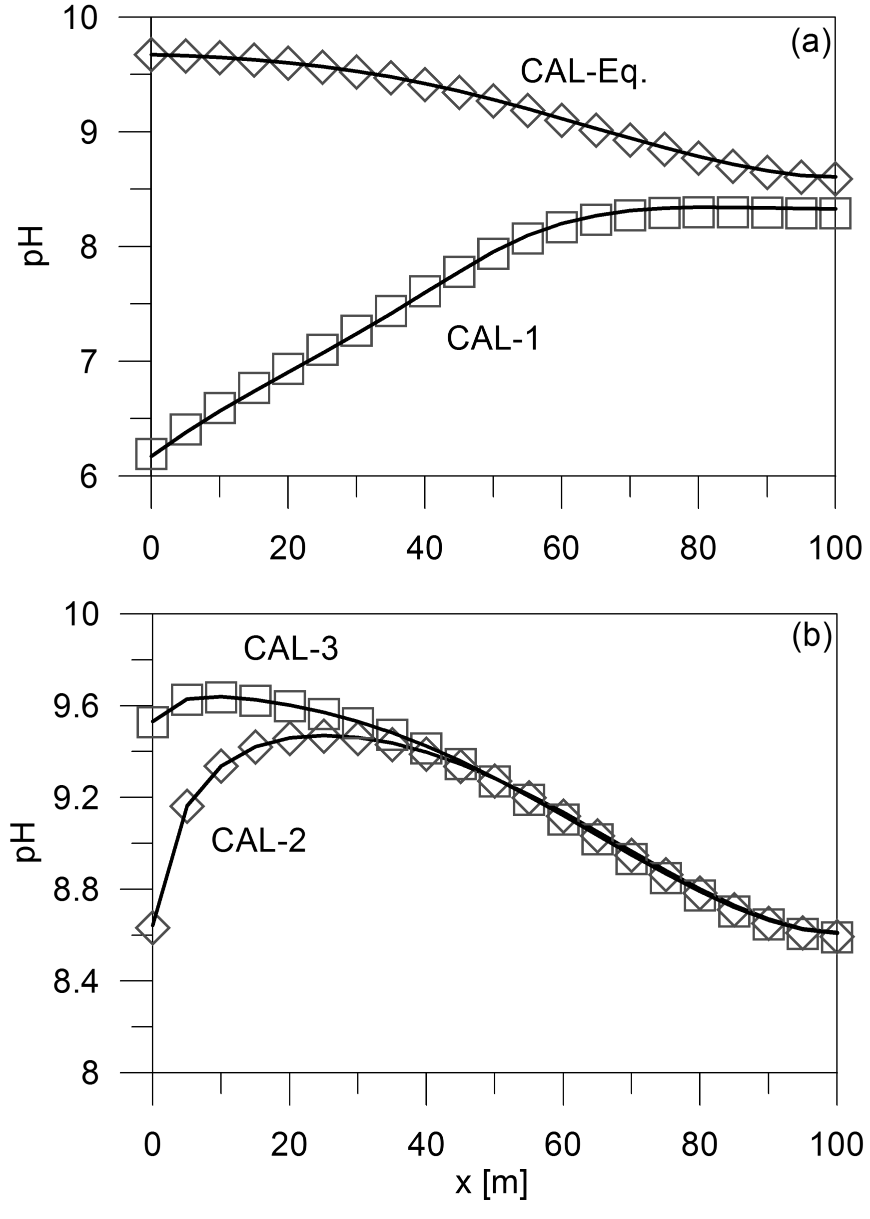


Figure 5.2: pH after 5 years obtained with the proposed method (line) and with the traditional method (symbols).

Chapter 6

Conclusions

This thesis presents new tools for hydrogeochemical modelling, such as speciation and mixing algorithms, that can be used for geochemical and reactive transport calculations.

In Chapter 3 we presented a method for geochemical speciation that accounts for redundant information while acknowledging errors in data. We define "redundant" the data that exceed the minimum set required to solve a geochemical speciation. We demonstrated that their use helps reducing estimation errors. Moreover, increasing the number of redundant data helps decreasing the estimation erros even further.

A mixing method that allows to calculate mixing proportions of a number of end-members in a sample was presented in Chapter 4. As well, quantities of reactions can be determined to account for reactive processes affecting the mixture. The novelty of the algorithm is the possibility of defining equilibrium conditions on the mixture. This can be useful for applications in which, for example, gas pressure data are available or carbonate rocks characterized by fast kinetics are present and equilibrium hypotheses can be made. We applied the algorithm to a mixing zone between freshwater and saltwater in a coastal aquifer of Mallorca (Spain). We used the mixing algorithm to characterize carbonate dissolution/precipitation processes and to quantify CO_2 production by means of microorganisms that could enhance the dissolution of carbonate rocks. The mixing proportions of freshwater and saltwater and the dissolution/precipitation of carbonate rocks were quantified. Results indicate that both CO_2 production and consumption need to occurr in order to explain the measurements. This might be due to limitations in the conceptual model.

These new algorithms have been implemented in CHEPROO++, an Object-Oriented library that offers methods for geochemical modelling. CHEPROO++ presents a definition of components that allows to decouple constant activity species (CAS) such as pure equilibrium minerals, pure gases with fixed partial pressure or proton when pH is constant. Decoupling CAS can be beneficial for solving the speciation because it decreases the size of the system to be solved iterativelly. We tested the advantage of decoupling CAS in the speciation necessary to solve the chemical step of the Sequential Iteration Approach (SIA). This components definition being conservative with respect to CAS it allows to solve the speciation only once per time step for every node, if the CAS set is fixed, no adsorption reactions are present and only equilibrium reactions are defined. We compared the proposed method with the traditional SIA for reactive transport by means of a synthetic example representing a one-dimensional domain for calcite dissolution in both equilibrium and kinetic conditions. The proposed method is advantageous for equilibrium dissolution (i.e., when calcite is decoupled as CAS) as it requires less iterations to converge. In the kinetic test cases, however, the proposed algorithm performs worse than the traditional method. As the two methods are identical for the kinetic cases, these differences could be due to differences in implementation and programming language of the reactive transport algorithms in the two code used for comparison.

Bibliography

Appelo, C., 1994. Cation and proton exchange, ph variations, and carbonate reactions in a freshening aquifer. Water Resources Research 30, 2793–2805.

Appelo, C.A.J., Postma, D., 2010. Geochemistry, groundwater and pollution. Taylor & Francis.

Back, W., Hanshaw, B.B., Herman, J.S., Van Driel, J.N., 1986. Differential dissolution of a pleistocene reef in the ground-water mixing zone of coastal yucatan, mexico. Geology 14, 137–140.

Back, W., Hanshaw, B.B., Pyle, T.E., Plummer, L.N., Weidie, A., 1979. Geochemical significance of groundwater discharge and carbonate solution to the formation of caleta xel ha, quintana roo, mexico. Water Resources Research 15, 1521–1535.

Ball, J.W., Nordstrom, D.K., 1991. User's manual for WATEQ4F, with revised thermodynamic data base and test cases for calculating speciation of major, trace, and redox elements in natural waters. US Geological Survey Denver, CO.

Bea, S., Carrera, J., Ayora, C., Batlle, F., Saaltink, M., 2009. Cheproo: A Fortran 90 object-oriented module to solve chemical processes in Earth Science models. Computers & Geosciences 35, 1098–1112.

Bear, J., 1999. Seawater intrusion in coastal aquifers concepts, methods and practices. Springer.

Bethke, C., 2008. Geochemical and biogeochemical reaction modeling. volume 543. Cambridge University Press Cambridge, UK.

Blum, J.D., Klaue, A., Nezat, C.A., Driscoll, C.T., Johnson, C.E., Siccama, T.G., Eagar, C., Fahey, T.J., Likens, G.E., 2002. Mycorrhizal weathering of apatite as an important calcium source in base-poor forest ecosystems. Nature 417, 729–731.

Bouchaou, L., Michelot, J., Vengosh, A., Hsissou, Y., Qurtobi, M., Gaye, C., Bullen, T., Zuppi, G., 2008. Application of multiple isotopic and geochemical tracers for investigation of recharge, salinization, and residence time of water in the Souss–Massa aquifer, southwest of Morocco. Journal of Hydrology 352, 267–287.

Brassard, P., Bodurtha, P., 2000. A feasible set for chemical speciation problems. Computers & Geosciences 26, 277–291.

Bundschuh, J., Zilberbrand, M., 2011. Geochemical Modeling of Groundwater, Vadose and Geothermal Systems. CRC Press.

Cabaniss, S.E., 1997. Propagation of Uncertainty in Aqueous Equilibrium Calculations : Non-Gaussian Output Distributions. Analytical Chemistry 69, 3658–3664.

Cabaniss, S.E., 1999. Uncertainty propagation in geochemical calculations: non-linearity in solubility equilibria. Applied Geochemistry 14, 255–262.

Carrayrou, J., Hoffmann, J., Knabner, P., Kräutle, S., De Dieuleveult, C., Erhel, J., Van Der Lee, J., Lagneau, V., Mayer, K.U., Macquarrie, K.T., 2010. Comparison of numerical methods for simulating strongly nonlinear and heterogeneous reactive transport problems - the momas benchmark case. Computational Geosciences 14, 483–502.

Carrayrou, J., Mose, R., Behra, P., 2002. New Efficient Algorithm for Solving Thermodynamic Chemistry. Environmental and Energy Engineering 48, 894–904.

Carrera, J., Galarza, G., Medina, A., 1989. Traconf. Programa de elementos finitos para la solución de las ecuaciones de flujo y transporte en acuíferos confinados, ETSI Caminos, Canales y Puertos, Universitat Politècnica de Catalunya .

Carrera, J., Vázquez-Suñé, E., Castillo, O., Sánchez-Vila, X., 2004. A methodology to compute mixing ratios with uncertain end-members. Water resources research 40.

Charlton, S.R., Macklin, C.L., Parkhurst, D., 1997. Phreeqci. A graphical user interface for the geochemical computer program phreeqc. US Geological Survey Water-Resources Investigations Report 9.

Christophersen, N., Hooper, R.P., 1992. Multivariate analysis of stream water chemical data: The use of principal components analysis for the end-member mixing problem. Water Resources Research 28, 99–107.

Corbella, M., Ayora, C., et al., 2003. Role of fluid mixing in deep dissolution of carbonates. Geologica Acta 1, 305–313.

Criscenti, L.J., Laniak, G.F., Erikson, R.L., 1996. Propagation of uncertainty through geochemical code calculations. Geochimica et Cosmochimica Acta 60, 3551–3568.

Currell, M.J., Cartwright, I., 2011. Major-ion chemistry, δ^{13}C and ^{87}Sr/^{86}Sr as indicators of hydrochemical evolution and sources of salinity in groundwater in the Yuncheng Basin, China. Hydrogeology Journal 19, 835–850.

Daniele, L., Vallejos, Á., Corbella, M., Molina, L., Pulido-Bosch, A., 2013. Hydrogeochemistry and geochemical simulations to assess water–rock interactions in complex carbonate aquifers: the case of Aguadulce (SE spain). Applied Geochemistry 29, 43–54.

De Gaspari, F., Saaltink, M.W., Carrera, J., Bea, S.A., Slooten, L., . An algorithm to calculate mixing ratios and reactions applied to mixing in carbonate coastal aquifers. Submitted.

De Gaspari, F., Saaltink, M.W., Carrera, J., Slooten, L., 2015. Use of redundant data to reduce estimation errors in geochemical speciation. Applied Geochemistry 55, 184–191.

De Gouw, J., Middlebrook, A., Warneke, C., Goldan, P., Kuster, W., Roberts, J., Fehsenfeld, F., Worsnop, D., Canagaratna, M., Pszenny, A., et al., 2005. Budget of organic carbon in a polluted atmosphere: Results from the new england air quality study in 2002. Journal of Geophysical Research: Atmospheres (1984–2012) 110.

De Simoni, M., Carrera, J., Sanchez-Vila, X., Guadagnini, A., 2005. A procedure for the solution of multicomponent reactive transport problems. Water resources research 41.

De Simoni, M., Sanchez-Vila, X., Carrera, J., Saaltink, M., 2007. A mixing ratios-based formulation for multicomponent reactive transport. Water Resources Research 43.

De Windt, L., Pellegrini, D., van der Lee, J., 2004. Reactive transport modelling of a spent fuel repository in a stiff clay formation considering excavation damaged zones. Radiochimica Acta/International journal for chemical aspects of nuclear science and technology 92, 841–848.

Dearlove, J., Longworth, G., Ivanovich, M., Kim, J., Delakowitz, B., Zeh, P., 1991. A study of groundwater-colloids and their geochemical interactions with natural radionuclides in gorleben aquifer systems. Radiochimca Acta 52, 83–90.

Denison, F.H., Garnier-Laplace, J., 2005. The effects of database parameter uncertainty on uranium (vi) equilibrium calculations. Geochimica et cosmochimica acta 69, 2183–2191.

Diersch, H., 1998. About the difference between the convective and the divergence form of the transport equation. FEFLOW Software White Papers 1.

Duan, Z., Li, D., 2008. Coupled phase and aqueous species equilibrium of the H_2O–CO_2–NaCl–$CaCO_3$ system from 0 to 250° c, 1 to 1000bar with nacl concentrations up to saturation of halite. Geochimica et Cosmochimica Acta 72, 5128–5145.

Duncan, T., Shaw, T.J., 2003. The mobility of rare earth elements and redox sensitive elements in the groundwater/seawater mixing zone of a shallow coastal aquifer. Aquatic geochemistry 9, 233–255.

Engel, A.S., Randall, K.W., 2011. Experimental evidence for microbially mediated carbonate dissolution from the saline water zone of the edwards aquifer, central texas. Geomicrobiology Journal 28, 313–327.

Gamazo, P., Bea, S., Saaltink, M., Carrera, J., Ayora, C., 2011. Modeling the interaction between evaporation and chemical composition in a natural saline system. Journal of Hydrology 401, 154–164.

Gamazo, P., Saaltink, M.W., Carrera, J., Slooten, L., Bea, S., 2012. A consistent compositional formulation for multiphase reactive transport where chemistry affects hydrodynamics. Advances in Water Resources 35, 83–93.

Garing, C., Luquot, L., Pezard, P., Gouze, P., 2013. Geochemical investigations of saltwater intrusion into the coastal carbonate aquifer of Mallorca, Spain. Applied Geochemistry 39, 1–10.

Gómez, J., Auqué, L., Gimeno, M., 2008. Sensitivity and uncertainty analysis of mixing and mass balance calculations with standard and pca-based geochemical codes. Applied Geochemistry 23, 1941–1956.

Gómez, J.B., Gimeno, M.J., Auqué, L.F., Acero, P., 2014. Characterisation and modelling of mixing processes in groundwaters of a potential geological repository for nuclear wastes in crystalline rocks of Sweden. Science of The Total Environment 468, 791–803.

Gupta, H.V., Clark, M.P., Vrugt, J.A., Abramowitz, G., Ye, M., 2012. Towards a comprehensive assessment of model structural adequacy. Water Resources Research 48.

Gustafsson, J.P., 2011. Visual MINTEQ 3.0 user guide. Royal Institute of Technology: Stockholm, Sweden.

Hammond, G., Lichtner, P., Mills, R., 2014. Evaluating the performance of parallel subsurface simulators: An illustrative example with pflotran. Water resources research 50, 208–228.

Han, D., Song, X., Currell, M.J., Yang, J., Xiao, G., 2014. Chemical and isotopic constraints on evolution of groundwater salinization in the coastal plain aquifer of Laizhou Bay, China. Journal of Hydrology 508, 12–27.

Helgeson, H.C., Kirkham, D.H., 1974. Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures; ii, debye-huckel parameters for activity coefficients and relative partial molal properties. American Journal of Science 274, 1199–1261.

Herman, J.S., Back, W., Pomar, L., 1985. Geochemistry of groundwater in the mixing zone along the east coast of Mallorca, Spain, in: Karst Water Resources (Proceedings of the Ankara-Antalya Symposium), pp. 467–479.

Howard, K., Lloyd, J., 1983. Major ion characterization of coastal saline ground waters. Groundwater 21, 429–437.

Huo, J.X., Song, H.Z., Wu, Z.W., 2014. Multi-component reactive transport in heterogeneous media and its decoupling solution. Journal of contaminant hydrology 166, 11–22.

Huyakorn, P.S., 2012. Computational methods in subsurface flow. Academic Press.

Jones, B., Vengosh, A., Rosenthal, E., Yechieli, Y., 1999. Geochemical investigations, in: Seawater Intrusion in Coastal Aquifers-Concepts, Methods and Practices. Springer, pp. 51–71.

Jørgensen, N.O., Andersen, M.S., Engesgaard, P., 2008. Investigation of a dynamic seawater intrusion event using strontium isotopes (^{87}Sr/^{86}Sr). Journal of Hydrology 348, 257–269.

Jurado, A., Vázquez-Suñé, E., Carrera, J., Tubau, I., Pujades, E., 2015. Quantifying chemical reactions by using mixing analysis. Science of The Total Environment 502, 448–456.

Jurado, A., Vàzquez-Suñé, E., Soler, A., Tubau, I., Carrera, J., Pujades, E., Anson, I., 2013. Application of multi-isotope data (O, D, C and S) to quantify redox processes in urban groundwater. Applied Geochemistry 34, 114–125.

Kalbus, E., Reinstorf, F., Schirmer, M., 2006. Measuring methods for groundwater–surface water interactions: a review. Hydrology and Earth System Sciences 10, 873–887.

Van der Kemp, W., Appelo, C., Walraevens, K., et al., 2000. Inverse chemical modeling and radiocarbon dating of palaeogroundwaters: the tertiary Ledo-Paniselian aquifer in Flanders, Belgium. Water Resources Research 36, 1277–1287.

Kim, Y., Lee, K.S., Koh, D.C., Lee, D.H., Lee, S.G., Park, W.B., Koh, G.W., Woo, N.C., 2003. Hydrogeochemical and isotopic evidence of groundwater salinization in a coastal aquifer: a case study in Jeju volcanic island, Korea. Journal of Hydrology 270, 282–294.

Knauss, K.G., Johnson, J.W., Steefel, C.I., 2005. Evaluation of the impact of co 2, co-contaminant gas, aqueous fluid and reservoir rock interactions on the geologic sequestration of co 2. Chemical geology 217, 339–350.

Krause, S., Aloisi, G., Engel, A., Liebetrau, V., Treude, T., 2014. Enhanced calcite dissolution in the presence of the aerobic methanotroph methylosinus trichosporium. Geomicrobiology Journal 31, 325–337.

Kräutle, S., 2011. The semismooth newton method for multicomponent reactive transport with minerals. Advances in Water Resources 34, 137–151.

Kräutle, S., Knabner, P., 2005. A new numerical reduction scheme for fully coupled multicomponent transport-reaction problems in porous media. Water resources research 41.

Kulik, D.A., Hummel, W., Lützenkirchen, J., Lefèvre, G., 2015. SI: Geochemical speciation codes and databases. Applied Geochemistry 55, 1–198.

Kulik, D.A., Wagner, T., Dmytrieva, S.V., Kosakowski, G., Hingerl, F.F., Chudnenko, K.V., Berner, U.R., 2013. Gem-selektor geochemical modeling package: revised algorithm and gems3k numerical kernel for coupled simulation codes. Computational Geosciences 17, 1–24.

Laaksoharju, M., Gascoyne, M., Gurban, I., 2008. Understanding groundwater chemistry using mixing models. Applied Geochemistry 23, 1921–1940.

Laaksoharju, M., Skårman, C., Skårman, E., 1999. Multivariate mixing and mass balance (M3) calculations, a new tool for decoding hydrogeochemical information. Applied Geochemistry 14, 861–871.

Lakshmanan, E., Kannan, R., Kumar, M.S., 2003. Major ion chemistry and identification of hydrogeochemical processes of ground water in a part of Kancheepuram district, Tamil Nadu, India. Environmental geosciences 10, 157–166.

Langmuir, C.H., Vocke Jr, R.D., Hanson, G.N., Hart, S.R., 1978. A general mixing equation with applications to Icelandic basalts. Earth and Planetary Science Letters 37, 380–392.

Le Gallo, Y., Bildstein, O., Brosse, E., 1998. Coupled reaction-flow modeling of diagenetic changes in reservoir permeability, porosity and mineral compositions. Journal of Hydrology 209, 366–388.

Lichtner, P.C., 1996. Continuum formulation of multicomponent-multiphase reactive transport. Reviews in Mineralogy and Geochemistry 34, 1–81.

Long, A.J., Valder, J.F., 2011. Multivariate analyses with end-member mixing to characterize groundwater flow: wind cave and associated aquifers. Journal of Hydrology 409, 315–327.

Mayer, K.U., Frind, E.O., Blowes, D.W., 2002. Multicomponent reactive transport modeling in variably saturated porous media using a generalized formulation for kinetically controlled reactions. Water Resources Research 38, 13–1.

Meeussen, J.C., 2003. Orchestra: An object-oriented framework for implementing chemical equilibrium models. Environmental science & technology 37, 1175–1182.

Merino, E., 1979. Internal consistency of a water analysis and uncertainty of the calculated distribution of aqueous species at 25 c. Geochimica et Cosmochimica Acta 43, 1533–1542.

Metz, V., Kienzler, B., Schüssler, W., 2003. Geochemical evaluation of different groundwater–host rock systems for radioactive waste disposal. Journal of Contaminant Hydrology 61, 265–279.

Millero, F.J., Morse, J., Chen, C.T., 1979. The carbonate system in the western Mediterranean Sea. Deep Sea Research Part A. Oceanographic Research Papers 26, 1395–1404.

Molins, S., Carrera, J., Ayora, C., Saaltink, M.W., 2004. A formulation for decoupling components in reactive transport problems. Water Resources Research 40.

Moore, W.S., 1999. The subterranean estuary: a reaction zone of ground water and sea water. Marine Chemistry 65, 111–125.

Morales-Casique, E., 2012. Mixing of groundwaters with uncertain end-members: case study in the Tepalcingo-Axochiapan aquifer, Mexico. Hydrogeology Journal 20, 605–613.

Nakaya, S., Uesugi, K., Motodate, Y., Ohmiya, I., Komiya, H., Masuda, H., Kusakabe, M., 2007. Spatial separation of groundwater flow paths from a multi-flow system by a simple mixing model using stable isotopes of oxygen and hydrogen as natural tracers. Water resources research 43.

Nitzsche, O., Meinrath, G., Merkel, B., 2000. Database uncertainty as a limiting factor in reactive transport prognosis. Journal of contaminant Hydrology 44, 223–237.

Ödegaard-Jensen, A., Ekberg, C., Meinrath, G., 2004. LJUNGSKILE: a program for assessing uncertainties in speciation calculations. Talanta 63, 907–916.

Panteleit, B., Hamer, K., Kringel, R., Kessels, W., Schulz, H., 2011. Geochemical processes in the saltwater–freshwater transition zone: comparing results of a sand tank experiment with field data. Environmental Earth Sciences 62, 77–91.

Parkhurst, D.L., 1997. Geochemical mole-balance modeling with uncertain data. Water Resources Research 33, 1957–1970.

Parkhurst, D.L., Appelo, C., et al., 1999. User's guide to PHREEQC (Version 2): A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. US Geological Survey Denver.

Paz-García, J.M., Johannessona, B., Ottosena, L.M., Ribeirob, A.B., Rodríguez-Marotoc, J.M., 2013. Computing multi-species chemical equilibrium with an algorithm based on the reaction extents. Computers & Chemical Engineering .

Phillips, D.L., Gregg, J.W., 2003. Source partitioning using stable isotopes: coping with too many sources. Oecologia 136, 261–269.

Pitkänen, P., Löfman, J., Koskinen, L., Leino-Forsman, H., Snellman, M., 1999. Application of mass-balance and flow simulation calculations to interpretation of mixing at äspö, sweden. Applied geochemistry 14, 893–905.

Plummer, L., 1975. Mixing of sea water with calcium carbonate ground water. Geological Society of America Memoirs 142, 219–236.

Plummer, L., Vacher, H., Mackenzie, F., Bricker, O., Land, L., 1976. Hydrogeochemistry of bermuda: A case history of ground-water diagenesis of biocalcarenites. Geological Society of America Bulletin 87, 1301–1316.

Plummer, L.N., Prestemon, E.C., Parkhurst, D.L., 1991. An interactive code (NETPATH) for modeling net geochemical reactions along a flow path. Department of the Interior, US Geological Survey.

Price, R.M., Herman, J.S., 1991. Geochemical investigation of salt-water intrusion into a coastal carbonate aquifer: Mallorca, Spain. Geological Society of America Bulletin 103, 1270–1279.

Rezaei, M., Sanz, E., Raeisi, E., Ayora, C., Vázquez-Suñé, E., Carrera, J., 2005. Reactive transport modeling of calcite dissolution in the fresh-salt water mixing zone. Journal of Hydrology 311, 282–298.

Rueedi, J., Purtschert, R., Beyerle, U., Alberich, C., Kipfer, R., 2005. Estimating groundwater mixing ratios and their uncertainties using a statistical multi parameter approach. Journal of hydrology 305, 1–14.

Saaltink, M.W., Ayora, C., Carrera, J., 1998. A mathematical formulation for reactive transport that eliminates mineral concentrations. Water Resources Research 34, 1649–1656.

Saaltink, M.W., Carrera, J., Ayora, C., 2001. On the behavior of approaches to simulate reactive transport. Journal of Contaminant Hydrology 48, 213–235.

Saaltink, M.W., Carrera, J., Olivella, S., 2004a. Mass balance errors when solving the convective form of the transport equation in transient flow problems. Water resources research 40.

Saaltink, M.W., Pifarré, F.B., Ayora, C., Carrera, J., Pastallé, S.O., 2004b. Retraso, a code for modeling reactive transport in saturated and unsaturated porous media. Geologica acta 2, 235.

Saaltink, M.W., Vilarrasa, V., De Gaspari, F., Silva, O., Carrera, J., Rötting, T.S., 2013. A method for incorporating equilibrium chemical reactions into multiphase flow models for CO_2 storage. Advances in Water Resources 62, 431–441.

Sanford, W.E., Konikow, L.F., 1989. Porosity development in coastal carbonate aquifers. Geology 17, 249–252.

Sanz, E., 2007. Brackish springs in coastal aquifers and the role of calcite dissolution by mixing waters. Ph.D. thesis. Department of Geotechnical Engineering and Geo-Sciences (ETCG), Technical University of Catalonia (UPC), BarcelonaTech.

Sanz, E., Ayora, C., Carrera, J., 2011. Calcite dissolution by mixing waters: geochemical modeling and flow-through experiments. Geologica Acta 9, 67–77.

Saravana Kumar, U., Sharma, S., Navada, S., Deodhar, A., 2009. Environmental isotopes investigation on recharge processes and hydrodynamics of the coastal sedimentary aquifers of Tiruvadanai, Tamilnadu State, India. Journal of Hydrology 364, 23–39.

Skov, H., Egelov, A.H., Granby, K., Nielsen, T., 1997. Relationships between ozone and other photochemical products at Ll. Valby, Denmark. Atmospheric Environment 31, 685–691.

Slomp, C.P., Van Cappellen, P., 2004. Nutrient inputs to the coastal ocean through submarine groundwater discharge: controls and potential impact. Journal of Hydrology 295, 64–86.

Slooten, L., Batlle, F., Carrera, J., 2010. An XML based problem solving environment for hydrological problems. XVIII International Conference on Water Resources .

Smith, S.D., Adams, N.W.H., Kramer, J.R., 1999. Resolving uncertainty in chemical speciation determinations. Geochimica et Cosmochimica Acta 63, 3337–3347.

Soler, J.M., Maeder, U.K., 2005. Interaction between hyperalkaline fluids and rocks hosting repositories for radioactive waste: reactive transport simulations. Nuclear science and engineering 151, 128–133.

Steefel, C., 2009. Crunchflow software for modeling multicomponent reactive flow and transport. users manual. Earth Sciences Division. Lawrence Berkeley, National Laboratory, Berkeley, CA. October , 12–91.

Steefel, C., Appelo, C., Arora, B., Jacques, D., Kalbacher, T., Kolditz, O., Lagneau, V., Lichtner, P., Mayer, K., Meeussen, J., et al., 2014. Reactive transport codes for subsurface environmental simulation. Computational Geosciences , 1–34.

Steefel, C., Yabusaki, S.B., 2000. OS3D/GIMRT software for modeling multicomponent-multidimensional reactive transport. Technical Report. Pacific Northwest National Lab., Richland, WA (US).

Steefel, C.I., Carroll, S., Zhao, P., Roberts, S., 2003. Cesium migration in hanford sediment: a multisite cation exchange model based on laboratory transport experiments. Journal of Contaminant Hydrology 67, 219–246.

Steefel, C.I., DePaolo, D.J., Lichtner, P.C., 2005. Reactive transport modeling: An essential tool and a new research approach for the earth sciences. Earth and Planetary Science Letters 240, 539–558.

Steefel, C.I., Lasaga, A.C., 1994. A coupled model for transport of multiple chemical-species and kinetic precipitation dissolution reactions with application to reactive flow in single-phase hydrothermal systems. American Journal of science , 529–592.

Steefel, C.I., MacQuarrie, K.T., 1996. Approaches to modeling of reactive transport in porous media. Reviews in Mineralogy and Geochemistry 34, 85–129.

Sun, L., Gui, H., 2014. Hydro-chemical evolution of groundwater and mixing between aquifers: a statistical approach based on major ions. Applied Water Science , 1–8.

Tubau, I., Vàzquez-Suñé, E., Jurado, A., Carrera, J., 2014. Using EMMA and MIX analysis to assess mixing ratios and to identify hydrochemical reactions in groundwater. Science of The Total Environment 470, 1120–1131.

Valder, J.F., Long, A.J., Davis, A.D., Kenner, S.J., 2012. Multivariate statistical approach to estimate mixing proportions for unknown end members. Journal of Hydrology 460, 65–76.

Van Der Lee, J., De Windt, L., 2001. Present state and future directions of modeling of geochemistry in hydrogeological systems. Journal of Contaminant Hydrology 47, 265–282.

Vázquez-Suñé, E., Carrera, J., Tubau, I., Sánchez-Vila, X., Soler, A., 2010. An approach to identify urban groundwater recharge. Hydrology and Earth System Sciences 14, 2085–2097.

Vengosh, A., Kloppmann, W., Marei, A., Livshitz, Y., Gutierrez, A., Banna, M., Guerrot, C., Pankratov, I., Raanan, H., 2005. Sources of salinity and boron in the Gaza strip: Natural contaminant flow in the southern Mediterranean coastal aquifer. Water Resources Research 41.

Walraevens, K., Cardenal-Escarcena, J., Van Camp, M., 2007. Reaction transport modelling of a freshening aquifer (Tertiary Ledo-Paniselian aquifer, Flanders-Belgium). Applied geochemistry 22, 289–305.

Walraevens, K., Van Camp, M., 2005. Advances in understanding natural groundwater quality controls in coastal aquifers, in: Groundwater and Saline Intrusion. Selected papers from the 18th SWIM meeting., 1st edn. IGGM, Madrid, pp. 449–463.

Weber, C.L., VanBriesen, J.M., Small, M.S., 2006. A stochastic regression approach to analyzing thermodynamic uncertainty in chemical speciation modeling. Environmental science & technology 40, 3872–3878.

Werner, A.D., Bakker, M., Post, V.E., Vandenbohede, A., Lu, C., Ataie-Ashtiani, B., Simmons, C.T., Barry, D.A., 2013. Seawater intrusion processes, investigation and management: Recent advances and future challenges. Advances in Water Resources 51, 3–26.

Wolery, T.J., 1983. EQ3NR: a computer program for geochemical aqueous speciation-solubility calculations. Users guide and documentation. Technical Report. Lawrence Livermore National Lab., CA (United States).

Wolery, T.J., 1992. EQ3NR, a Computer Program for Geochemical Aqueous Speciation-solubility Calculations: Theoretical Manual, User's Guide and Related Documentation (Version 7.0). Lawrence Livermore Laboratory, University of California.

Xu, T., Apps, J.A., Pruess, K., 2003. Reactive geochemical transport simulation to study mineral trapping for co2 disposal in deep arenaceous formations. Journal of Geophysical Research: Solid Earth (1978–2012) 108.

Xu, T., Sonnenthal, E., Spycher, N., Pruess, K., 2006. TOUGHREACT - A simulation program for non-isothermal multiphase reactive geochemical transport in variably saturated geologic media: Applications to geothermal injectivity and CO_2 geological sequestration. Computers & Geosciences 32, 145–165.

Xu, T., White, S.P., Pruess, K., Brimhall, G.H., 2000. Modeling of pyrite oxidation in saturated and unsaturated subsurface flow systems. Transport in porous media 39, 25–56.

Yeh, G.T., Tripathi, V.S., 1989. A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components. Water resources research 25, 93–108.

Zheng, C., Bennett, G.D., 1995. Applied contaminant transport modeling: theory and practice. Van Nostrand Reinhold.

APPENDIX A

Error definitions of Chapter 3

The errors $\boldsymbol{\varepsilon}$ allowed in the proposed method can be additive or multiplicative. Additive errors should lead to gaussian distributions, whereas multiplicative to lognormal distributions. Depending on the type of error, the function $\mathbf{g}(\mathbf{c})$ and the data $\mathbf{x}$ must be defined accordingly: aritmetic or logaritmic for additive and multiplicative errors, respectively.

Data used in speciation calculations are typically combinations of concentrations or activity values. The former, which we name balance equations, are linear combinations of concentrations representing, e.g., total concentrations, alkalinity, electrical conductivity, charge balance or TIC values. The latter are usually employed to fix pH values or to impose equilibrium with minerals or gases.

Distinguishing between these two types of data equations we can define

$$\boldsymbol{\varepsilon} = \mathbf{Bc} - \mathbf{x} \tag{A.1}$$

for the balance equations and

$$\varepsilon_i = \prod_{n=1}^{N_s} {a_n}^{L_{in}} - x_i \tag{A.2}$$

for each $i-th$ activity combination, respectively. There, $\mathbf{B}$ is a matrix of dimension $(N_b \times N_s)$, where N_b is the number of balance equations, that contains the coefficients of the balance equation: ionic charge for charge neutrality, specific (limiting) conductance for electrical conductivity, coefficients defining alkalinity or TIC, or component matrix elements for total concentration. $\mathbf{L}$ is a matrix of dimension $(N_a \times N_s)$ containing the coefficients of the activity for every species involved in the combination. N_a is the number of activity conditions imposed.

If we want to use a log-normal instead of a normal distribution of errors, one should use

$$\boldsymbol{\varepsilon} = \ln(\mathbf{Bc}) - \ln \mathbf{x} \tag{A.3}$$

for the balance equations and

$$\boldsymbol{\varepsilon} = \mathbf{L} \ln \mathbf{a} - \ln \mathbf{x} \tag{A.4}$$

for the activity combinations, respectively.

Appendix B

Jacobian calculation of Equation 3.8 in Chapter 3

The jacobian, containining the derivatives of $\boldsymbol{\varepsilon}$ with respect to the state variables $\ln \mathbf{c}_1$ at every step of the iterative method, can be calculated as

$$\begin{aligned} \frac{\partial \varepsilon_i}{\partial \ln c_{1,j}} &= \frac{\partial \varepsilon_i}{\partial c_{1,j}} \cdot c_{1,j} \\ &= \mathrm{B}_{1,ij} \cdot c_{1,j} + \sum_{l=1}^{N_2} \mathrm{B}_{2,il} \cdot \frac{\partial c_{2,l}}{\partial c_{1,j}} \cdot c_{1,j} \\ & i = 1, \ldots, N_b \\ & j = 1, \ldots, N_1 \end{aligned} \tag{B.1}$$

from the definition (A.1), whereas by means of definition (A.3) results

$$\begin{aligned} \frac{\partial \varepsilon_i}{\partial \ln c_{1,j}} &= \frac{\partial \ln z_i}{\partial \ln c_{1,j}} \\ &= \frac{1}{z_i} \cdot \frac{\partial z_i}{\partial \ln c_{1,j}} \\ &= \frac{1}{z_i} \cdot \left(\mathrm{B}_{1,ij} \cdot c_{1,j} + \sum_{l=1}^{N_2} \mathrm{B}_{2,il} \cdot \frac{\partial c_{2,l}}{\partial c_{1,j}} \cdot c_{1,j} \right) \\ & i = 1, \ldots, N_b \\ & j = 1, \ldots, N_1 \end{aligned} \tag{B.2}$$

being

$$z_i = \left(\sum_{m=1}^{N_1} \mathrm{B}_{1,im} \cdot c_{1,m} + \sum_{l=1}^{N_2} \mathrm{B}_{2,il} \cdot \frac{\partial c_{2,l}}{\partial c_{1,j}} \cdot c_{1,j} \right) \tag{B.3}$$

Matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ are the parts of matrix $\mathbf{B}$ relative to primary and secondary species, respectively, and the derivatives of secondary concentrations with respect to primary concentrations can be calculated considering that at every step of the iterative method the total derivative of $\mathbf{f}_{MAL}$ with respect to primary species concentrations is null

$$\frac{d\mathbf{f}_{MAL}}{d \ln \mathbf{c}_1} = \frac{\partial \mathbf{f}_{MAL}}{\partial \ln \mathbf{c}_1} + \frac{\partial \mathbf{f}_{MAL}}{\partial \ln \mathbf{c}_2} \frac{\partial \ln \mathbf{c}_2}{\partial \ln \mathbf{c}_1} = 0 \tag{B.4}$$

Those derivatives can be calculated by means of the following linear system

$$\frac{\partial \mathbf{f}_{MAL}}{\partial \ln \mathbf{c}_2}\frac{\partial \ln \mathbf{c}_2}{\partial \ln \mathbf{c}_1} = -\frac{\partial \mathbf{f}_{MAL}}{\partial \ln \mathbf{c}_1} \tag{B.5}$$

The conversion to $\partial \mathbf{c}_2/\partial \mathbf{c}_1$ is straightforward, recalling that $d \ln x/dx = 1/x$

$$\frac{\partial c_{2,i}}{\partial c_{1,j}} = \frac{c_{2,i}}{c_{1,j}}\frac{\partial \ln c_{2,i}}{\partial \ln c_{1,j}} \tag{B.6}$$

The derivatives of (A.4), remembering the definition of activity ($\mathbf{a} = \boldsymbol{\gamma} \cdot \mathbf{c}$) and that $\boldsymbol{\gamma} = f(\mathbf{c})$, read

$$\begin{aligned}\frac{\partial \varepsilon_i}{\partial \ln c_{1,j}} &= \mathrm{L}_{1,ij} + \sum_{m=1}^{N_1} \mathrm{L}_{1,im}\frac{\partial \ln \gamma_{1,mj}}{\partial \ln c_{1,j}} + \\ &+ \sum_{l=1}^{N_2} \mathrm{L}_{2,il}\left(\frac{\partial \ln \gamma_{2,lj}}{\partial \ln c_{1,j}} + \frac{\partial \ln c_{2,lj}}{\partial \ln c_{1,j}}\right) \\ & i = 1, \ldots, N_a \\ & j = 1, \ldots, N_1\end{aligned} \tag{B.7}$$

Matrices $\mathbf{L}_1$ and $\mathbf{L}_2$ are the parts of matrix $\mathbf{L}$ relative to primary and secondary species, respectively.

The derivatives of (A.2) with respect to the state variables can be calculated from

$$\begin{aligned}\frac{\partial \varepsilon_i}{\partial \ln c_{1,j}} &= \prod_{n=1}^{N_s} {a_n}^{\mathrm{L}_{in}} \cdot \\ &\cdot \left[\mathrm{L}_{1,ij} + \sum_{m=1}^{N_1} \mathrm{L}_{1,im}\frac{\partial \ln \gamma_{1,mj}}{\partial \ln c_{1,j}} + \sum_{l=1}^{N_2} \mathrm{L}_{2,il}\left(\frac{\partial \ln \gamma_{2,lj}}{\partial \ln c_{1,j}} + \frac{\partial \ln c_{2,lj}}{\partial \ln c_{1,j}}\right)\right] \\ & i = 1, \ldots, N_a \\ & j = 1, \ldots, N_1\end{aligned} \tag{B.8}$$

Appendix C

Calculation of $\partial\mathbf{c}/\partial\mathbf{u}$ of Equation 4.13 in Chapter 4

The derivatives $\partial\mathbf{c}/\partial\mathbf{u}$ can be calculated from the results of a speciation calculation. A standard geochemical speciation calculation requires to solve a non-linear system like the following

$$\mathbf{g}(\mathbf{u}) = \begin{cases} \mathbf{g}_1(\mathbf{u}) = \mathbf{U}\cdot\mathbf{c} - \mathbf{u} = \mathbf{0} \\ \mathbf{g}_2(\mathbf{u}) = \mathbf{S}_e \log\mathbf{a} - \log\mathbf{K} = \mathbf{0} \end{cases} \tag{C.1}$$

where $\mathbf{U}$ is the component matrix ($N_c \times N_s$), $\mathbf{c}$ is the concentration vector (N_s) and $\mathbf{u}$ is a known vector of components (N_c). N_c and N_s are the numbers of components and of species, respectively. Normally $\mathbf{u}$ contains measured values of total concentrations of elements, alkalinity or electrical conductivity. In our method, $\mathbf{u} = \mathbf{X}_u\boldsymbol{\lambda} + \mathbf{U}\mathbf{S}_k^t\mathbf{q}_k$ (see equation 4.7). Vector $\mathbf{a}$ contains the activities of the N_s species, $\mathbf{S}_e$ is a matrix ($N_{re} \times N_s$) with the stoichiometric coefficients of the equilibrium reactions (N_{re}) and $\mathbf{K}$ is a vector (N_{re}) of equilibrium constants. Various algorithms exist in order to solve system (C.1) (De Gaspari et al., 2015; Paz-García et al., 2013; Bea et al., 2009; Parkhurst et al., 1999, amongst others). Independently on the method used, once system (C.1) has been solved, i.e., when $\mathbf{c}$ has been calculated, $\partial\mathbf{c}/\partial\mathbf{u}$ can be evaluated knowing that

$$\frac{d\mathbf{g}}{d\mathbf{u}} = \frac{\partial\mathbf{g}}{\partial\mathbf{u}} + \frac{\partial\mathbf{g}}{\partial\mathbf{c}}\cdot\frac{\partial\mathbf{c}}{\partial\mathbf{u}} = \mathbf{0} \tag{C.2}$$

This means that $\partial\mathbf{c}/\partial\mathbf{u}$ can be calculated from the following system:

$$\frac{\partial\mathbf{g}}{\partial\mathbf{c}}\cdot\frac{\partial\mathbf{c}}{\partial\mathbf{u}} = -\frac{\partial\mathbf{g}}{\partial\mathbf{u}} \tag{C.3}$$

Given (C.1), $\partial\mathbf{g}/\partial\mathbf{u}$ ($N_s \times N_1$) reads

$$\frac{\partial\mathbf{g}}{\partial\mathbf{u}} = \begin{pmatrix} \partial\mathbf{g_1}/\partial\mathbf{u} \\ \partial\mathbf{g_2}/\partial\mathbf{u} \end{pmatrix} = \begin{pmatrix} \mathbf{I} \\ \mathbf{0} \end{pmatrix} \tag{C.4}$$

while $\partial\mathbf{g}/\partial\mathbf{c}$ ($N_s \times N_s$) reads

$$\frac{\partial\mathbf{g}}{\partial\mathbf{c}} = \begin{pmatrix} \partial\mathbf{g_1}/\partial\mathbf{c} \\ \partial\mathbf{g_2}/\partial\mathbf{c} \end{pmatrix} = \begin{pmatrix} \mathbf{U} \\ \mathbf{G}^* \end{pmatrix} \tag{C.5}$$

$\mathbf{G}^*$ ($N_{re} \times N_s$) can also be written as

$$\mathbf{G}^* = \begin{pmatrix} \dfrac{\partial\mathbf{g_2}}{\partial\mathbf{c}_1} & \dfrac{\partial\mathbf{g_2}}{\partial\mathbf{c}_2} \end{pmatrix} \tag{C.6}$$

where $\partial\mathbf{g_2}/\partial\mathbf{c}_1$ and $\partial\mathbf{g_2}/\partial\mathbf{c}_2$ can be evaluated from definition (C.1), remembering that $\mathbf{a} = \mathbf{c}\cdot\boldsymbol{\gamma}$ and differentiating between primary and secondary species

$$\frac{\partial g_{2,i}}{\partial c_{1,j}} = \frac{1}{c_{1,j}} \left(S_{e1,ij} + \sum_{m=1}^{N_1} S_{e1,im} \frac{\partial \log \gamma_{1,m}}{\partial \log c_{1,j}} + \sum_{n=1}^{N_{re}} S_{e2,in} \frac{\partial \log \gamma_{2,n}}{\partial \log c_{1,j}} \right) \tag{C.7}$$

$$\frac{\partial g_{2,i}}{\partial c_{2,j}} = \frac{1}{c_{2,j}} \left(\sum_{m=1}^{N_1} S_{e1,im} \frac{\partial \log \gamma_{1,m}}{\partial \log c_{2,j}} + S_{e2,ij} + \sum_{n=1}^{N_{re}} S_{e2,in} \frac{\partial \log \gamma_{2,n}}{\partial \log c_{2,j}} \right) \tag{C.8}$$

$\mathbf{S}_{e1}$ and $\mathbf{S}_{e2}$ are the parts of the stoichiometric matrices relative to primary and secondary species, of dimension $(N_{re} \times N_1)$ and $(N_{re} \times N_{re})$, respectively.

Appendix D
Main CHEPROO++ methods

Glossary

- N_s: number of species
- N_{re} : number of equilibrium reactions
- N_1 : number of (N_s-N_{re}) primary species
- N_2 : number of N_{re} secondary species
- N_{CAS} : number of constant activity species
- N_{ph} : number of phases
- $N_{1,nc}$: number of non-constant (reduced) primary species (N_s-N_{re}- N_{CAS})
- $\mathbf{a}$: vector of N_s activities
- $\mathbf{c}$: vector of N_s concentrations
- $\boldsymbol{\gamma}$: vector of N_s activity coefficients
- $\mathbf{c}_1$: vector of N_1 primary species concentrations
- $\mathbf{c}_{1,nc}$: vector of $N_{1,nc}$ reduced primary species concentrations
- $\mathbf{c}_2$: vector of N_2 secondary species concentrations
- $\mathbf{S}_e$: equilibrium stoichiometric matrix ($N_{re} \times N_s$)
- $\mathbf{S}_{e1}$: part of equilibrium stoichiometric matrix relative to primary species ($N_{re} \times N_1$)
- $\mathbf{S}_{e2}$: part of equilibrium stoichiometric matrix relative to secondary species ($N_{re} \times N_{re}$)
- $\mathbf{S}_{e1,nc}$: part of equilibrium stoichiometric matrix relative to reduced primary species ($N_{re} \times N_{1,nc}$)
- $\mathbf{S}_{e1,CAS}$: part of equilibrium stoichiometric matrix relative to reduced primary species ($N_{re} \times N_{CAS}$)
- $\mathbf{c}_{aq}$: vector of aqueous concentrations
- $\mathbf{u}_{aq}$: vector of aqueous components
- $\mathbf{u}_{tot}$: vector of total components
- $\mathbf{U}_{tot}$: component matrix relative to all species in all phases

- $\mathbf{U}_{aq}$: component matrix relative to aqueous species
- $\mathbf{U}_{CAS}$: component matrix relative to constant activity species
- $[\,]_{\alpha}$: general phase subscript
- $[\,]_{aq}$: liquid phase subscript
- $[\,]_{gas}$: gas phase subscript
- $[\,]_{min}$: mineral phase subscript
- $[\,]_{surf}$: gas phase subscript
- θ_α: volumetric content of the $\alpha - th$ phase

Methods

Some of the main methods that CHEPROO++ provides, besides the ones presented in chapters 3 and 4 are:

1. Given $\mathbf{u}^*_{aq}$ of a local chemical system, speciate
2. Given $\mathbf{u}^*_{tot}$ of a local chemical system, speciate
3. Given $\mathbf{c}_1$ of a local chemical system, evaluate $\mathbf{c}_2$ and $\partial\mathbf{c}_2/\partial\mathbf{c}_1$
4. Given n waters, mix them in (another) local chemical system

D.1 Given $\mathbf{u}^*_{aq}$ of a local chemical system, speciate

Equations and definitions

Performing the speciation requires to solve the following system of equations

$$\begin{cases} \mathbf{f}_t = \theta_{aq}\mathbf{U}_{aq}\mathbf{c}_{aq} - \theta^*_{aq}\mathbf{u}^*_{aq} = 0 \\ \mathbf{f}_{MAL} = \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \end{cases} \tag{D.1}$$

in order to evaluate $\mathbf{c}_{aq}$ and θ_{aq}.

$\mathbf{f}_t$ is a mass balance corresponding to the definition of components and $\mathbf{f}_{MAL}$ represents the mass action laws, described in Chapter 2.

The approach to solve this system consists of dividing the set of species in primary and secondary species. Since it is always possible to express the activities of the secondary species as explicit function of the activities of the primary species ($\mathbf{a}_2 = f(\mathbf{a}_1)$), it is possible to calculate $\mathbf{c}_2 = f(\mathbf{c}_1)$ from $\mathbf{f}_{MAL} = 0$, and then subsituting it into f_t to evaluate the concentration of primary species $\mathbf{c}_1$. We assume that the constant activity species have been decoupled as explained in Chapter 2 - Section 2.2, so that the concentrations of primary species that we calculate are the ones of the "reduced" primary species, $\mathbf{c}_{1,nc}$.

Given the non-linear relationship between $\mathbf{c}_2$ and $\mathbf{c}_{1,nc}$ an iterative method is required to solve $\mathbf{f}_t = 0$. We use Newton-Raphson, which requires to approximate linearly $\mathbf{f}_t$, so that

$$\mathbf{f}_t = 0 \Leftrightarrow \mathbf{J}^i(\mathbf{c}^{i+1}_{1,nc} - \mathbf{c}^i_{1,nc}) = -\mathbf{f}^i_t \tag{D.2}$$

Where $\mathbf{J}^i$ is the Jacobian matrix at iteration i. By means of Eq. (D.2) it is possible to calculate $\mathbf{c}_{1,nc}^{i+1}$.

The Jacobian matrix is defined as follows:

$$\mathbf{J}^i = \frac{d\mathbf{f}_t^i}{d\mathbf{c}_{1,nc}^i} = \mathbf{U}_{1,aq} + \mathbf{U}_{2,aq}\frac{\partial \mathbf{c}_2^i}{\partial \mathbf{c}_{1,nc}^i} \tag{D.3}$$

$\mathbf{U}_{1,aq}$ and $\mathbf{U}_{2,aq}$ are the parts of the component matrix relative to primary and secondary species, respectively. The derivatives of secondary species with respect to primary species are evaluated from the solution of $\mathbf{f}_{MAL} = 0$ (see Section D.3 for details). The residual is defined as

$$-\mathbf{f}_t^i = \mathbf{u}_{aq}^* - \mathbf{U}_{1,aq}\mathbf{c}_{1,nc}^i - \mathbf{U}_{2,aq}\mathbf{c}_2^i \tag{D.4}$$

The algorithm is different depending on whether the liquid volume content θ_{aq} is constant or not. We assume that is constant and therefore can be eliminated from the system of equations.

Algorithm

- Input: $\mathbf{u}_{aq}^*$, P, T
- Output: $\mathbf{c}_{aq}$

Description of the algorithm:

1. Get initial guess $\mathbf{c}_{1,\mathrm{nc}}^0$
2. Main Newton-Raphson loop to evaluate $\mathbf{c}_{1,nc}$
    (a) Compute $\mathbf{c}_2^i = f(\mathbf{c}_{1,nc}^i)$ from $\mathbf{f}_{MAL} = 0$ (see Section D.3), together with $\partial\mathbf{c}_2/\partial\mathbf{c}_1$
    (b) Compute residual $-\mathbf{f}_t^i$
    (c) Compute Jacobian $\mathbf{J}^i$
    (d) Solve the system and evaluate $\Delta\mathbf{c}_{1,nc}^i$
    (e) Update the solution $\mathbf{c}_{1,nc}^{i+1}=\mathbf{c}_{1,nc}^i+\Delta\mathbf{c}_{1,nc}^i$
    (f) Check convergence:
        IF YES: convert $\mathbf{c}$ to molality and save solution
        IF NO: Go to (a) and perform another iteration

D.2 Given $\mathbf{u}^*_{tot}$ of a local chemical system, speciate

Equations and definitions

Performing this speciation requires to solve the following system of equations

$$\begin{cases} \mathbf{f}_t = \sum_\alpha \theta_\alpha \mathbf{U}_\alpha \mathbf{c}_\alpha - \mathbf{u}_{tot}^* = 0 \\ \mathbf{f}_{MAL} = \mathbf{S}_e \log \mathbf{a} - \log \mathbf{k} = 0 \end{cases} \tag{D.5}$$

in order to evaluate $\mathbf{c}_\alpha$, $\alpha = 1,...,N_{ph}$.

$\mathbf{f}_t$ contains mass balance equations corresponding to the definition of components and $\mathbf{f}_{MAL}$ represents the mass action laws, described in Chapter 2. .

To solve this system we use Newton-Raphson method, substituting the secondary species concentrations in $\mathbf{f}_t$.

$$\mathbf{f}_t = 0 \Leftrightarrow \mathbf{J}^i(\mathbf{c}_{1,nc}^{i+1} - \mathbf{c}_{1,nc}^{i}) = -\mathbf{f}_t^i \tag{D.6}$$

Where $\mathbf{J}^i$ and $\mathbf{c}_{1,nc}^i$ are the Jacobian matrix and the vector of concentrations of "reduced" primary species at the iteration i. By means of Eq. (D.6) it is possible to calculate $\mathbf{c}_{1,nc}^{i+1}$.

The Jacobian matrix is defined as follows:

$$\mathbf{J}^i = \frac{d\mathbf{f}_t^i}{d\mathbf{c}_{1,nc}^i} = \sum_\alpha \theta_\alpha \mathbf{U}_{\alpha 1} + \sum_\alpha \theta_\alpha \mathbf{U}_{\alpha 2} \frac{\partial \mathbf{c}_2^i}{\partial \mathbf{c}_{1,nc}^i} \tag{D.7}$$

$\mathbf{U}_{\alpha 1}$ and $\mathbf{U}_{\alpha 2}$ are the parts of the component matrix of a phase α relative to primary and secondary species, respectively. Where the derivatives of secondary species with respect to primary species are evaluated from the solution of $\mathbf{f}_{MAL} = 0$ (see Section 1.3 for details). The residual is defined as

$$-\mathbf{f}_t^i = \mathbf{u}_{tot}^* - \sum_\alpha \theta_\alpha \mathbf{U}_{\alpha 1} \mathbf{c}_{1,nc}^i \sum_\alpha \theta_\alpha \mathbf{U}_{\alpha 2} \mathbf{c}_2^i \tag{D.8}$$

The algorithm is different depending on whether the liquid volume content θ_{aq} is constant or not. By now, we assume that is constant and therefore can be eliminated from the system of equations.

Algorithm

- Input: $\mathbf{u}_{aq}^*$, θ_α P, T
- Output: $\mathbf{c}_{aq}$

Description of the algorithm:

1. Get initial guess $\mathbf{c}_{\mathrm{1,nc}}^0$
2. Main Newton-Raphson loop to evaluate $\mathbf{c}_{1,nc}$
   (a) Compute $\mathbf{c}_2^i = f(\mathbf{c}_{1,nc}^i)$ from $\mathbf{f}_{MAL} = 0$ (see Section D.3), together with $\partial \mathbf{c}_2 / \partial \mathbf{c}_1$
   (b) Compute residual $-\mathbf{f}_t^i$
   (c) Compute Jacobian $\mathbf{J}^i$
   (d) Solve the system and evaluate $\Delta \mathbf{c}_{1,nc}^i$
   (e) Update the solution $\mathbf{c}_{1,nc}^{i+1} = \mathbf{c}_{1,nc}^i + \Delta \mathbf{c}_{1,nc}^i$
   (f) Check convergence:
      IF YES: convert $\mathbf{c}$ to molality and save solution
      IF NO: Go to (a) and perform another iteration

D.3 Given $\mathbf{c}_1$ of a local chemical system, evaluate $\mathbf{c}_2$ and $\partial \ln \mathbf{c}_2/\partial \ln \mathbf{c}_1$

Equations and definitions

It is possible to calculate $\mathbf{c}_2$ from $\mathbf{c}_1$ from the mass action laws, already defined in Chapter 2, equation (2.24) which relate activities of secondary and primary species at equilibrium:

$$\mathbf{f}_{MAL} = \mathbf{S}_{e1,nc} \log \mathbf{a}_{1,nc} + \mathbf{S}_{e1,CAS} \log \mathbf{a}_{1,CAS} + \mathbf{S}_{e2} \log \mathbf{a}_2 - \log \mathbf{k} = 0 \tag{D.9}$$

Equation (D.9) is not linear, therefore an iterative process is required to solve it. In CHEPROO++ Picard and Newton-Raphson methods are implemented. The first one is quite straightforward and it is almost always applicable, except when concentrations have a strong effect on activity coefficients, e.g. for very concentrated solutions. In those cases Newton-Raphson is better, as it accounts for the derivative of activity coefficients with respect to concentrations.

Newton-Raphson method requires to approximate linearly $\mathbf{f}_{MAL}$ so that

$$\mathbf{f}_{MAL} = 0 \Leftrightarrow \mathbf{J}^i(\ln \mathbf{c}_2^{i+1} - \ln \mathbf{c}_2^i) = -\mathbf{f}_{MAL}^i \tag{D.10}$$

Where $\mathbf{J}^i$and ln$\mathbf{c}_2^i$ are the Jacobian matrix and the vector of logarithmic secondary species concentrations at the iteration i.

Reminding that $\mathbf{a} = \boldsymbol{\gamma} \cdot \mathbf{c}$ and that $\boldsymbol{\gamma} = f(\mathbf{c})$, the Jacobian matrix can be calculated as follows:

$$\mathbf{J}^i = \frac{\partial \mathbf{f}_{MAL}^i}{\partial \text{ln}\mathbf{c}_2^i} = \mathbf{I} - \mathbf{S}_{e1}^* \frac{\partial \text{ln}\, \boldsymbol{\gamma}_1^i}{\partial \, \text{ln}\mathbf{c}_2^i} + \frac{\partial \text{ln}\, \boldsymbol{\gamma}_2^i}{\partial \, \text{ln}\mathbf{c}_2^i} \tag{D.11}$$

Note that the derivatives of $\mathbf{a}_{1,CAS}$ with respect to $\ln \mathbf{c}_2$ are null, as $\mathbf{a}_{1,CAS}$ are constant. Given $\ln \mathbf{c}_2^i$, by means of Eq. (D.10) it is possible to calculate $\ln \mathbf{c}_2^{i+1}$ at every iteration until convergence.

When convergence is reached, $\partial \ln \mathbf{c}_2/\partial \ln \mathbf{c}_1$ can be calculated from

$$\frac{d\mathbf{f}_{MAL}}{d\text{ln}\mathbf{c}_1} = \frac{\partial \mathbf{f}_{MAL}}{\partial \text{ln}\mathbf{c}_1} + \frac{\partial \mathbf{f}_{MAL}}{\partial \text{ln}\mathbf{c}_2} \frac{\partial \text{ln}\, \mathbf{c}_2}{\partial \ln \mathbf{c}_1} = 0$$

$$\frac{\partial \mathbf{f}_{MAL}}{\partial \text{ln}\mathbf{c}_2} \frac{\partial \text{ln}\, \mathbf{c}_2}{\partial \ln \mathbf{c}_1} = -\frac{\partial \mathbf{f}_{MAL}}{\partial \text{ln}\mathbf{c}_1} \tag{D.12}$$

Note that the system matrix in (D.12) coincides with the Jacobian defined in (D.11), while the right hand side can be defined as

$$\frac{\partial \mathbf{f}_{MAL}}{\partial \text{ln}\mathbf{c}_1} = -\mathbf{S}_{e1}^* - \mathbf{S}_{e1}^* \frac{\partial \text{ln}\, \boldsymbol{\gamma}_1}{\partial \, \text{ln}\mathbf{c}_1} + \frac{\partial \text{ln}\, \boldsymbol{\gamma}_2}{\partial \, \text{ln}c_1} \tag{D.13}$$

Algorithm

- Input: $\mathbf{c}_1$, P, T
- Output: $\mathbf{c}_2$, $\frac{d\mathbf{c}_2}{d\mathbf{c}_1}$

Description of the algorithm:

1. Chech which iterative method is used:
2. IF Newton-Raphson
    (a) Transform $\mathbf{c}_1$ to $\ln \mathbf{c}_1$

(b) Get initial guess of $\ln \mathbf{c}_2{}^0$

(c) Compute $\boldsymbol{\gamma}^0$

(d) Main Newton-Raphson loop:

```
i. Compute Jacobian $\mathbf{J}^i$
ii. Compute residual $-\mathbf{f}_{MAL}$
iii. Solve the system and evaluate $\Delta \ln \mathbf{c}_2^i$
iv. Update the solution $\ln \mathbf{c}_2^{i+1}=\ln \mathbf{c}_2^i+\Delta \ln \mathbf{c}_2^i$
v. Evaluate $\boldsymbol{\gamma}^{i+1}$
vi. Check convergence:
        IF YES: convert $\mathbf{c}$ to molality and save solution
        IF NO: go to (i) and perform another iteration
```

3. IF Picard

(a) Get initial guess of $\mathbf{c}_2{}^0$

(b) Compute $\boldsymbol{\gamma}^0$

(c) Main Picard loop:

```
i. Evaluate $\mathbf{c}_2^{i+1}$ from $\mathbf{f}_{MAL}$
ii. Evaluate $\boldsymbol{\gamma}^{i+1}$
iii. Check convergence:
        IF YES: convert $\mathbf{c}$ to molality and save solution
        IF NO: go to (i) and perform another iteration
```

D.4 Given n waters, mix them in (another) local chemical system

Equations and definitions

We want to evaluate $\mathbf{c}_{aq}$ of the mixed water (${\mathbf{c}^m}_{aq}$) by speciating it from ${\mathbf{u}^m}_{aq}$, defined as the linear combination of $\mathbf{c}_{aq,i}$, the aqueous concentrations of the initial waters:

$$
{\mathbf{u}^m}_{aq} = \frac{\sum_i \chi_i {\mathbf{U}^m}_{aq} \mathbf{c}_{aq,i}}{\chi_{tot}} \qquad i = 1, ..., n \tag{D.14}
$$

Where χ_i is the volume of the $i-th$ initial water and $\sum_i \chi_i/\chi_{tot}=1$. ${\mathbf{U}^m}_{aq}$ is the aqueous component matrix of the mixed water and concentrations are expressed in molarity (mol/Vol_{aq}).

We suppose that the volumetric content of the liquid phase remain constant during the mixing process, therefore it doesn't explicitly appear in Eq. (D.14).

Algorithm

- Input: χ_i, $\mathbf{c}_{aq,i}$, $P_i, T_i \quad i = 1, ..., n$
- Output: ${\mathbf{c}^m}_{aq}$

Description of the algorithm:

1. Evaluate $\chi_{tot} = \sum_i \chi_i$
2. Start loop over initial waters
   (a) Convert molality to molarity
   (b) Update $\mathbf{u}^m{}_{aq}$ as defined in Eq. (D.14)
3. Speciate mixed water from $\mathbf{u}^m{}_{aq}$ (see Section 1.1)
4. Option: check how much of minerals dissolve/precipitate

Appendix E
CHEPROO++ manual

The speciation algorithms and functionalities implemented in CHEPROO++ have been described in the previous chapters of the thesis. it is explained how to download it, install it and use it on a PC (Windows/Linux machines). This manual is organised in the following sections:

- How to install CHEPROO++
- Programs that use CHEPROO++
- How to build a program with CHEPROO++
- Input file description

E.1 How to install CHEPROO++

When this manual was written, CHEPROO++ was coupled with Proost – a program developed in the group to solve flow and conservative transport. This means that when you download Cheproo++, you download Proost too. However, this does not mean that you can only use CHEPROO++ with Proost. In the following sections it is explained how to use CHEPROO++ as "standalone" program (i.e., without Proost).

E.1.1) Programs and libraries required to install Cheproo++

In order to download and compile CHEPROO++ a few programs need to be installed and some configurations need to be changed in your computer, as explained below.

- **TortoiseSVN**. It's a revision control / version control / source control software for Windows explorer. It is also available for Linux or Mac under the name of "SmartSVN". Download it from the website http://tortoisesvn.net/downloads.html and install it.

- **Qt**. Qt provides libraries for XML parsing for C++. Download it from http://www.qt.io/download-open-source/#. Choose the LGPL licence. There are different versions for Windows. Pick the one called “Qt x.x.x for Windows (VS 2010)”, being careful with the Visual Studio version that you have (in this case, VS2010 for Visual Studio 2010). Install Qt. Then install the "visual studio add-in" from the same web page.
- After the setup is completed, go to your computer properties. Click on “advanced system settings”. Open the environmental variables window. Add a system variable called QTDIR with value "c:\qt\x.x.x" where x.x.x is the version number of your Qt distribution (with default installation).
- In the environmental variables window, add %QTDIR%\bin to the start of the system path (use a semicolon to separate it from the existing entries).
- **Clapack**. Download it from http://www.netlib.org/clapack/index.html. Choose the version for CMake (i.e., it has cmake in the name). When this document was written its name was "clapack-3.2.1-CMAKE.tgz". Unpack this compressed file in a folder outside CHEPROO++ directory.
- **Visual Studio** (for Windows users). UPC student can download it from the intranet.
- **CMake**. CMake can be downloaded from http://www.cmake.org/download/. We have been working with the 2.8.7 version of CMake. Download it and install it. Choose all the default settings in the setup process.

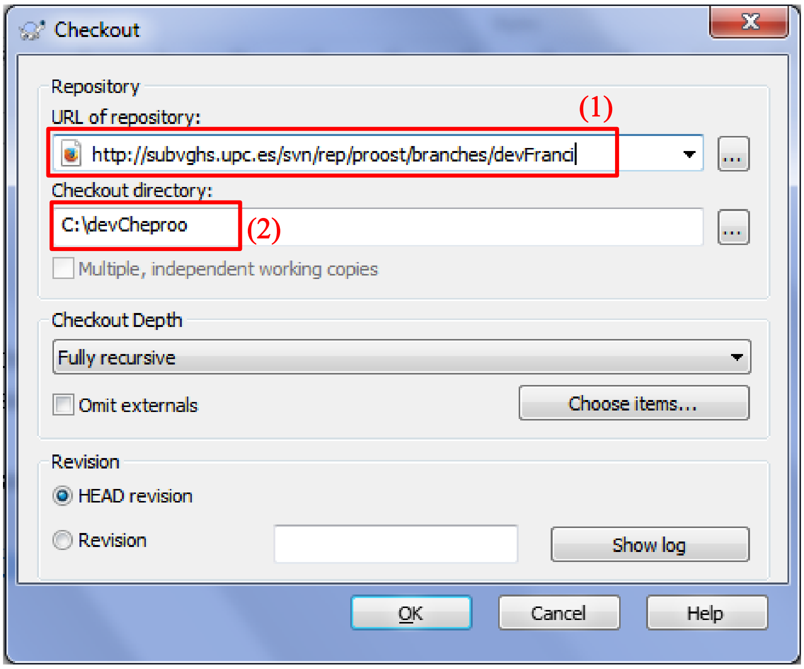
Now that all the programs and libraries have been downloaded and installed, you can follow the steps below to download CHEPROO++ and compile it.

E.1.2) CHEPROO++ setup

CHEPROO++ can be downloaded from the following repository:

http://subvghs.upc.es/svn/rep/proost/branches/devFranci

In order to download it to your computer, create a folder in which you will copy the library and create the project (i.e., “C:\devCheproo”). Right click on this folder and choose “SVN Checkout…”. The following window (Figure 1) will appear in which you have to copy (1) the repository URL and (2) the directory you just created – in which the library will be copied and the project created.



URL of repository: http://subvghs.upc.es/svn/rep/proost/branches/devFranci

Checkout directory: C:\devCheproo

☐ Multiple, independent working copies

Checkout Depth: Fully recursive

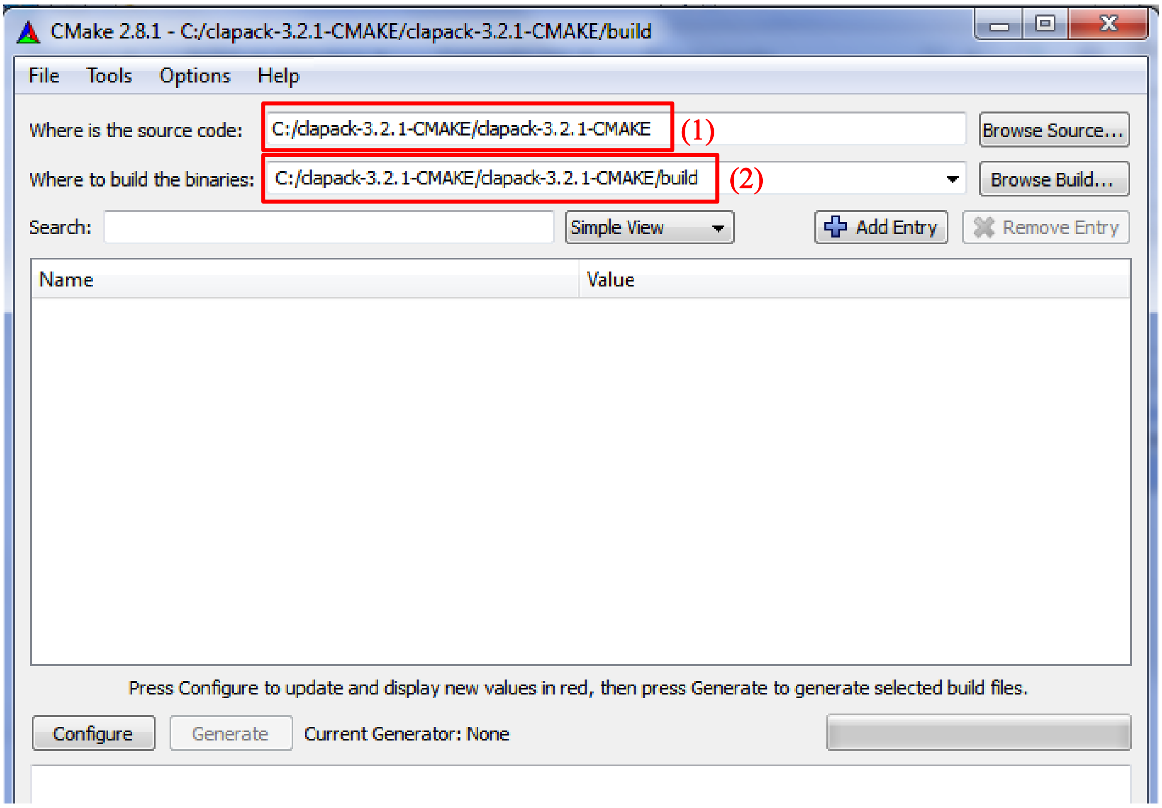
☐ Omit externals

Revision: ● HEAD revision ○ Revision

Figure 1 – SVN checkout

Click on “OK” to start downloading Cheproo++.

a) Open CMake to compile Clapack. Copy the directory where you unpacked the library in (1) and choose a subdirectory “build” where to build the library in (2) (Figure 2). Click on “Configure” and CMake will ask you for which version of Visual Studio needs to create the project (Figure 3). Choose the Visual Studio version you installed on your computer. Click on finish and CMake will create Clapack solution.



CMake 2.8.1 - C:/clapack-3.2.1-CMAKE/clapack-3.2.1-CMAKE/build

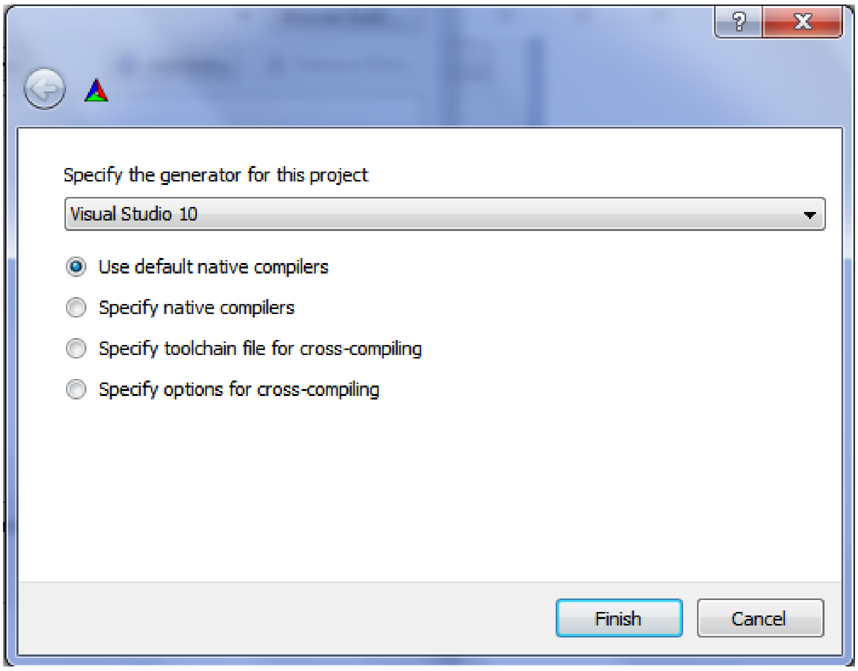
Where is the source code: C:/clapack-3.2.1-CMAKE/clapack-3.2.1-CMAKE (1)

Where to build the binaries: C:/clapack-3.2.1-CMAKE/clapack-3.2.1-CMAKE/build (2)

Press Configure to update and display new values in red, then press Generate to generate selected build files.

Current Generator: None

Figure 2 – CMake creation of Clapack project



Specify the generator for this project: Visual Studio 10

● Use default native compilers

○ Specify native compilers

○ Specify toolchain file for cross-compiling

○ Specify options for cross-compiling

Figure 3 – CMake choice for Visual Studio version

b) Compile Clapack. Go to the folder with the solution of Clapack just created with CMake (“C:\clapack-3.2.1-CMAKE\clapack-3.2.1-CMAKE\build”) and open the solution with Visual Studio (CLAPACK.sln). Build the solution with Visual Studio.

c) Open CMake to create the solution of CHEPROO++ for Visual Studio. The directory of the source code will be where you did the Checkout with SVN (i.e., “C:\devCheproo”), while the one to build the libraries will be “C:\devCheproo\build”. Click on Configure and choose the version of Visual Studio for which the project needs to be created. If asked for the “..\build” folder to be created, say yes. After the first time you click on Configure, you have to copy the folder where you built Clapack (“C:\clapack-3.2.1-CMAKE\clapack-3.2.1-CMAKE\build”) in the variable “clapack_DIR”. Configure again. When using the compiler on a computer with multiple cores, it is advantageous to add the compiler flags "/MP" to the CMAKE_CXX_FLAGS and CMAKE_C_FLAGS variables.

d) This should have created a solution called “CheProost.sln” in the folder “C:\devCheproo\build”. Every time you add or remove a file from the solution, add and then remove a space from one of the “CMakeLists.txt” files, save and rebuild the solution. This automatically reruns CMake.

e) The solution “CheProost.sln” contains different libraries
- CheprooLib: CHEPROO++ library
- proostLib: Proost library
- commonLib: library that CHEPROO++ and Proost share
- GALib: Genetic Algorithm library (used only by Proost)
- SparseLib: library for dealing with sparse matrices (used only by Proost)

And programs

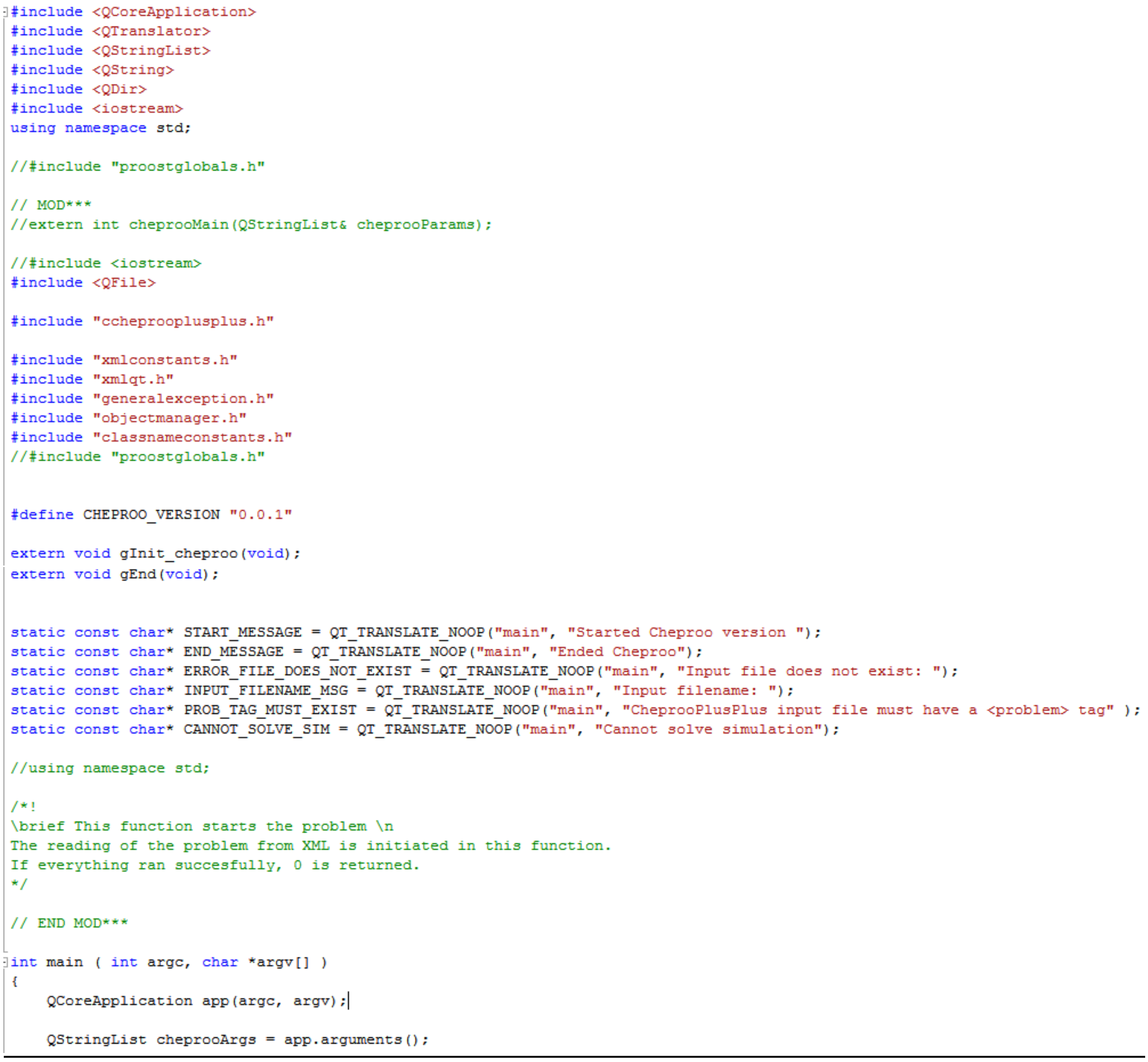
- CheprooUnitTests: unit tests for Cheproo++
- prstUnitTests: unit tests for Proost
- prstSystemTests: system tests for Proost
- cheprooPlusPlus: program to run CHEPROO++ standalone (see Section 4.1 for details)
- reactiveMix: program to use reactive mix method in CHEPROO++ (see Section 4.2 for details)

E.2 Programs that use CHEPROO++

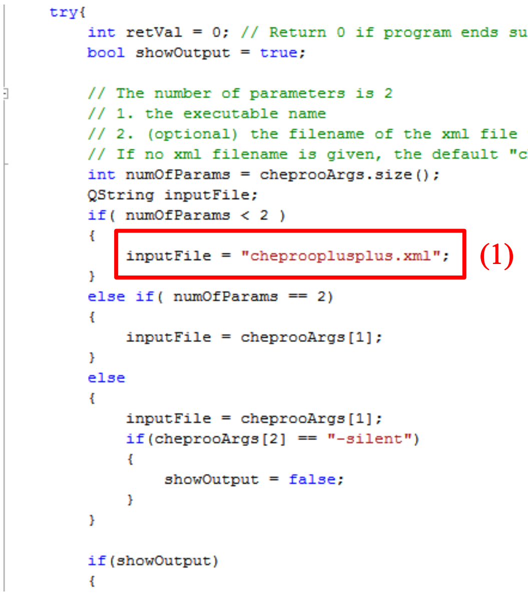
At the moment, the programs using CHEPROO++ are two: cheprooPlusPlus and reactiveMix.

E.2.1) cheprooPlusPlus

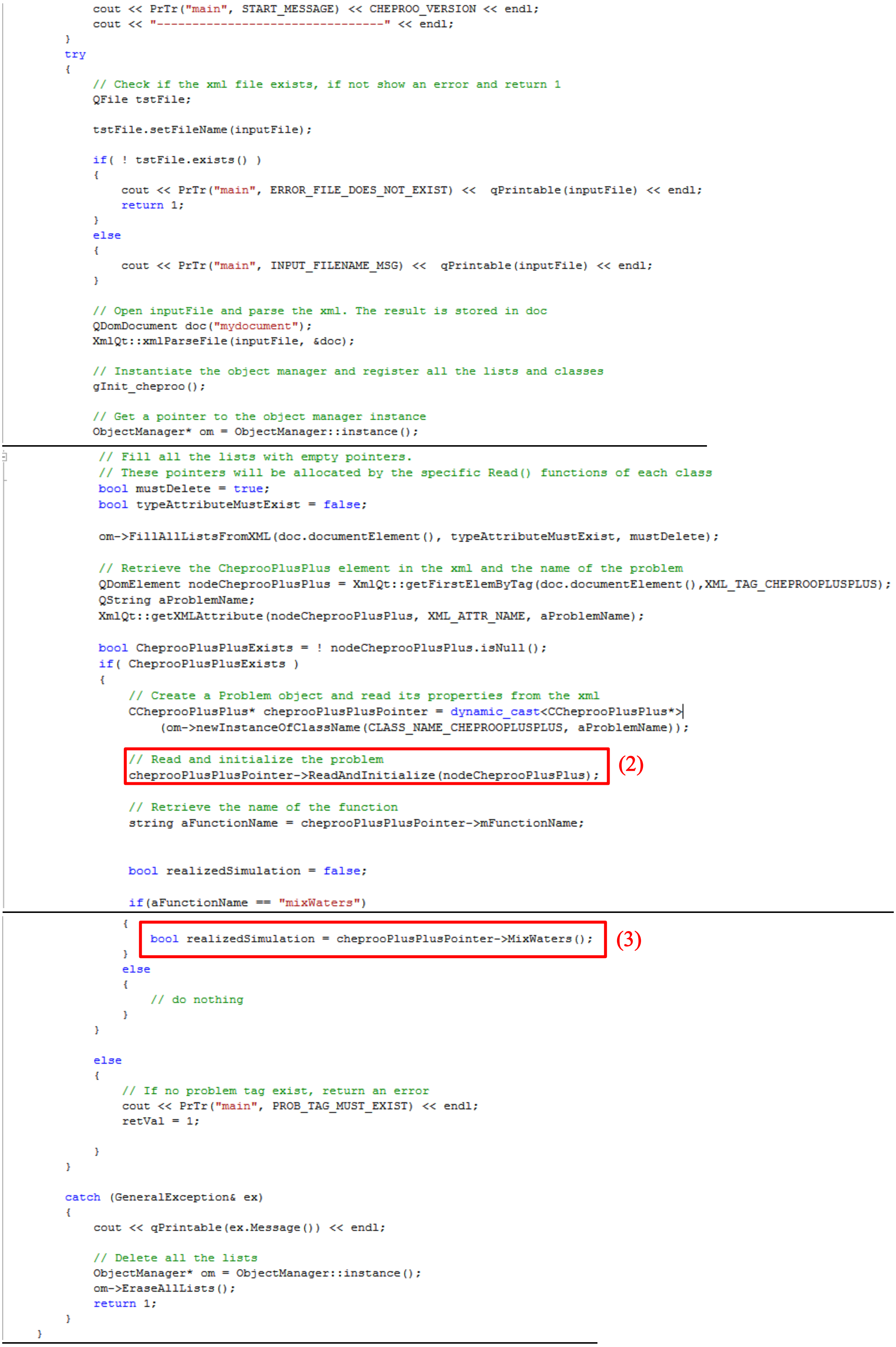
cheprooPlusPlus is a program that initializes CHEPROO++ and uses the attribute “`mFunctionName`” of CCheprooPlusPlus to perform the calculations. At the moment this program can be used to speciate and mix waters in known mixing proportions (i.e., “`mFunctionName = mixWaters`”). The main.cpp of this program is shown in Figure 4.



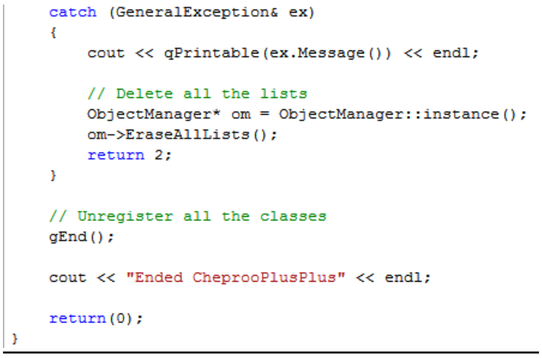
```
#include <QCoreApplication>
#include <QTranslator>
#include <QStringList>
#include <QString>
#include <QDir>
#include <iostream>
using namespace std;

//#include "proostglobals.h"

// MOD***
//extern int cheprooMain(QStringList& cheprooParams);

//#include <iostream>
#include <QFile>

#include "cchepro oplusplus.h"

#include "xmlconstants.h"
#include "xmlqt.h"
#include "generalexception.h"
#include "objectmanager.h"
#include "classnameconstants.h"
//#include "proostglobals.h"


#define CHEPROO_VERSION "0.0.1"

extern void gInit_cheproo(void);
extern void gEnd(void);


static const char* START_MESSAGE = QT_TRANSLATE_NOOP("main", "Started Cheproo version ");
static const char* END_MESSAGE = QT_TRANSLATE_NOOP("main", "Ended Cheproo");
static const char* ERROR_FILE_DOES_NOT_EXIST = QT_TRANSLATE_NOOP("main", "Input file does not exist: ");
static const char* INPUT_FILENAME_MSG = QT_TRANSLATE_NOOP("main", "Input filename: ");
static const char* PROB_TAG_MUST_EXIST = QT_TRANSLATE_NOOP("main", "CheprooPlusPlus input file must have a <problem> tag" );
static const char* CANNOT_SOLVE_SIM = QT_TRANSLATE_NOOP("main", "Cannot solve simulation");

//using namespace std;

/*!
\brief This function starts the problem \n
The reading of the problem from XML is initiated in this function.
If everything ran succesfully, 0 is returned.
*/

// END MOD***

int main ( int argc, char *argv[] )
{
    QCoreApplication app(argc, argv);

    QStringList cheprooArgs = app.arguments();
```



```
    try{
        int retVal = 0; // Return 0 if program ends succesfully
        bool showOutput = true;

        // The number of parameters is 2
        // 1. the executable name
        // 2. (optional) the filename of the xml file that contains the problem to be solved
        // If no xml filename is given, the default "chepro oplusplus.xml" is used
        int numOfParams = cheprooArgs.size();
        QString inputFile;
        if( numOfParams < 2 )
        {
            inputFile = "chepro oplusplus.xml";
        }
        else if( numOfParams == 2)
        {
            inputFile = cheprooArgs[1];
        }
        else
        {
            inputFile = cheprooArgs[1];
            if(cheprooArgs[2] == "-silent")
            {
                showOutput = false;
            }
        }

        if(showOutput)
        {
```



```
        cout << PrTr("main", START_MESSAGE) << CHEPROO_VERSION << endl;
        cout << "---------------------------------" << endl;
    }
    try
    {
        // Check if the xml file exists, if not show an error and return 1
        QFile tstFile;

        tstFile.setFileName(inputFile);

        if( ! tstFile.exists() )
        {
            cout << PrTr("main", ERROR_FILE_DOES_NOT_EXIST) <<  qPrintable(inputFile) << endl;
            return 1;
        }
        else
        {
            cout << PrTr("main", INPUT_FILENAME_MSG) <<  qPrintable(inputFile) << endl;
        }

        // Open inputFile and parse the xml. The result is stored in doc
        QDomDocument doc("mydocument");
        XmlQt::xmlParseFile(inputFile, &doc);

        // Instantiate the object manager and register all the lists and classes
        gInit_cheproo();

        // Get a pointer to the object manager instance
        ObjectManager* om = ObjectManager::instance();
        // Fill all the lists with empty pointers.
        // These pointers will be allocated by the specific Read() functions of each class
        bool mustDelete = true;
        bool typeAttributeMustExist = false;

        om->FillAllListsFromXML(doc.documentElement(), typeAttributeMustExist, mustDelete);

        // Retrieve the CheprooPlusPlus element in the xml and the name of the problem
        QDomElement nodeCheprooPlusPlus = XmlQt::getFirstElemByTag(doc.documentElement(),XML_TAG_CHEPROOPLUSPLUS);
        QString aProblemName;
        XmlQt::getXMLAttribute(nodeCheprooPlusPlus, XML_ATTR_NAME, aProblemName);

        bool CheprooPlusPlusExists = ! nodeCheprooPlusPlus.isNull();
        if( CheprooPlusPlusExists )
        {
            // Create a Problem object and read its properties from the xml
            CCheprooPlusPlus* cheprooPlusPlusPointer = dynamic_cast<CCheprooPlusPlus*>
                (om->newInstanceOfClassName(CLASS_NAME_CHEPROOPLUSPLUS, aProblemName));

            // Read and initialize the problem
            cheprooPlusPlusPointer->ReadAndInitialize(nodeCheprooPlusPlus);

            // Retrieve the name of the function
            string aFunctionName = cheprooPlusPlusPointer->mFunctionName;


            bool realizedSimulation = false;

            if(aFunctionName == "mixWaters")
            {
                bool realizedSimulation = cheprooPlusPlusPointer->MixWaters();
            }
            else
            {
                // do nothing
            }
        }

        else
        {
            // If no problem tag exist, return an error
            cout << PrTr("main", PROB_TAG_MUST_EXIST) << endl;
            retVal = 1;

        }
    }

    catch (GeneralException& ex)
    {
        cout << qPrintable(ex.Message()) << endl;

        // Delete all the lists
        ObjectManager* om = ObjectManager::instance();
        om->EraseAllLists();
        return 1;
    }
}
```



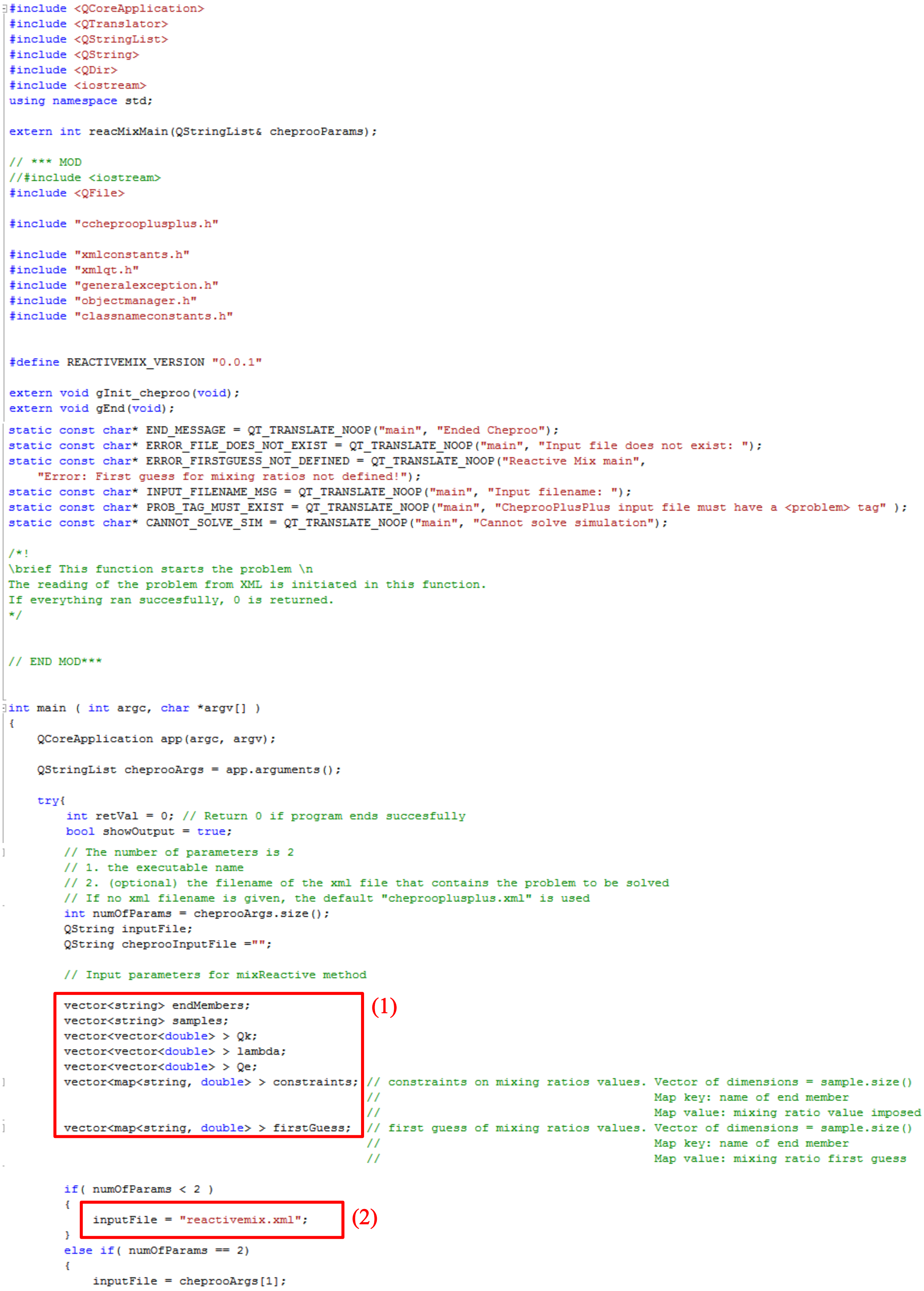
```
    catch (GeneralException& ex)
    {
        cout << qPrintable(ex.Message()) << endl;

        // Delete all the lists
        ObjectManager* om = ObjectManager::instance();
        om->EraseAllLists();
        return 2;
    }

    // Unregister all the classes
    gEnd();

    cout << "Ended CheprooPlusPlus" << endl;

    return(0);
}
```

Figure 4 – cheprooPlusPlus main file

The name of the CHEPROO++ input file by default is “cheprooplusplus.xml”, as shown in (1) (Figure 4). The method “ReadAndInitialize(nodeCheprooPlusPlus)” (2) trigger an initial speciation of all the waters defined in the input file, either “traditional” or with RISA (described in Chapter 1). The method to mix waters is called in (3).

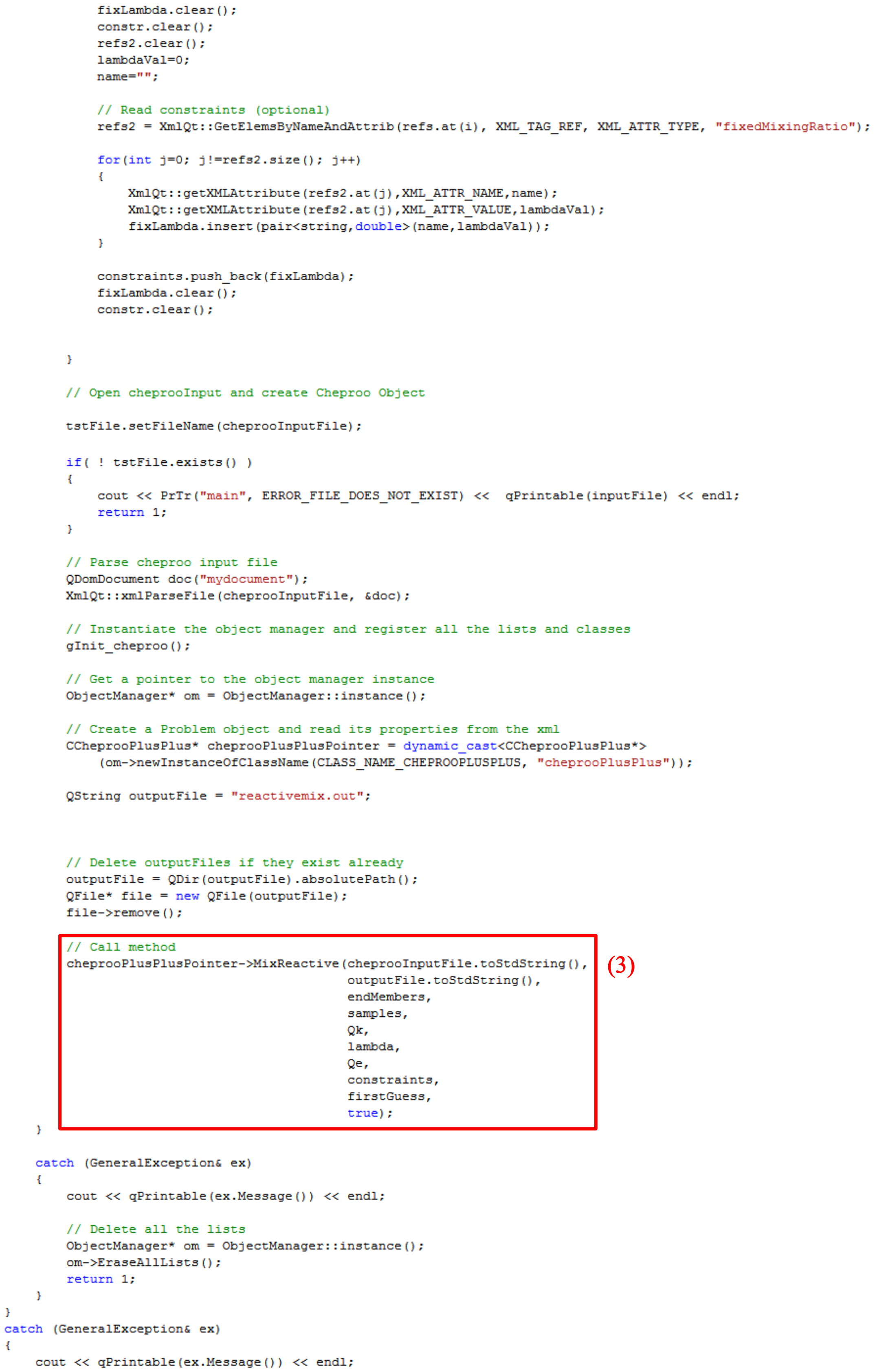
E.2.2) reactiveMix

reactiveMix is a program that allows the calculation of mixing proportions and quantities of reacted species in a number of samples, given a number of end-members and uncertain data associated to the samples. The algorithm is presented in Chapter 2. The main file of this program is shown in Figure 5.

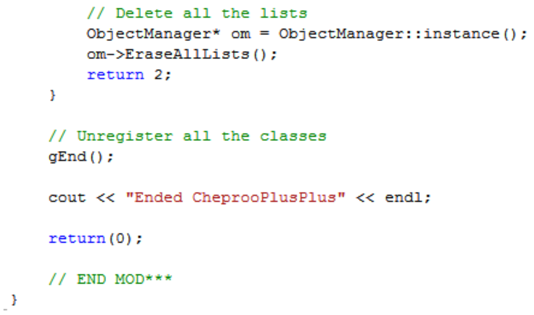


```
#include <QCoreApplication>
#include <QTranslator>
#include <QStringList>
#include <QString>
#include <QDir>
#include <iostream>
using namespace std;

extern int reacMixMain(QStringList& cheprooParams);

// *** MOD
//#include <iostream>
#include <QFile>

#include "ccheprooplusplus.h"

#include "xmlconstants.h"
#include "xmlqt.h"
#include "generalexception.h"
#include "objectmanager.h"
#include "classnameconstants.h"


#define REACTIVEMIX_VERSION "0.0.1"

extern void gInit_chepro(void);
extern void gEnd(void);
static const char* END_MESSAGE = QT_TRANSLATE_NOOP("main", "Ended Cheproo");
static const char* ERROR_FILE_DOES_NOT_EXIST = QT_TRANSLATE_NOOP("main", "Input file does not exist: ");
static const char* ERROR_FIRSTGUESS_NOT_DEFINED = QT_TRANSLATE_NOOP("Reactive Mix main",
    "Error: First guess for mixing ratios not defined!");
static const char* INPUT_FILENAME_MSG = QT_TRANSLATE_NOOP("main", "Input filename: ");
static const char* PROB_TAG_MUST_EXIST = QT_TRANSLATE_NOOP("main", "CheprooPlusPlus input file must have a <problem> tag" );
static const char* CANNOT_SOLVE_SIM = QT_TRANSLATE_NOOP("main", "Cannot solve simulation");

/*!
\brief This function starts the problem \n
The reading of the problem from XML is initiated in this function.
If everything ran succesfully, 0 is returned.
*/


// END MOD***


int main ( int argc, char *argv[] )
{
    QCoreApplication app(argc, argv);

    QStringList cheprooArgs = app.arguments();

    try{
        int retVal = 0; // Return 0 if program ends succesfully
        bool showOutput = true;
        // The number of parameters is 2
        // 1. the executable name
        // 2. (optional) the filename of the xml file that contains the problem to be solved
        // If no xml filename is given, the default "cheprooplusplus.xml" is used
        int numOfParams = cheprooArgs.size();
        QString inputFile;
        QString cheprooInputFile ="";

        // Input parameters for mixReactive method

        vector<string> endMembers;
        vector<string> samples;
        vector<vector<double> > Qk;
        vector<vector<double> > lambda;
        vector<vector<double> > Qe;
        vector<map<string, double> > constraints; // constraints on mixing ratios values. Vector of dimensions = sample.size()
                                                  //                                       Map key: name of end member
                                                  //                                       Map value: mixing ratio value imposed
        vector<map<string, double> > firstGuess;  // first guess of mixing ratios values. Vector of dimensions = sample.size()
                                                  //                                       Map key: name of end member
                                                  //                                       Map value: mixing ratio first guess

        if( numOfParams < 2 )
        {
            inputFile = "reactivemix.xml";
        }
        else if( numOfParams == 2)
        {
            inputFile = cheprooArgs[1];
```

```
    if(cheprooArgs[2] == "-silent")
    {
        showOutput = false;
    }
}

if(showOutput)
{
    cout << PrTr("main", START_MESSAGE) << REACTIVEMIX_VERSION << endl;
    cout << "--------------------------------" << endl;
}
try
{
    // Check if the xml file exists, if not show an error and return 1
    QFile tstFile;

    tstFile.setFileName(inputFile);

    if( ! tstFile.exists() )
    {
        cout << PrTr("main", ERROR_FILE_DOES_NOT_EXIST) <<  qPrintable(inputFile) << endl;
        return 1;
    }
    else
    {
        cout << PrTr("main", INPUT_FILENAME_MSG) <<  qPrintable(inputFile) << endl;
    }

    // Open inputFile and parse the xml. The result is stored in doc
    QDomDocument doc1("mydocument");
    XmlQt::xmlParseFile(inputFile, &doc1);

    // Read name of input file for cheproo
    QDomElement elem= XmlQt::getFirstElemByTag(doc1.documentElement(),"inputFileCheproo");
    XmlQt::getXMLAttribute(elem, XML_ATTR_NAME, cheprooInputFile);

    // Read endmembers names

    elem = XmlQt::getFirstElemByTag(doc1.documentElement(),"endMembers");

    vector<QDomElement> refs = XmlQt::GetElemsByNameAndAttrib(elem, XML_TAG_REF, XML_ATTR_TYPE, "endMember");

    string name ="";

    for(int i=0; i!=refs.size(); i++)
    {
        XmlQt::getXMLAttribute(refs.at(i),XML_ATTR_NAME,name);
        endMembers.push_back(name);
    }

    // Read samples names, first guess of lambda and constraints (if there are)

    elem = XmlQt::getFirstElemByTag(doc1.documentElement(),"samples");

    refs = XmlQt::GetElemsByNameAndAttrib(elem, XML_TAG_REF, XML_ATTR_TYPE, "sample");
    name ="";
    double lambdaVal = 0.0;
    vector<QDomElement> constr;
    map<string,double> fixLambda;
    vector<QDomElement> refs2;

    for(int i=0; i!=refs.size(); i++)
    {
        XmlQt::getXMLAttribute(refs.at(i),XML_ATTR_NAME,name);
        samples.push_back(name);

        // Read first guess values
        refs2 = XmlQt::GetElemsByNameAndAttrib(refs.at(i), XML_TAG_REF, XML_ATTR_TYPE, "firstGuessMixingRatio");

        if( refs2.size()==0 )
        {
            cout << PrTr("Reactive Mix main", ERROR_FIRSTGUESS_NOT_DEFINED) << endl;
            return 1;
        }

        for(int j=0; j!=refs2.size(); j++)
        {
            XmlQt::getXMLAttribute(refs2.at(j),XML_ATTR_NAME,name);
            XmlQt::getXMLAttribute(refs2.at(j),XML_ATTR_VALUE,lambdaVal);
            fixLambda.insert(pair<string,double>(name,lambdaVal));
        }

        firstGuess.push_back(fixLambda);
```



```
            fixLambda.clear();
            constr.clear();
            refs2.clear();
            lambdaVal=0;
            name="";

            // Read constraints (optional)
            refs2 = XmlQt::GetElemsByNameAndAttrib(refs.at(i), XML_TAG_REF, XML_ATTR_TYPE, "fixedMixingRatio");

            for(int j=0; j!=refs2.size(); j++)
            {
                XmlQt::getXMLAttribute(refs2.at(j),XML_ATTR_NAME,name);
                XmlQt::getXMLAttribute(refs2.at(j),XML_ATTR_VALUE,lambdaVal);
                fixLambda.insert(pair<string,double>(name,lambdaVal));
            }

            constraints.push_back(fixLambda);
            fixLambda.clear();
            constr.clear();


        }

        // Open cheprooInput and create Cheproo Object

        tstFile.setFileName(cheprooInputFile);

        if( ! tstFile.exists() )
        {
            cout << PrTr("main", ERROR_FILE_DOES_NOT_EXIST) <<  qPrintable(inputFile) << endl;
            return 1;
        }

        // Parse cheproo input file
        QDomDocument doc("mydocument");
        XmlQt::xmlParseFile(cheprooInputFile, &doc);

        // Instantiate the object manager and register all the lists and classes
        gInit_cheproo();

        // Get a pointer to the object manager instance
        ObjectManager* om = ObjectManager::instance();

        // Create a Problem object and read its properties from the xml
        CCheprooPlusPlus* cheprooPlusPlusPointer = dynamic_cast<CCheprooPlusPlus*>
            (om->newInstanceOfClassName(CLASS_NAME_CHEPROOPLUSPLUS, "cheprooPlusPlus"));

        QString outputFile = "reactivemix.out";



        // Delete outputFiles if they exist already
        outputFile = QDir(outputFile).absolutePath();
        QFile* file = new QFile(outputFile);
        file->remove();

        // Call method
        cheprooPlusPlusPointer->MixReactive(cheprooInputFile.toStdString(),
                                            outputFile.toStdString(),
                                            endMembers,
                                            samples,
                                            Qk,
                                            lambda,
                                            Qe,
                                            constraints,
                                            firstGuess,
                                            true);
    }

    catch (GeneralException& ex)
    {
        cout << qPrintable(ex.Message()) << endl;

        // Delete all the lists
        ObjectManager* om = ObjectManager::instance();
        om->EraseAllLists();
        return 1;
    }
}
catch (GeneralException& ex)
{
    cout << qPrintable(ex.Message()) << endl;
```



```
        // Delete all the lists
        ObjectManager* om = ObjectManager::instance();
        om->EraseAllLists();
        return 2;
    }

    // Unregister all the classes
    gEnd();

    cout << "Ended CheprooPlusPlus" << endl;

    return(0);

    // END MOD***
}
```

Figure 5 – reactiveMix main file

The name of the reactiveMix input file by default is “reactivemix.xml”, as shown in (2) (Figure 5). In the main the input/output parameters for the method are declared (1) and then the input parameters are read. The reactive mix method is called in (3).

E.3 How to build a program with CHEPROO++

Let’s assume that a user wants to build a new program called “programA” that uses CHEPROO++ through a new method, called “methodA”. The following steps need to be followed:

a) Create a new folder (e.g., “programA”) in “C:\devCheproo\cheproo\programA”

b) The new folder just created needs to contain two files: the main file for “programA” (e.g., “programA.cpp”) and a CMake file (“CMakeLists.txt”) to include the new program in the project. The easiest way for a user to create these two files is to copy them from the reactiveMix folder and to paste them into “C:\devCheproo\cheproo\programA”, for then modifying them for the new program.

c) Once “programA.cpp” and “CMakeLists.txt” are contained in the folder “C:\devCheproo\cheproo\ programA”, open the “programA.cpp” file and modify it for the new method: declare input/output parameters (see (1) in Figure 5) and the name of the new input file (see (2) in Figure 5). Parse the input file like shown in Figure 5 and store the variables in the appropriate input parameters. Call the new method like shown in (3) in Figure 5, passing the parameters to the method. Clearly, “methodA” needs to be implemented in CCheprooPlusPlus, CGlobalChemicalSystem and CLocalChemicalSystem.

d) Open “CMakeLists.txt” in “C:\devCheproo\cheproo\programA”. The file will be the same as shown in Figure 6, and by now it coincides with the CMakeLists.txt of the reactiveMix program. A few things need to be changed though: the name of the project, (1) in Figure 6, the name of the variable to store the name of the main file, (2) in Figure 6, the folder to store the executable, (3) in Figure 6, and finally the variables in the ADD_EXECUTABLE and TARGET_LINK_LIBRARIES commands, (4) and (5) in Figure 6, respectively.

e) Open “CMakeLists.txt” in “C:\devCheproo\cheproo”. The file will be the same as shown in Figure 7, but it needs to be modified to account for the new project. Copy the lines contained in box (1) of Figure 7 and paste them under the same box. Change the “reactiveMix” variables for the same variables defined in (4) and (5) in Figure 6.

```
CMAKE_MINIMUM_REQUIRED(VERSION 2.6)

PROJECT (ReactiveMix)

file(GLOB REACTIVEMIX_SRC "*.cpp") #glob looks for all files in current directory.
 message(STATUS  ${CHEPROO_ROOT_DIR})
include_directories( ${PROJECT_SOURCE_DIR} )

SET( EXECUTABLE_OUTPUT_PATH ${CHEPROO_ROOT_DIR}/bin/reactiveMix )
SET( LIBRARY_OUTPUT_PATH ${CHEPROO_ROOT_DIR}/lib )

FIND_PACKAGE( Qt4 REQUIRED)
SET( QT_DONT_USE_QTGUI true )
SET(QT_USE_QTXML true)
INCLUDE( ${QT_USE_FILE} )


#get eigen
find_package(Eigen3 REQUIRED)
if(EIGEN3_FOUND)
    include_directories(${EIGEN3_INCLUDE_DIR})
endif()


include_directories(${COMMONLIB_SRC_DIR} ${CHEPROO_ROOT_DIR}/lib/proostfiles ${CHEPROO_ROOT_DIR}/lib/cheprooLib)


ADD_EXECUTABLE (reactiveMix ${REACTIVEMIX_SRC} )

TARGET_LINK_LIBRARIES(reactiveMix commonLib CheprooLib ${QT_LIBRARIES})
```

Figure 6 – CMakeLists.txt file in “C:\devCheproo\cheproo\programA”


(1)

```
cmake_minimum_required(VERSION 2.6)

project(Cheproo++)


set(CMAKE_MODULE_PATH ${PROJECT_SOURCE_DIR}/cmake)
set(CHEPROO_ROOT_DIR ${PROJECT_SOURCE_DIR})




#add the tests
if( IS_DIRECTORY ${PROJECT_SOURCE_DIR}/testing)
    add_subdirectory ( ${PROJECT_SOURCE_DIR}/testing )
endif()

#add the executable
if( IS_DIRECTORY ${PROJECT_SOURCE_DIR}/src)
    add_subdirectory ( ${PROJECT_SOURCE_DIR}/src )
endif()

#add the reactiveMix
if( IS_DIRECTORY ${PROJECT_SOURCE_DIR}/reactiveMix)
    add_subdirectory ( ${PROJECT_SOURCE_DIR}/reactiveMix )
endif()


#add the library
if( IS_DIRECTORY ${PROJECT_SOURCE_DIR}/lib/cheprooLib )
    message(STATUS "library found")
    add_subdirectory ( ${PROJECT_SOURCE_DIR}/lib/cheprooLib )
else()
    message(STATUS "library source not found")
endif()
```

Figure 7 – CMakeLists.txt file in “C:\devCheproo\cheproo”

You have now built a “programA”, which can use the libraries CHEPROO++, Qt and Eigen.

E.4 Input file description

CHEPROO++ uses four xml files, some compulsory and some optional:

- “cheprooplusplus.xml” (compulsory)
- “mastertemp.xml” (compulsory)
- “kinetics.xml” (only if kinetic reactions are defined)
- “reactivemix.xml” (only for ReactiveMix simulations)

The definition of the input xml files reflects the structure of CHEPROO++, so that every element represents a class of CHEPROO++.

E.4.1) File “cheprooplusplus.xml”

The structure of “cheprooplusplus.xml” is shown below.

```
<cheprooPlusPlusInputFile name="cheproo_input">

        <cheprooPlusPlus  name="simulation1">
            ...
         </cheprooPlusPlus>

        <globalChemicalSystem  buildOption="readFromMaster25"
                               thermodynamicDatabase="completemaster25.xml">
            ...
         </ globalChemicalSystem  >


        <proostList type = "water">
            ...
         </ proostList  >


</cheprooPlusPlusInputFile>
```

The element `<cheprooPlusPlusInputFile>` is compulsory. The value of the attribute “name” can be chosen from the user. This element contains three elements: `<cheprooPlusPlus>`, `<globalChemicalSystem>` and `<proostList>`. Let’s analize the element `<cheprooPlusPlus>` first.

```
<cheprooPlusPlus name="simulation1">
        <function name="mixWaters" >
                <ref type="chemicalComposition" ID="1"  mixingRatio="0.6">water_1</ref>
                <ref type="chemicalComposition" ID="2" mixingRatio="0.4">water_2</ref>
        </function>
</cheprooPlusPlus>
```

The element <cheprooPlusPlus> is compulsory. The value of the attribute “name” can be chosen from the user. The element <function> represents the function of CHEPROO++ that will be called. The definition of only one element <function> is allowed. The user need to define one element <ref> for every water that needs to be mixed. For the definition of their attributes see the Tables below.

Element <function>			
Attribute	Possible values	Description	Remarks
name	mixWaters	name of the method to mix waters in fixed proportions	Compulsory

Element <ref>			
Attribute	Possible values	Description	Remarks
type	chemicalComposition		Compulsory
ID	1, 2, 3,…		Compulsory
mixingRatio	0 < mixingRatio < 1	Proportions of mixing waters	Compulsory

The element <globalChemicalSystem>, which is compulsory, can be defined as follows in the input file:

```
<globalChemicalSystem buildOption="readFromMasterTemp"
                      thermodynamicDatabase="mastertemp.xml"
                      kineticDatabase="kinetics.xml">

        <phase name="aqueous1" type="aqueousDebyeHuckel" model="aqueous">
                <ref type="species">h+</ref>
                <ref type="species">oh-</ref>
                <ref type="species">hco3-</ref>
                <ref type="species">co2(aq)</ref>
                <ref type="species">cahco3+</ref>
                <ref type="species">ca+2</ref>
                <ref type="species">co3-2</ref>
                <ref type="species">cl-</ref>
                <ref type="species">h2o</ref>
        </phase>
        <phase name="mineral1" type="mineralPure" model="mineral">
                <ref type="species">calcite</ref>
        </phase>

        <kinReactions>
                <ref type="reaction">calcite</ref>
        </kinReactions>

        <localChemicalSystems>
                <localChemicalSystem name="local1" type="localChemicalSystemConstPrimary"
                                     maxRelativeError="1e-7" maxResidual="1e-10"
                                     maxIterNum="100" maxIncrementFactor="100"
                                     writeConvInfos="true"/>
                </localChemicalSystems>

</globalChemicalSystem>
```

Element < globalChemicalSystem >			
Attribute	Possible values	Description	Remarks
buildOption	readFromMasterTemp	Only option to build GCS object	Compulsory
thermodynamicDatabase	mastertemp.xml	Name of the thermodynamic database	Compulsory
kineticDatabase	kinetics.xml	Name of the kinetic database	Compulsory

Element < phase >			
Attribute	Possible values	Description	Remarks
name	any	Name of the phase	Compulsory
type	aqueousDebyeHuckel(*) aqueousIdeal mineralPure gasIdeal	Model to calculate activity coefficients	Compulsory
kineticDatabase	kinetics.xml	Name of the kinetic database	Compulsory
model	aqueous mineral surface gas	Type of phase	Compulsory

(*) The type of phase determines how the activity coefficients of the N_s species belonging to the phase ($\boldsymbol{\gamma}$, dim($\boldsymbol{\gamma}$)=N_s) are calculated. For ideal phases, $\boldsymbol{\gamma}$=1. For aqueousDebyeHuckel phase, the activity coefficients of every *i*-th species are evaluated by means of the following expression:

$$\gamma_i = -\frac{A z_i \sqrt{I}}{1 + r_i B \sqrt{I}} + \dot{b} I \tag{E.1}$$

where z_i and r_i are the ionic charge and radius of the species, respectively. A, B and $\dot{b}$ are coefficients that depend on the temperature, while I is the ionic strength of the solution defined as

$$I = 0.5 \sum_{i=1}^{Ns} \left(z_i^2 c_i \right) \tag{E.2}$$

c_i is the concentration of the *i*-th species.

The elements <ref> contain the species belonging to the phase that will be included in the chemical system. They must be defined for every phase.

Element <ref>			
Attribute	Possible values	Description	Remarks
type	species		Compulsory
text		Name of the species	Compulsory

Optional Element < kinReactions >	
Description	Remarks
Element containing the names of the kinetic reactions.	Optional: if no <kinReactions>is defined, all reactions are in equilibrium

The elements <ref> contain the names of the kinetic reactions:

Element <ref>			
Attribute	Possible values	Description	Remarks
type	reaction		Compulsory
text		Name of the kinetic reaction	Compulsory

Element < localChemicalSystems >	
Description	Remarks
Element that wraps the Local Chemical Systems defined.	By now, the definition of only one Local Chemical System is allowed. This element is compulsory

Element < localChemicalSystem >			
Attribute	Possible values	Description	Remarks
name		Name of the Local Chemical System	Compulsory
type	localChemicalSystem ConstPrimary	Type of Local Chemical System	Compulsory
solverType	N-R (for Newton-Raphson, Picard otherwise)	Solver type to calculate $\mathbf{c}_2$ Default: Picard	Optional
maxRelativeError		Maximum value of relative error in speciation	Compulsory
maxResidual		Maximum value of residual in speciation	Compulsory
maxIterNum		Maximum number of iteration in speciation	Compulsory
maxIncrement Factor		Maximum increment allowed for conc. in speciation (*) Default value: 10	Optional
writeConvInfos		If true, convergence info about speciation are printed in a file	Optional

(*) At every iteration i+1 of the speciation methods the concentration of a j-th species (c_j^{i+1}) is compared to the concentration at the previous iteration by means of the factor f :

1) If $c_j^{i+1} < \frac{c_j^i}{f} \Rightarrow c_j^{i+1} = \frac{c_j^i}{f}$
2) If $c_j^{i+1} > c_j^i \cdot f \Rightarrow c_j^{i+1} = c_j^i \cdot f$

The element `<proostList>`, containing the list of the chemical compositions, is also compulsory. It can be defined as follows:

```
<proostList type="chemicalComposition">
        <chemicalComposition ID="1" name="water1" temp="25.00">
                <ref type="firstGuess" cGuess="1.0" primarySpecies="h2o"/>

                <ref type="firstGuess" cGuess="1.05e-8" primarySpecies="h+"/>
                <ref type="firstGuess" cGuess="1.108e-4" primarySpecies="cl-"/>
                <ref type="firstGuess" cGuess="1.99e-5" primarySpecies="hco3-"/>
                <ref type="firstGuess" cGuess="9.68e-4" primarySpecies="ca+2"/>

                <ref type="constraint" iCon="activity" value="1.0" name="h2o"/>
                <ref type="constraint" iCon="cTot" value="9.68e-4" name="ca+2" />
                <ref type="constraint" iCon="cTot" value="1.00e-03" name="hco3-" />
                <ref type="constraint" iCon="activity" value="1.00e-08" name="h+"/>
                <ref type="constraint" iCon="cTot" value="1.0807e-04" name="cl-"/>

                <ref type="phaseContent" name="aqueous1"  value="1.0"/>
                <ref type="phaseContent" name="mineral1"  value="0.5"/>
                 <ref type="reactiveArea" name="mineral1"  value="6.8e-5"/>

        </chemicalComposition>
</proostList>
```

Element < proostList >		
Attribute	Value	Remarks
type	chemicalComposition	Compulsory

In the `< proostList >` element the user can define as many chemical compositions as needed. Every chemical composition needs to be included in a separate `< chemicalComposition >` element, whose attributes are described below.

Element < chemicalComposition >			
Attribute	Possible values	Description	Remarks
ID	1, 2, 3, …	The ID needs to match the IDs defined in the <function> element, if defined.	Compulsory
name		Name of the Chemical Composition	Compulsory
temp	0ºC < temp < 300ºC	Temperature [ºC] at which the Chemical Composition is defined	Compulsory

In `< chemicalComposition >`, the `<ref>` elements represent different types of entities (e.g., first guess of concentrations, constraints, phase contents and reactive area of minerals), which can have different attributes. For this reason, we show below the list of attributes for every entity in separate Tables. All these `<ref>` elements are compulsory, except for the reactive area.

Element <ref type="firstGuess">		
Attribute	Description	Remarks
cGuess	guess value of concentration	Compulsory
primarySpecies	names of primary species	Compulsory
species	names of secondary species	Optional

Element <ref type="constraint">			
Attribute	Possible values	Description	Remarks
iCon	cTot activity conc alk chgbal TDIC eqmin eqgas EC	total concentration of a species is imposed activity of a species is imposed concentration of a species is imposed alkalinity is imposed (*) charge balance is imposed total dissolved inorganic carbon is imposed (**) equilibrium with mineral phase is imposed equilibrium with gas phase is imposed electrical conductivity is imposed	Compulsory Only for RISA
value	concentrations in [mol/kgw], EC in [mS/cm]	values of the constraints imposed	Compulsory
name		names of the species relative to the constraint	Optional only for chgbal constraint
isLog10Value	true/false	True if the value imposed is in log10. Default true for iCon=eqmin,activity,eqgas Default false for other iCon	Only for RISA
isRelError	true/false	True if the error is relative. Default true if isLog10Value=false Default false if isLog10Value=true	Only for RISA
std		Parameter to evaluate uncertainty relative to the data (***)	Only for RISA

(*) alk = $HCO_3^- + 2CO_3^{2-} + OH^- + 2CaCO_{3(aq)} - H^+$

(**) TDIC = $HCO_3^- + CO_3^{2-} + CO_{2(aq)}$

(***) The uncertainty of each *i*-th datum (σ_i of the covariance matrix **V**, Eq. 2.7, Chapter 2) is calculated as follows:

$$\sigma_i = \begin{cases} (std \cdot value) & \text{if isRelError = true} \\ (std \cdot std) & \text{if isRelError = false} \end{cases} \tag{E.3}$$

Element <ref type=”phaseContent”>		
Attribute	Description	Remarks
name	name of the phase defined in Global Chemical System	Compulsory
value	value of the phase content in $[m^3_\alpha/m^3_{tot}]$, α = phase index	Compulsory

Element <ref type=”reactiveArea”>		
Attribute	Description	Remarks
name	name of the phase defined in Global Chemical System	Compulsory
value	value of the reactive area in $[m^{-1}]$, α = phase index	Compulsory

E.4.2) File “mastertemp.xml”

“mastertemp.xml” is a file containing the properties of each species belonging to every phase and the definition of the reactions. The structure of “mastertemp.xml” is shown below.

```
<dataBase name ="mastertemp">

        <phases>
        ...
        </phases>

        <reactions>
        ...
        </reactions>

 </dataBase>
```

This input file must contain at least the species/phases defined in “cheprooplusplus.xml”. A reaction is possible (i.e., it will be included in the chemical system) for CHEPROO++ if all the species of the reaction have been defined by the user in the “cheprooplusplus.xml” input file. A complete list of species, phases and reactions is available in a file called “mastertemp_complete.xml”, which is automatically downloaded with CHEPROO++. It is recommended to copy species, phases and reactions from this file and paste them in a smaller “mastertemp.xml” file for the reading process to be faster.

The element <dataBase> is compulsory. The value of the attribute “name” can be chosen from the user. This element contains two elements: <phases> and <reactions>. Both of them are compulsory. Let’s analyse the element <phases> first.

```
<phases>
   <phase name ="phase1" type="aqueous">
      <species name="h2o" ionicRadius="3.0" ionicCharge=" 0.0" molWeight="18"
                       isClassic ="true"/>
      <species name="ca+2" ionicRadius="6.0" ionicCharge=" 2.0" molWeight="40.078"
                        isClassic ="true"/>
      <species name="cl-" ionicRadius="3.0" ionicCharge="-1.0" molWeight="35.453"
                      isClassic ="true"/>
      <species name="h+" ionicRadius="9.0" ionicCharge=" 1.0" molWeight="1"
```

```
                         isClassic ="true"/>
      <species name="hco3-" ionicRadius="4.0" ionicCharge="-1.0" molWeight="61"
                            isClassic ="true"/>
      <species name="co2(aq)" ionicRadius="3.0" ionicCharge=" 0.0" molWeight="44.01"/>
      <species name="co3-2" ionicRadius="5.0" ionicCharge="-2.0" molWeight="60"/>
      <species name="cahco3+" ionicRadius="4.0" ionicCharge=" 1.0" />
      <species name="oh-" ionicRadius="3.0" ionicCharge="-1.0"  molWeight="17"/>
   </phase>

   <phase name ="phase2" type="mineral">
      <species name="calcite" molarVolume=" 36.934" />
   </phase>

   <phase name ="phase3" type="gas">
      <species name="co2(g)" molarVolume="24465.00" difVolume="0.0" />
   </phase>

   <phase name ="phase4" type="surface">
      <species name="xoh"/>
   </phase>
</phases>
```

Element <phase>			
Attribute	Possible values	Description	Remarks
name	any	name of the phase	Compulsory
type	aqueous mineral gas surface	type of the phase	Compulsory

Each phase element contains elements corresponding to the species belonging to the phase. Since species belonging to different phases can have different attributes we show below the list of attributes for every entity in separate Tables.

Element <species> of <phase type=”aqueous”>			
Attribute	Possible values	Description	Remarks
name		name of the species	Compulsory
ionicRadius		ionic radius of the species	Compulsory
ionicCharge		ionic charge of the species	Compulsory
molWeight		molar weight of the species [g/mol]	Compulsory
isClassic	true/false	true for component species(*) Default: false	Optional

(*) Component species are the species that build reactions

Element <species> of <phase type=”mineral”>		
Attribute	Description	Remarks
name	name of the species	Compulsory
molarVolume	molar volume of the species [cm^3/mol]	Compulsory

Element <species> of <phase type=”gas”>		
Attribute	Description	Remarks
name	name of the species	Compulsory
molarVolume	molar volume of the species [cm^3/mol]	Compulsory

Element <species> of <phase type=”surface”>		
Attribute	Description	Remarks
name	name of the species	Compulsory

The element `<reactions>` contains the reactions.

```
<reactions>

     <reaction name="oh-" >
       <logK>
         <ref type="temp" ID="  0.0" value="   14.9398" />
         <ref type="temp" ID=" 25.0" value="   13.9951" />
         <ref type="temp" ID=" 60.0" value="   13.0272" />
         <ref type="temp" ID="100.0" value="   12.2551" />
         <ref type="temp" ID="150.0" value="   11.6308" />
         <ref type="temp" ID="200.0" value="   11.2836" />
         <ref type="temp" ID="250.0" value="   11.1675" />
         <ref type="temp" ID="300.0" value="   11.3002" />
       </logK>
         <ref type="species" ID="-1.0" value="oh-              " />
         <ref type="species" ID="-1.0" value="h+               " />
         <ref type="species" ID=" 1.0" value="h2o              " />
     </reaction>

    <reaction name="calcite" >
       <logK>
         <ref type="temp" ID="  0.0" value="    2.2257" />
         <ref type="temp" ID=" 25.0" value="    1.8487" />
         <ref type="temp" ID=" 60.0" value="    1.3330" />
         <ref type="temp" ID="100.0" value="    0.7743" />
         <ref type="temp" ID="150.0" value="    0.0999" />
         <ref type="temp" ID="200.0" value="   -0.5838" />
         <ref type="temp" ID="250.0" value="   -1.3262" />
         <ref type="temp" ID="300.0" value="   -2.2154" />
       </logK>
         <ref type="species" ID="-1.0" value="calcite          " />
         <ref type="species" ID="-1.0" value="h+               " />
         <ref type="species" ID=" 1.0" value="ca+2             " />
         <ref type="species" ID=" 1.0" value="hco3-            " />
```

```
    </reaction>

</reactions>
```

Element <reaction>		
Attribute	Description	Remarks
name	name of the reaction	Compulsory

For each reaction, logK values for different temperatures are given. In fact CHEPROO++ calculates the coefficients $A_0,\ldots, A_4$ necessary to evaluate the equilibrium constants as a function of the temperature according to:

$$\log K = A_0 \log T + A_1 + A_2 T + A_3 / T + A_4 / T^2 \qquad \text{(E.4)}$$

Element <ref type=”temp”> of <logK>		
Attribute	Description	Remarks
ID	Temperature [ºC]	Compulsory
value	logK at Temperature ID	Compulsory

In the elements <ref type="species"> are listed the species and stoichiometric coefficients for every reaction.

Element <ref type=”species”> of <reaction>		
Attribute	Description	Remarks
ID	Stoichiometric coefficient of the species	Compulsory
value	name of the species	Compulsory

E.4.3) File “kinetics.xml”

“kinetics.xml” is a file containing the definition of the kinetic rate laws. It must contain at least the kinetic rate laws of the kinetic reactions defined in the file “cheprooplusplus.xml”. At the moment in CHEPROO++ the following expression of dissolution/precipitation rate can be defined for a *k*-th mineral

$$r_k = \sigma_k \xi_k \exp\left(-\frac{E_{a,k}}{RT}\right) \sum_{i=1}^{Nk} k_i \prod_{j=1}^{Ns} a_{ij}^{n_{ij}} \left(\Omega_k^{p_i} - 1\right)^{q_i} \qquad \text{(E.5)}$$

where r_k is the mineral dissolution rate (moles per unit volume per unit time), σ_k is the reactive surface of the mineral per unit of volume, E_a is the activation energy of the reaction, Nk is the number of experimental terms and k an experimental constant. The coefficients $a_{ij}^{n_{ij}}$ account for the catalytic effect of some species (often H^+), the value n_{ij} is determined experimentally, and Ω is the saturation (ratio between ion activity product and equilibrium constant). Also p_i and q_i are determined experimentally. Factor ξ_k can be +1 or -1 depending on whether Ω_k is larger or smaller than 1 (precipitation or dissolution), respectively.

One possible definition of a reaction rate law is shown below.

```
<dataBase name ="kinetics">
        <reactionRateLaw name="calcite" type="reactionRateLawLasaga"
                            ea="0.0" precThreshold="1.0">
                <term ID="1" k="7.0e-2" p="1.0" q="1.0">
                        <ref type="species"  name="h+" n="0.5" />
                </term>
        </reactionRateLaw>
</dataBase>
```

The element `<dataBase>` is compulsory. The value of the attribute “name” can be chosen from the user. This element contains as many `< reactionRateLaw >` elements as reaction rate laws that need to be defined. The attributes of the `< reactionRateLaw >` elements are explained in the following table.

Element < reactionRateLaw type=” reactionRateLawLasaga”>		
Attribute	Description	Remarks
name	Name of the kinetic reaction that is associated to	Compulsory
ea	E_a of equation (E.5) Default value: 0.0	Optional
precThreshold	Threshold value for precipitation Default: 1.0	Optional

Inside the `< reactionRateLaw >` element, Nk `<term>` elements corresponding to the catalytic terms can be defined.

Element <term> of <reactionRateLaw type=” reactionRateLawLasaga”>			
Attribute	Possible values	Description	Remarks
ID	1, 2, 3, …	Term ID	Optional
k		k of equation (E.5)	Compulsory
p		p of equation (E.5) Default: 1.0	Optional
q		q of equation (E.5) Default: 1.0	Optional

Each `<term>` element can contain Ns `<ref type="species">` elements, one for every catalytic species.

Element <ref type=”species”> of < term >		
Attribute	Description	Remarks
name	Name of the catalyric species	Compulsory
n	n of equation (E.5) Default: zero	Optional

E.4.4) File “reactivemix.xml”

“reactivemix.xml” is the input file for ReactiveMix simulations (see Chapter 4 for details on the method). Its structure is shown below.

```
<reactiveMix>

   <inputFileCheproo name="cheprooplusplus.xml"/>

   <endMembers>
       <ref type="endMember" name="endmember1"/>
       <ref type="endMember" name="endmember2"/>
   </endMembers>

   <samples>
       <ref type="sample" name="Sample-1">
              <ref type="firstGuessMixingRatio" name="endmember1" value="0.75"/>
              <ref type="firstGuessMixingRatio" name="endmember2" value="0.25"/>

              <ref type="fixedMixingRatio" name="endmember1" value="0.75"/>
       </ref>
       <ref type="sample" name="Sample-2">
        …
       </ref>
   </samples>

</reactiveMix>
```

The element <reactiveMix> is compulsory.

Element < inputFileCheproo >		
Attribute	Description	Remarks
name	Name of CHEPROO++ input file with definition of the chemical system	Compulsory

The element <endMembers> contains the references to the names of the end-members.

Element <ref> of < endMembers >			
Attribute	Possible values	Description	Remarks
type	endMember		Compulsory
name		Name of the end-member. It must correspond to a ChemicalComposition defined in <inputFileCheproo>	Compulsory

The element <samples> contains the list of the samples. Each sample must be defined in an element <ref type =”sample”>, whose attributes are listed below.

Element <ref type =”sample”> of < samples >			
Attribute	Possible values	Description	Remarks
type	sample		Compulsory
name		Name of the sample. It must correspond to a ChemicalComposition defined in <inputFileCheproo>	Compulsory

For each sample, first-guess values of the mixing ratios corresponding to each end-member must be defined (in element <ref type =” firstGuessMixingRatio”>). Optionally, the user can fix values of the mixing ratios for some end-members in the element <ref type =” fixedMixingRatio”.

Element <ref type =” firstGuessMixingRatio”> of <ref type =”sample”>			
Attribute	Possible values	Description	Remarks
type	firstGuessMixingRatio		Compulsory
name		Name of the end-member the first-guess value refers to. It must correspond to a ChemicalComposition defined in <inputFileCheproo>	Compulsory
value	0 <= value <=1	Value of the mixing ratio first-guess	Compulsory

Optional Element <ref type =” fixedMixingRatio”> of <ref type =”sample”>			
Attribute	Possible values	Description	Remarks
type	fixedMixingRatio		Compulsory
name		Name of the end-member the fixed mixing ratio value refers to. It must correspond to a ChemicalComposition defined in <inputFileCheproo>	Compulsory
value	0 <= value <=1	Value of the fixed mixing ratio value	Compulsory