Dynamic modeling of aqueous electrolyte systems in Modelica

Andreas M. Bremen[a], Katharina M. Ebeling[a], Victor Schulte[a], Jan Pavšek[a], Alexander Mitsos[a,b,c]

[a] *Process Systems Engineering (AVT.SVT), RWTH Aachen University, 52074 Aachen, Germany*
[b] *Institute of Energy and Climate Research: Energy Systems Engineering (IEK-10), Forschungszentrum Julich GmbH, 52425 Julich, Germany*
[c] *JARA-ENERGY, 52056 Aachen, Germany*

Abstract

Modeling transient aqueous electrolyte systems is increasingly gaining attention for designing and optimizing processes where the equilibrium composition becomes relevant. This work revisits appropriate formulations to cope with embedding equilibrium conditions in transient systems. We follow the equation-oriented approach of writing balance equations in terms of reaction invariants and replacing the embedded Gibbs free energy minimization problem with a reformulation of the Karush-Kuhn-Tucker conditions to yield a system of differential-algebraic equations. We provide the open-source Modelica package `ElectrolyteMedia` for the modeling of transient aqueous electrolyte systems under consideration of chemical equilibrium in combination with detailed thermodynamic model equations for gas, liquid, and solid phases. With tailored initialization algorithms, we can integrate customized models using numerical solvers provided by a Modelica simulation environment, e.g., Dymola. We show simulation results of multiple case studies ranging from the simulation of titration experiments to complex unit models.

Keywords: Modelica, electrolyte thermodynamics, dynamic modeling, chemical equilibrium

---

1. Introduction

Aqueous electrolyte solutions are complex systems with numerous dissolved species that may dissociate and interact with potential gas and solid phases. Quantifying the species composition in these systems is relevant for designing and optimizing chemical processes. The mathematical representation of these systems involves nonlinear thermodynamic models and species concentrations varying by multiple orders of magnitude.

With generally fast rates of dissociation reactions and mass transfer at phase boundaries, the system can typically be considered at equilibrium, enabling the computation of equilibrium species concentrations by minimizing Gibbs free energy.

In addition, modeling the transient behavior of such aqueous electrolyte solutions requires equilibrium constraints on species compositions while adhering to prescribed dynamic mass and energy balances. At constant overall composition and thermodynamic state, the forward and reverse reaction rate of equilibrium reactions and mass transfer at phase boundaries are equal. Hence, the species composition remains constant. However, with imposed transient mass and energy balances, the forward and backward rates diverge (net reaction rate nonzero), leading to a change in species composition. Additionally, phases may disappear and reappear, resulting in the nonsmooth behavior of transient systems [1, 2].

Two competing approaches have been proposed to impose chemical equilibrium on a transient, complex multi-phase system: a sequential approach and an equation-oriented approach [3, 1, 2]. Alternative terminologies are the operator splitting technique and global implicit method, respectively [4]. In the sequential approach, the global solution of the Gibbs free energy minimization problem of a closed system at constant temperature, pressure, and overall composition leads to the correct number of phases and their species compositions at chemical equilibrium [3]. Transient systems with changing temperature, pressure, and overall composition globally minimize the Gibbs free energy at each time step, which may result in a high computational effort. Alternatively, the equation-oriented approach embeds the chemical equilibrium conditions as algebraic equations into the system of differential-algebraic equations (DAE) [5]. Here, the closed-system constraint is intrinsically given in the mass balance equations, while energy balance and constitutive equations determine temperature and pressure. This strategy allows straightforward model integration while adhering to the chemical equilibrium conditions. The chemical equilibrium conditions result from the Karush-Kuhn-Tucker (KKT) conditions that denote the first-order optimality criterion of the Gibbs free energy minimization problem. The KKT conditions are necessary and sufficient only in the case of convex Gibbs free energy of the underlying system [3].

Generally, solution approaches to minimizing the Gibbs free energy problem consider a stoichiometric and a nonstoichiometric formulation [6]. The stoichiometric formulation requires the manual selection of linearly independent reactions with stoichiometries satisfying the atom balance and the representation of Gibbs free energy in terms of the extent of reaction. Then, the first-order necessary conditions of the unconstrained minimization problem lead to a form of the law of mass action (LMA) equations. In contrast, the nonstoichiometric formulation considers the minimization of Gibbs free energy in terms of moles, subject to the closed-system constraint, e.g., in the form of the atom balance with the element-abundance matrix denoting the number of atoms in each species.

The stoichiometric and nonstoichiometric formulations become equivalent under the three following conditions [6]: First, the Gibbs free energy in terms of the extent of the reaction and the moles of species is convex. Second, the stoichiometric formulation considers all independent reactions of the present species. Third, the Lagrangian multipliers in the KKT conditions of the nonstoichiometric formulation cancel out by multiplying the derivative of the Lagrangian $\mathcal{L}$ with respect to moles $\mathbf{n}$ by the nullspace of the element-abundance matrix. The nullspace of the element-abundance matrix, in return, physically denotes a linear combination of the stoichiometry matrix in the stoichiometric formulation.

Abundant software applications can compute the equilibrium composition in aqueous electrolyte solutions. Extensive databases with equilibrium constants of reactions and mass transfer in aqueous electrolyte systems enable their use in software based on LMA to compute species equilibrium, e.g., WATEQ [7], EQ/3 [8], and PHREEQC [9], to name a few. A significant disadvantage of the LMA approach is that it only considers stable species with a nonzero mole amount, therefore, requiring stability tests [10]. More sophisticated software employs local, nonstoichiometric Gibbs free energy minimization, e.g., GEMS [11, 12] and Reaktoro [13]. This approach enables the computation of stable phase compositions for a predefined closed-system constraint and given temperature and pressure. In their solver, GEMS3K, Kulik et al. [11] employ an interior-point method to minimize the Gibbs free energy efficiently at given composition, temperature, and pressure. Leal et al. [14] provide the derivation of KKT conditions and the efficient solution of the nonlinear system of equations by perturbing the complementarity conditions of the KKT conditions for faster convergence. In a similar procedure, they provide a solution strategy for an LMA approach that results in the same formulation as the local Gibbs free energy minimization [10].

The efficient computation of Gibbs free energy in the software mentioned above is crucial. The goal is to enable fast convergence of reactive mass transfer systems, e.g., to compute phase assemblage and species concentration in subsurface reservoir simulations using temporal and spatial discretization [11]. The solution strategy for these systems commonly employs the sequential approach by solving mass balances and Gibbs free energy minimization sequentially. However, this strategy requires the step size to be sufficiently small to capture the influencing effects of both mass transfer and equilibrium.

For aqueous electrolyte systems, the formulation of an equation-oriented approach is generally considered inefficient [4]. Yeh and Tripathi [15] argue an equation-oriented approach that includes spatial discretization is impractical due to excessive CPU time. Instead, sequential procedures enable the simulation of reactive mass transfer problems solved with spatial and temporal discretization. However, the benefits of a sequential approach vanish in aqueous electrolyte systems that do not require a high number of spatial discretization nodes. Hence, the advantages of an equation-oriented formulation become apparent, as it enables control and dynamic optimization of such systems. Furthermore, from an engineering perspective, the set-up of additional constitutive equations, e.g., volume constraints, next to differential and algebraic equations that describe the equilibrium conditions is more intuitive than incorporating these in a sequential solution strategy.

This work introduces the open-source modeling framework `ElectrolyteMedia` in the object-oriented modeling language Modelica based on the equation-oriented approach for modeling transient aqueous electrolyte systems under consideration of reaction and phase equilibria. The framework enables the dynamic modeling of user-specific aqueous electrolyte systems coupled with custom balance equations. In the following, we present the general formulation of balance equations and the constitutive equations to denote chemical equilibrium. We discuss the necessary and sufficient conditions and underlying assumptions for the minimization of Gibbs free energy. We provide the general aspects of the implementation of the equation-oriented formulation within the `BaseProperties` model of the `Media` package within the Modelica Standard Library [16]. We demonstrate the framework in exemplary simulation studies, including titration simulations, an application in unit operation models, and a reactive mass transfer simulation. We close with concluding remarks on the applicability and limitations of this approach and future avenues for framework extensions.

2. Differential-algebraic model equations for aqueous electrolyte systems

In this Section, we derive the formulation of the equation-oriented representation of transient aqueous electrolyte systems. We formulate all equations on a mass basis and note that a molar basis works analogously. The reason for mass-based equations is the mass-based notation in Modelica [16] that becomes relevant in Section 3. We use the notation of Smith and Missen [6] to distinguish chemical species by their chemical formulae, and the phase in which they occur, e.g., $H_2O(g)$ and $H_2O(l)$ are distinct species.

2.1. Modeling of flows with equilibrium conditions

We consider two approaches to address the formulation of dynamic mass balance equations with underlying chemical equilibrium constraints. First, we write balance equations with an embedded Gibbs free energy minimization problem that we replace by its first-order optimality conditions in Section 2.1.1. Second, we revisit the approach of reformulating a high index DAE that incorporates species balance equations accounting for net reaction rates of equilibrium reactions in Section 2.1.2.

2.1.1. Embedded Gibbs free energy minimization

The DAE formulation to describe transient aqueous electrolyte systems with embedded equilibrium constraints originates from the work of Ploch et al. [17], who provide a thorough analysis of semi-explicit DAE with embedded optimization criteria with application in the formulation of metabolic reaction networks. First, we write the linearly independent atomic mass balances (Eq. (1a)) to obtain the correct amount of atom mass in each time step. The embedded optimization problem is then the minimization of Gibbs free energy (Eq. (1c)), subject to the closed-system constraint (Eq. (1d)), and nonnegativity of species mass (Eq. (1e)). The DAE with embedded equilibrium composition then reads

$$\begin{aligned}
&\dot{\tilde{\boldsymbol{m}}}(t) = \tilde{\boldsymbol{m}}_{\text{in}}(t) - \tilde{\boldsymbol{m}}_{\text{out}}(t), \qquad \tilde{\boldsymbol{m}}(0) = \tilde{\boldsymbol{m}}^0, && \text{(1a)}\\
&\boldsymbol{0} = \boldsymbol{g}(\tilde{\boldsymbol{m}}(t), \boldsymbol{m}_{\text{eq}}(t), \boldsymbol{y}(t)), && \text{(1b)}\\
&\boldsymbol{m}_{\text{eq}}(t) \in \arg \min_{\hat{\boldsymbol{m}}_{\text{eq}} \in \mathbb{R}^I} G(\hat{\boldsymbol{m}}_{\text{eq}}, T, p), && \text{(1c)}\\
&\quad \text{s.t.} \;\; \boldsymbol{0} = \boldsymbol{A}\hat{\boldsymbol{m}}_{\text{eq}} - \tilde{\boldsymbol{m}}(t), && \text{(1d)}\\
&\qquad\quad\;\; \boldsymbol{0} \leqslant \hat{\boldsymbol{m}}_{\text{eq}}, && \text{(1e)}
\end{aligned}$$

where $\tilde{\boldsymbol{m}} \in \mathbb{R}^J$ and $\boldsymbol{m}_{\text{eq}} \in \mathbb{R}^I$ are the atomic mass vector and the species mass vector in equilibrium, respectively. J is the number of elements in the considered system and I is the number of chemical species. $\tilde{\boldsymbol{m}}_{\text{in}} \in \mathbb{R}^J$ and $\tilde{\boldsymbol{m}}_{\text{out}} \in \mathbb{R}^J$ denote the sum of entering and exiting streams, respectively. $\tilde{\boldsymbol{m}}^0 \in \mathbb{R}^J$ is the initial element mass vector in the system. Constitutive equations are given in Eq. (1b) with $\boldsymbol{y}$ being further algebraic variables, e.g., to determine feed flow rates or to impose volume constraints ($\dim(\boldsymbol{g}) = \dim(\boldsymbol{y})$). G is the Gibbs free energy of the entire system with fixed T and p denoting temperature and pressure, respectively. $\hat{\boldsymbol{m}}_{\text{eq}}$ denotes the equilibrium species mass vector whose elements are the decision variables in the minimization problem. $\boldsymbol{A} \in \mathbb{R}^{J \times I}$ is the mass-based element abundance matrix required for the closed-system constraint of the embedded optimization problem.

To cope with the embedded minimization problem in Eqs. (1c)-(1e), we can apply the Karush-Kuhn-Tucker (KKT) conditions [6]. The embedded optimization problem satisfies the linear independence constraint qualification [18]. Hence, with the constraint qualification, the KKT conditions are generally a necessary first-order optimality criterion. They are also sufficient if the underlying minimization problem is convex. The closed-system constraint in Eq. (1d) is linear, and the Gibbs free energy functions of each phase (aqueous, solid, or gaseous) in electrolyte thermodynamics are typically convex. We argue that a nonconvex Gibbs free energy function would result in a potential phase split. Only in the case of two-phase equilibrium systems, e.g., liquid-liquid equilibrium, where one Gibbs free energy model describes the heterogeneous nature of the system, do the Gibbs free energy functions become nonconvex, e.g., [2]. However, in modeling aqueous electrolyte systems, each phase is described by a distinct Gibbs free energy model.

Substituting the embedded optimization problem, Eqs. (1c) - (1e), by the KKT conditions and some further reformulation yields [17, 10]

$$\begin{aligned}
\dot{\tilde{\boldsymbol{m}}}(t) &= \tilde{\boldsymbol{m}}_{\text{in}}(t) - \tilde{\boldsymbol{m}}_{\text{out}}(t), & \tilde{\boldsymbol{m}}(0) &= \tilde{\boldsymbol{m}}^0, && \text{(2a)}\\
\mathbf{0} &= \boldsymbol{g}(\tilde{\boldsymbol{m}}(t), \boldsymbol{m}_{\text{eq}}(t), \boldsymbol{a}(t), \boldsymbol{K}(t), \boldsymbol{y}(t)), &&&& \text{(2b)}\\
\mathbf{0} &= \boldsymbol{A}\boldsymbol{m}_{\text{eq}}(t) - \tilde{\boldsymbol{m}}(t), &&&& \text{(2c)}\\
\mathbf{0} &= \boldsymbol{\nu}\left(\ln \boldsymbol{a}(t) - \boldsymbol{z}(t)\right) - \ln \boldsymbol{K}(t), &&&& \text{(2d)}\\
\mathbf{0} &= \boldsymbol{z}(t)\boldsymbol{m}_{\text{eq}}(t), &&&& \text{(2e)}\\
\mathbf{0} &\leqslant \boldsymbol{z}(t), \boldsymbol{m}_{\text{eq}}(t), &&&& \text{(2f)}
\end{aligned}$$

where $\boldsymbol{\nu} \in \mathbb{R}^{R \times I}$ is the mass based stoichiometry matrix of independent reactions and mass transfer with $R = I - J$ being the number of independent reactions. $\boldsymbol{z}$ are reformulated Lagrange multipliers, $\boldsymbol{a}$ are the species activities, $\boldsymbol{K}$ are equilibrium constants of equilibrium reactions and mass transfer. We refrain from explicitly writing constitutive equations for activities and equilibrium constants, and incorporate them in Eq. (2b). Eq. (2d) represents a relaxed form of logarithmic law of mass action that remains valid for vanishing species mass [2, 5, 10]. The reformulated Lagrangian multipliers $\boldsymbol{z}$ become nonzero in case of species disappearance (Eq. (2e)), and then relax the logarithmic law of mass action equations in Eq (2d). In summary, DAE (2) is of index-1, semi-explicit and equivalent to DAE (1) with the embedded minimization of Gibbs free energy.

2.1.2. High index DAE formulation

An alternative approach considers the equilibrium in terms of equilibrium reactions in the differential balance equations. The mass action equations implicitly provide information on these equilibrium reaction rates. This approach originates from Moe et al. [19] and is applied to electrolyte systems by Kakhu and Pantelides [5]. The semi-explicit DAE consists of differential equations (Eq. (3a)), the logarithmic law of mass action

(Eq. 3b), and further potential algebraic equations (Eq. (3c)):

$$\dot{\boldsymbol{m}}_{\mathrm{eq}}(t) = \boldsymbol{m}_{\mathrm{in}}(t) - \boldsymbol{m}_{\mathrm{out}}(t) + \boldsymbol{\nu}^{\intercal}\boldsymbol{r}(t), \qquad \boldsymbol{m}_{\mathrm{eq}}(0) = \boldsymbol{m}_{\mathrm{eq}}^{0}, \tag{3a}$$

$$\boldsymbol{0} = \boldsymbol{\nu} \ln \boldsymbol{a}(t) - \ln \boldsymbol{K}(t), \tag{3b}$$

$$\boldsymbol{0} = \boldsymbol{g}(\boldsymbol{m}_{\mathrm{eq}}(t), \boldsymbol{a}(t), \boldsymbol{K}(t), \boldsymbol{y}(t)), \tag{3c}$$

where $\boldsymbol{r} \in \mathbb{R}^{R}$ is the vector of net rates of reactions and mass transfer. These rates are algebraic variables and only appear in the differential equations. Hence, the algebraic equations cannot be solved for $\boldsymbol{r}$. Therefore, DAE (3) is smooth, but its differential index is greater than one.

To obtain a DAE of differential index-1 that may be solved with common numerical solvers [20], three options are available: (1) replacing algebraic variables $\boldsymbol{r}$ with the derivative of the extent of reaction ($\boldsymbol{r} = \dot{\boldsymbol{\xi}}$), (2) index reduction via differentiation [21], and (3) reformulation in terms of reaction invariants [19]. The first option renders a higher number of differential states and results in an implicit DAE that can be a challenge for some numerical solvers [20]. The second option requires the differentiation of thermodynamic equations that can be costly, but the knowledge of reaction rates and mass transfer rates becomes apparent. However, the replacement of algebraic variables and the differentiation of algebraic equations still requires the consistent choice of initial conditions that fulfill the law of mass action equations. These drawbacks make the application of the differentiation of algebraic equations and the replacement by differential variables rather difficult in a generalized formulation.

Consequently, we decide on reformulating differential equations in terms of reaction invariants. This approach leads to a reduced number of differential equations while setting the initial conditions is straightforward. To do so, we multiply the differential equations in Eq. (3a) with the nullspace $\boldsymbol{\lambda} \in \mathbb{R}^{I \times J}$ of the mass-based stoichiometry matrix, for which $\boldsymbol{\lambda}^{\intercal}\boldsymbol{\nu}^{\intercal} = \boldsymbol{0}$. Hence, this multiplication eliminates the reaction rates $\boldsymbol{r}$ from the differential equations ($\boldsymbol{\lambda}^{\intercal}\boldsymbol{\nu}^{\intercal}\boldsymbol{r} = 0$). The linear correlation of reaction invariants and full species mass vector in Eq. (4c) is a new (nontrivial) algebraic equation introduced to the DAE by this approach. It complements the set of equations to an index-1 DAE:

$$\dot{\bar{\boldsymbol{m}}}(t) = \boldsymbol{\lambda}^{\intercal}(\boldsymbol{m}_{\mathrm{in}}(t) - \boldsymbol{m}_{\mathrm{out}}(t)), \qquad \bar{\boldsymbol{m}}(0) = \boldsymbol{\lambda}^{\intercal}\boldsymbol{m}^{0}, \tag{4a}$$

$$\boldsymbol{0} = \boldsymbol{\nu} \ln \boldsymbol{a}(t) - \ln \boldsymbol{K}(t), \tag{4b}$$

$$\boldsymbol{0} = \boldsymbol{\lambda}^{T}\boldsymbol{m}_{\mathrm{eq}}(t) - \bar{\boldsymbol{m}}(t), \tag{4c}$$

$$\boldsymbol{0} = \boldsymbol{g}(\bar{\boldsymbol{m}}(t), \boldsymbol{m}_{\mathrm{eq}}(t), \boldsymbol{a}(t), \boldsymbol{K}(t), \boldsymbol{y}(t)). \tag{4d}$$

We note that an identical formulation results from a time-scale analysis of reaction rates [22, 23, 24]. The net rates of equilibrium reaction and mass transfer are fast and cancel out using singular perturbation theory. The obtained algebraic equations then denote the law of mass action equations.

2.1.3. Formulation similarity

Although DAE (4) inherently comprises information on the reaction equilibrium, we observe a structural similarity to DAE (2), i.e., the flow equations with the embedded optimization criteria. However, the meaning of the variables differ: First, DAE (2) incorporates differential equations with physical meaning, i.e., they balance the atomic mass $\tilde{\boldsymbol{m}}$. This is required for the closed system constraint in Eq. (2c). In contrast, Eq. (4a) balances a linear combination of the atomic mass $\bar{\boldsymbol{m}}$ that is computed solely from the nullspace of the mass based stoichiometry matrix $\boldsymbol{\nu}$. In fact, the mass-based element abundance matrix $\boldsymbol{A}^\intercal$ in DAE (2) meets the requirement for the nullspace of the mass-based stoichiometry matrix $\boldsymbol{\nu}$, i.e., $\boldsymbol{A}\boldsymbol{\nu}^\intercal = \mathbf{0}$. Second, DAE (4) may become singular in case of vanishing species masses, e.g., in case of complete dissolution of a mineral phase in an aqueous solution. This is mitigated in DAE (2) with the reformulated Lagrangian multipliers $\boldsymbol{z}$ and the complementarity conditions comprising Eqs. (2e) and (2f). In other words, z_i allows for the corresponding $m_{\mathrm{eq},i}$ to become zero.

However, DAE (2) is nonsmooth, as the embedded KKT conditions allow the nonnegativity constraints denoted by the complementarity conditions in Eqs. (2e) and (2f) to become active/inactive at any point in time. Therefore, DAE may be ill-posed without reformulation. Especially in combination with aqueous concentrations varying by magnitudes and nonlinear thermodynamic models, we found the DAE integration to fail in multiple cases, as the Jacobian is close to singularity.

2.2. Smooth approximation

We follow a smoothing approach, i.e., we regularize the complementarity conditions with a small parameter $\tau > 0$ that leads to a smooth DAE by substituting Eq. (2e) by [25, 10]

$$0 = z_i(t) m_{\mathrm{eq},i}(t) - \tau, \quad i = 1, \ldots, I \tag{5}$$

We reformulate Eq. (5) to the smoothed Fischer-Burmeister function [26, 27] to enforce the nonnegativity of $m_{\mathrm{eq},i}$ and z_i to hold (Eq. (2f)):

$$0 = z_i(t) + m_{\mathrm{eq},i}(t) - \sqrt{z_i(t)^2 + m_{\mathrm{eq},i}(t)^2 + 2\tau}, \quad i = 1, \ldots, I \tag{6}$$

This reformulation is equivalent to the regularized complementarity conditions (Eq. (5)) and leads to a well-posed DAE with smoothed complementarity conditions [28]. Yet, in aqueous electrolyte systems, the species masses can differ by multiple orders of magnitude. Therefore, the reformulation of complementarity constraints can still lead to failing DAE integration due to small species masses, e.g., $m_{\mathrm{eq},\mathrm{H}^+}$ becomes small in alkaline solutions. As a remedy, from a physical point of view, using an educated guess, we can decide a priori which species we expect to be present at all times. This assumption simplifies DAE (2) by eliminating the Langrangian multipliers and complementarity conditions for species i being always present.

To make use of the benefits of each formulation, we combine the two DAE formulations (2) and (4) by (i) writing balance equations in terms of reaction invariants that are a generalization of the atomic mass balance, and (ii) eliminating the complementarity conditions for species that are present at all times. From a Gibbs free energy minimization perspective, eliminating complementarity conditions of some species is equal to neglecting the nonnegativity constraints of these species in the embedded optimization problem in

Eqs. (1c) - (1e). In summary, we consider the following DAE:

$$\begin{aligned}
\dot{\bar{\boldsymbol{m}}}(t) &= \bar{\boldsymbol{m}}_{\text{in}}(t) - \bar{\boldsymbol{m}}_{\text{out}}(t), \qquad \bar{\boldsymbol{m}}(0) = \boldsymbol{\lambda}^{\intercal}\boldsymbol{m}^{0}, && \text{(7a)}\\
\boldsymbol{0} &= \boldsymbol{\nu}\left(\ln \boldsymbol{a}(t) - \boldsymbol{D}\boldsymbol{z}(t)\right) - \ln \boldsymbol{K}(t), && \text{(7b)}\\
\boldsymbol{0} &= \boldsymbol{\lambda}^{T}\boldsymbol{m}_{\text{eq}}(t) - \bar{\boldsymbol{m}}(t), && \text{(7c)}\\
0 &= z_i(t) + m_{\text{eq},i}(t) - \sqrt{z_i(t)^2 + m_{\text{eq},i}(t)^2 + 2\tau}, \quad i = 1, \ldots, I, && \text{(7d)}\\
\boldsymbol{0} &= \boldsymbol{g}(\bar{\boldsymbol{m}}(t), \boldsymbol{m}_{\text{eq}}(t), \boldsymbol{a}(t), \boldsymbol{K}(t), \boldsymbol{y}(t)), && \text{(7e)}
\end{aligned}$$

where $\boldsymbol{D}$ is a diagonal matrix for which holds

$$D_{ii} = \begin{cases} 1 & \text{if } m_{\text{eq},i} \geqslant 0, \\ 0 & \text{if } m_{\text{eq},i} > 0, \end{cases} \qquad i = 1, \ldots, I. \tag{8}$$

2.3. DAE initialization

The initialization of a DAE involves setting initial conditions of differential variables and initial guesses of algebraic variables. In DAE (7), the initial conditions $\bar{\boldsymbol{m}}(0)$ are calculated from an initial species mass vector $\boldsymbol{m}^0$ with species masses initially present in the system. These species masses do not need to be in equilibrium but rather denote the amount of each species in its initially undissociated, solid, liquid, or gaseous state.

With given initial conditions, the DAE integrator needs to solve a nonlinear system of equations at $t = 0$ to start the DAE integration. Modelica is capable of solving this problem by employing a Newton solver. However, it is likely to fail in the case of variables being close to zero. In other words, an initialization failure is expected for species concentrations in the liquid phase and in the vicinity of disappearing or reappearing chemical species. The iterative procedure allows negative values of these variables, which makes the initialization nonphysical. Additionally, logarithmic mathematical expressions in thermodynamic models, i.e., for the excess Gibbs free energy, do not allow negative values within Modelica's Newton iterations to solve the nonlinear system of algebraic equations. Hence, the initialization of aqueous electrolyte system models commonly fails without providing sufficiently good start values. In the following, we distinguish between two cases. The first case is the calculation of the equilibrium composition at given temperature and pressure. We then extend this method to the case when either temperature or pressure is unknown, but instead a further intensive thermodynamic property is given, i.e., enthalpy, entropy, or density.

One method to set start values is to provide start values by hand for each specific simulation. However, this requires the user to have expert knowledge of the considered system. Instead, a computational approach allows the calculation of the species equilibrium composition by providing only the species masses $\boldsymbol{m}^0$ initially added to the system. We can distinguish between two methods for the computation of the equilibrium composition: A dynamic approach and an iterative approach. The dynamic approach [29] considers a DAE with dynamic mass balances for each chemical species and net reaction rates that let the system converge into the equilibrium composition at a steady state. In other words, the dynamic simulation converges toward a steady state that denotes the equilibrium composition. We applied this approach in previous work on modeling direct mineral carbonation [30]. Still, we found it cumbersome because it requires a separate DAE simulation to find the equilibrium composition. Alternatively, an iterative approach is more suitable for computing the equilibrium composition, as we can incorporate this procedure directly into the model to provide initial guesses. We can use the numerous iterative procedures applied in the sequential approach to model transient aqueous electrolyte systems [11, 12, 10, 14]. In detail, we adopt the approach by Leal et al. [10], who compute equilibrium compositions at constant temperature and pressure via the minimization of Gibbs free energy by applying KKT conditions and solving the underlying system of equations with an interior-point method [18]. The derivation of nonlinear equations is similar to Eqs. (2c)-(2e), however, with a mole based closed system constraint and regularized complementarity conditions:

$$\mathbf{0} = \boldsymbol{\lambda}_{\mathrm{m}} \boldsymbol{n}_{\mathrm{eq}} - \tilde{\boldsymbol{n}} \tag{9a}$$

$$\mathbf{0} = \boldsymbol{\nu} \left(\ln \boldsymbol{a} - \boldsymbol{z} \right) - \ln \boldsymbol{K} \tag{9b}$$

$$\tau = z_i n_{\mathrm{eq},i}, \qquad i = 1, \ldots, I \tag{9c}$$

where $\tilde{\boldsymbol{n}} = \boldsymbol{\lambda}_{\mathrm{m}} \boldsymbol{n}^0$ with $\boldsymbol{\lambda}_{\mathrm{m}}$ denoting the nullspace of the molar stoichiometric matrix $\boldsymbol{\lambda}_{\mathrm{m}}$. The unknowns in Eq. (9) to be solved for are the equilibrium species moles $\boldsymbol{n}_{\mathrm{eq}}$ and the reformulated Lagrangian multipliers $\boldsymbol{z}$. In line with Leal et al. [10], we apply a Newton method with an approximation of the logarithmic activity derivative in each iteration to be

$$\frac{\partial \ln(a_i)}{\partial n_i} \approx \frac{\partial \ln(y_i)}{\partial n_i} = \frac{1 - y_i}{n_i} \qquad i = 1, \ldots, I, \tag{10}$$

where y_i is the molar fraction of species i in its corresponding phase. Furthermore, in each iteration, the Newton step of the reformulated Lagrangian multipliers $\Delta \boldsymbol{z}$ can be calculated explicitly, reducing the number of unknowns to the number of chemical species I. This reformulation requires the Newton solver only to solve for species moles $\boldsymbol{n}_{\mathrm{eq}}$. Additionally, we limit the step size in each Newton iteration to keep species moles $n_{\mathrm{eq},i}$ strictly positive. We refer to Leal et al. [10] for further detail on the algorithm.

The iterative procedure for the equilibrium calculation requires fixed temperature and pressure. However, in a transient aqueous electrolyte model, temperature and pressure are not always known at $t = 0$. Instead, next to the initial species moles $\boldsymbol{n}^0$, a combination of two intensive thermodynamic properties defines the thermodynamic state of the system. The Modelica *Media* library requires the combinations (1) temperature-pressure, (2) pressure-enthalpy, (3) pressure-entropy, and (4) temperature-density as input variables for the initialization of a DAE. Therefore, we extend the initialization with given temperature and pressure to be applicable for the combinations of the intensive thermodynamic properties (2)-(4). Following Smith and Missen [6], we use an inner-outer algorithm to find the equilibrium composition at given intensive thermodynamic properties. Alternative simultaneous procedures solve the composition and temperature/pressure in one instant, but they reportedly suffer from convergence difficulties [6]. Here, we guide the inner-outer algorithm for given pressure and enthalpy and note that the approach works analogously for other combinations of intensive thermodynamic properties.

First, we set the temperature to a feasible value T^0 at a given pressure and solve the equilibrium composition. In the next step, we alternately calculate the temperature T^{k+1} at the given composition $\boldsymbol{x}^k$ using a Newton-Raphson procedure (outer algorithm) and calculate a new composition $\boldsymbol{x}^{k+1}$ at T^{k+1} (inner algorithm). The Newton-Raphson method calculates the enthalpy difference

$$f(T^k) = h(T^k, p, \boldsymbol{x}^k) - h \tag{11}$$

and performs an iteration step

$$T^{k+1} = T^k - \frac{f(T^k)}{f'(T^k)} = T^k - \frac{h(T^k, p, \boldsymbol{x}^k) - h}{c_p(T^k, p, \boldsymbol{x}^k)}, \tag{12}$$

where c_p is the heat capacity of the underlying system. The final temperature and composition is given, once $f(T^k) < \epsilon$ with ϵ being a sufficiently small value.

Next to the initial guesses for the initial composition and temperature/pressure of the aqueous electrolyte system, the modeler can provide further initial guesses manually. Otherwise, Modelica uses generic start values and performs its own Newton iteration scheme for the initialization of the DAE before starting integration. We provide robust procedures for initializing the DAE with embedded equilibrium conditions with these iterative methods.

3. `ElectrolyteMedia` library

We introduce the `ElectrolyteMedia` library that implements the methods presented in Section 2 for transient modeling of aqueous electrolyte systems in multiple `Media` libraries that extend the `Media` package of the Modelica Standard Library. It is available as an open-source third-party Modelica library [31], also available at `http://permalink.avt.rwth-aachen.de/?id=807420` under the 3-clause BSD license. The object-oriented package provides a standardized structure for calculating intensive thermodynamic properties of a fluid medium incorporating thermodynamic models and equilibrium conditions of aqueous electrolyte systems. The intensive thermodynamic properties of the `Media` libraries find use in component models comprising mass and energy balances. Modelica provides generic models containing mass and energy balances in the `Fluid` library of the Modelica Standard Library. Alternatively, user-specific models with customized mass and energy balances can be implemented and interfaced with the `Media` package models to calculate intensive thermodynamic properties.

Distinct thermodynamic models for each aggregate state (gas, liquid, solid) provide the basis for calculating the thermodynamic properties of chemical species. We implement these as Modelica functions with explicit, intensive thermodynamic properties as inputs. The gas phase considers the ideal gas model with a temperature-dependent heat capacity and the Peng Robinson equation of state [32]. Within the liquid phase, we distinguish between solutes and solvent water. The IF97 formulation readily provides water properties implemented as a separate medium within the Modelica Standard Library [33]. We implement the revised Helgeson-Kirkham-Flowers model for the thermodynamic properties of solute species at infinite dilution [34]. Excess Gibbs free energy models, namely the extended Debye-Hückel model, the Bromley model [35], and the Pitzer model [36], account for the nonideal contributions with increasing ionic strengths. Solid species properties are assumed to be ideal (no consideration of solid mixtures) based on an equation of state and a heat capacity polynomial [37]. The database `spronshp.dat` provides the parameter values for the ideal gas model, the revised Helgeson-Kirkham-Flowers model, and the solid properties model used in the SUPCRTBL software for the computation of single species thermodynamic properties [38].

Based on these thermodynamic property functions, different media for the gas phase (`GasPhase`), the liquid phase (`LiquidPhase`), and the solid phase (`SolidPhase`), each consisting of multiple species, are implemented. Within the gas phase and the solid phase, we do not consider any reaction equilibrium. We use the single-phase property functions to extend multi-phase media, namely `GasLiquidPhase` and `SolidLiquidPhase`. The `LiquidPhase` and `GasLiquidPhase` media assumes all species to be nonnegative, i.e., $D_{ii} = 0$ for all chemical species i. The `SolidLiquidPhase` package enables solid species to disappear and reappear, i.e., $D_{ii} = 1$ for all solid species i, whereas liquid species are always present. The object-oriented nature of the Modelica framework allows these media extensions for user-specific models.

A user interface enables setting all required parameters to fully specify a medium. In detail, the modeler sets the chemical species and their number in each phase. Furthermore, the modeler selects thermodynamic property models for the gas and liquid phases. Their corresponding parameters for the thermodynamic models are stored in a Modelica `record` structure. Furthermore, the atomic stoichiometry matrix $\boldsymbol{\nu}_m$, and a reference composition, temperature, and pressure for generic initialization are required. We refer to the User Guide of the `ElectrolyteMedia` package for further details on setting up a user-specific medium.

A fully-defined medium allows instantiating the `BaseProperties` model in a user-specific unit model containing the mass and energy balances. The mass balance equations need to be written in terms of reaction invariants as in Eq. (7a). The algebraic equations (7b) and (7d) in combination with further constitutive equations, e.g., for the calculation of specific enthalpy, density, and mole fractions, are given in the `BaseProperties` model. The `BaseProperties` model requires the normalized reaction invariant vector and two further intensive thermodynamic properties, e.g., temperature and pressure, specified in the unit model.

This object-oriented implementation incorporating a database with a wide range of chemical species allows for efficient modeling of transient aqueous electrolyte systems. Furthermore, the equation-oriented approach allows for dynamic optimization and control of chemical systems and processes that involve aqueous electrolyte systems. The framework is not limited to the implemented media packages and extensions to different thermodynamic property models, and alternative databases are possible.

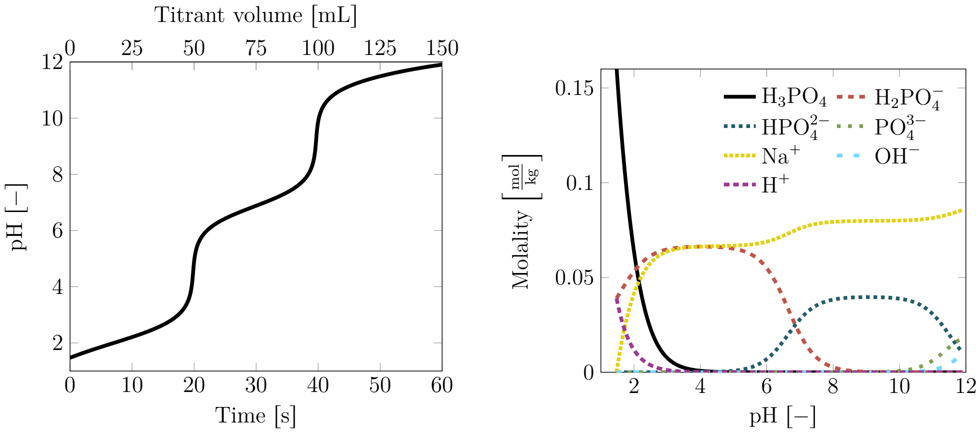
4. Application of `ElectrolyteMedia` in case studies

In this Section, we provide illustrative examples to show the applicability of the object-oriented framework for the dynamic modeling of aqueous electrolyte systems. We offer three case studies with increasing unit model complexity and the number of considered embedded equilibrium systems. First, we show a simulation of a laboratory-scale titration in Section 4.1. Subsequently, we underline the framework's capabilities combined with a Modelica test model based on the `Fluid` library by using its unit models in Section 4.2. Finally, in Section 4.3, we provide a computationally expensive reactive transport model with many embedded equilibrium conditions due to the discretization of partial differential balance equations.

Within this study, all simulations are performed using Dymola 2020 [39]. For integration, we use the integrator LSODAR [40] with an integration tolerance of 10^{-8}.

4.1. Titration simulation

We model the conventional set-up of a laboratory-scale titration with an open volume and a constant feed flow rate representing the beaker containing the analyte and the titrant being dosed in by a burette, respectively. We consider the titration of a phosphoric acid (H_3PO_4) solution of $b_{H_3PO_4} = 0.2\,\mathrm{mol/kg_{solvent}}$ with an initial volume of $V^0 = 25\,\mathrm{mL}$ using a sodium hydroxide (NaOH) solution of $b_{NaOH} = 0.1\,\mathrm{mol/kg_{solvent}}$ at a feed flow rate of $\dot{m}_{\mathrm{titrant}} = 25\,\frac{\mathrm{mg}}{\mathrm{s}}$. The simulation is performed at ambient temperature ($T = 25\,°\mathrm{C}$) and ambient pressure ($p = 1\,\mathrm{bar}$). We use the `LiquidPhase` medium package, as the system is in the liquid phase only. We consider the extended Debye-Hückel model to account for nonideality. Phosphoric acid is a triprotic acid. Hence, the liquid phase contains the chemical species H_3PO_4, $H_2PO_4^-$, HPO_4^{2-}, PO_4^{3-}, Na^+, OH^-, H^+, and H_2O. The temporal course of the pH and the species molalities are shown in Figs. 1a and 1b, respectively. The simulation results predict the dissociation state over the course of the experiment with deprotonation of phosphoric species with increasing pH.



(a) pH over time and titrant volume.

(b) Species molalities over pH.

Figure 1: Titration of a $25\,\mathrm{mL}$ solution of $0.2\,\mathrm{mol/kg_{solvent}}$ phosphoric acid with a titrant of $0.1\,\mathrm{mol/kg_{solvent}}$ NaOH. The titrant feed increases the pH resulting in the dissociation of phosphoric acid.

4.2. Modelica `Media` test model simulation

To show the modeling capabilities in combination with the Modelica `Fluid` library, we consider a Modelica test model that comprises a system of constant volume with $V = 1\,\mathrm{m}^3$, one constant inlet mass flow rate, and one outlet modeled as a pipe model (cf. Fig. 2). Here, we assume an ideally mixed gas-liquid phase and hence consider the `GasLiquidPhase` package. The gas phase contains H_2O, CO_2, and O_2, and the liquid phase comprises CO_2, HCO_3^-, CO_3^{2-}, Na^+, Cl^-, OH^-, H^+, and H_2O. Gaseous CO_2 dissolves in the liquid phase and dissociates into HCO_3^- and CO_3^{2-}, whereas O_2 is only present in the gaseous phase. Gaseous H_2O is in equilibrium with liquid H_2O. The volume is initially composed of a gas phase with 80 wt.% CO_2 and 20 wt.% O_2, and a liquid phase containing 1 mol/kg$_{\text{solvent}}$ NaCl that is fully dissociated. The system is initially composed of 80 wt.% liquid phase. The volume is initially at ambient conditions, i.e., $T = 25\,^\circ\text{C}$ and $p = 1\,\text{bar}$. The constant feed flow rate of $1\,\frac{\text{kg}}{\text{s}}$ is composed of the same gas phase composition, and the same liquid phase fraction, but the liquid phase is composed of a solution containing 0.1 mol/kg$_{\text{solvent}}$ NaOH at a temperature of $T = 50\,^\circ\text{C}$.

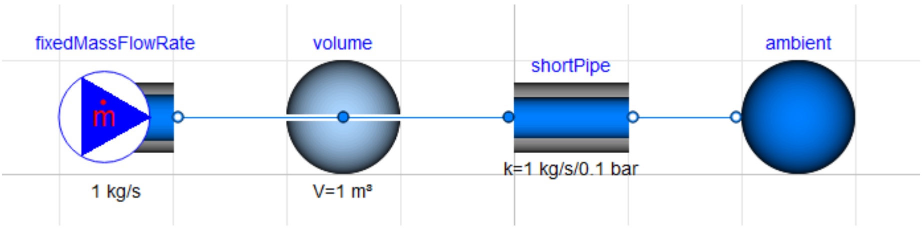
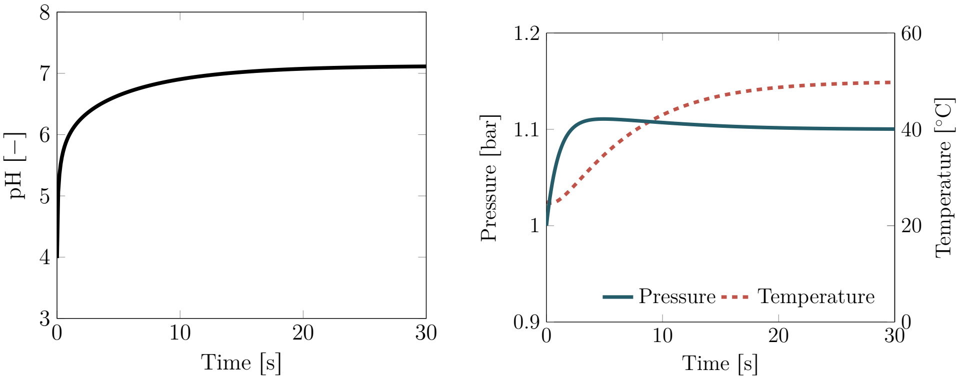


Figure 2: Illustration of a Modelica model based on the `Fluid` library for media testing. The model comprises a fixed mass flow rate into a constant volume with an outlet consisting of a pipe model.

Throughout the simulation, the composition and temperature in the volume asymptotically approach the specifications of the constant inlet mass flow rate. Fig. 3 shows that the alkaline feed increases the pH of the initially acidic liquid phase of the volume. The temperature increases from initially 25 °C to 50 °C, and the pipe imposes a pressure difference of 0.1 bar resulting in an overall pressure of $p = 1.1\,\text{bar}$. The simulation shows that the coupling of complex unit models is feasible, thus justifying the approach of embedding the equilibrium constraints within the algebraic equations of the medium model.



(a) pH in the liquid phase over time.

(b) Pressure and temperature over time.

Figure 3: Simulation results of the constant volume show asymptotic approaching of feed flow rate conditions with a pressure difference of 0.1 bar due to the pipe at the volume outlet.

4.3. Reactive transport simulation

The previous case studies each involve one embedded equilibrium problem to be solved in combination with suitable balance equations. These case studies underline the general applicability in small-scale modeling problems. In contrast, transient reactive transport models incorporate many embedded equilibrium problems as the underlying differential equations are spatial discretizations for numerical integration. In this section, we show the capabilities and limits of the DAE formulation by employing a reactive transport simulation with a total of 20 embedded equilibrium problems.

We consider the exemplary case study of Leal et al. [10] who simulate the injection of a CO_2 saturated brine into a porous rock with an initial volumetric composition of 98 % quartz (SiO_2) and 2 % calcite ($CaCO_3$) and an initial porosity of $\phi = 50\,\%$. The brine constitutes $0.9\,\mathrm{mol/kg_{solvent}}$ NaCl, $0.05\,\mathrm{mol/kg_{solvent}}$ $MgCl_2$, $0.01\,\mathrm{mol/kg_{solvent}}$ $CaCl_2$, and $0.75\,\mathrm{mol/kg_{solvent}}$ CO_2. The brine flows through the porous rock at a temperature of $T = 60\,^\circ\mathrm{C}$ and a pressure of 100 bar with a Darcy velocity of $u_D = 1.2 \times 10^{-5}\,\frac{\mathrm{m}}{\mathrm{s}}$. With the given brine composition, $CaCO_3$ is expected to dissolve allowing the formation of dolomite ($MgCa(CO_3)_2$). Hence, we consider the `SolidLiquidPhase` package, where pure solid phases may disappear and reappear. Considered solid phases are quartz, calcite and dolomite and the species in the liquid phase are CO_2, HCO_3^-, CO_3^{2-}, Mg^{2+}, Ca^{2+}, Na^+, Cl^-, OH^-, H^+, and H_2O.

We only consider the convective liquid flow and neglect potential dispersion and diffusion effects. The aim is only to show the dynamic computation of the equilibrium composition in a large-scale problem. Furthermore, dispersion and diffusion effects may lead to undesired oscillation during integration due to discretization. The partial differential mass balance of reaction invariants then reads

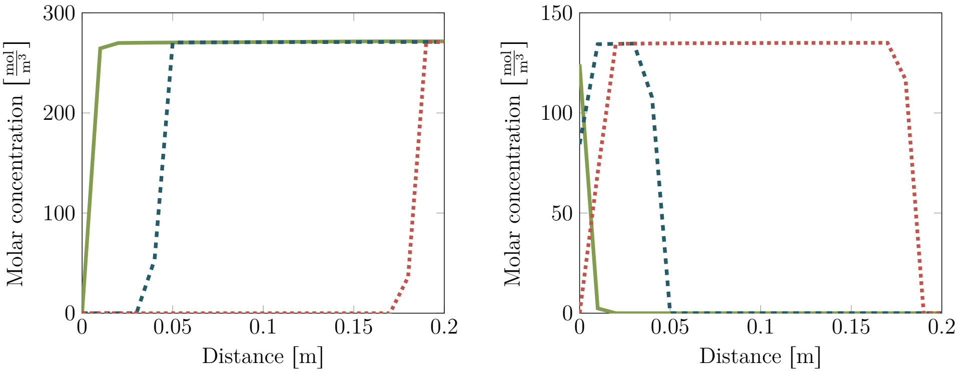
$$\frac{\partial \bar{\boldsymbol{m}}}{\partial t} + \nabla \cdot \bar{\boldsymbol{m}}\boldsymbol{u} = \boldsymbol{0} \tag{13}$$

We assume a one-dimensional flow and discretize Eq. (13) with finite volumes to obtain reaction invariant balance equations

$$\begin{aligned}
&\dot{\bar{\boldsymbol{m}}}_k(t) = \bar{\boldsymbol{m}}_{\mathrm{in},k}(t) - \bar{\boldsymbol{m}}_{\mathrm{out},k}(t) && k = 1,\dots,K && \text{(14a)}\\
&\bar{\boldsymbol{m}}_k(0) = \boldsymbol{\lambda}^\intercal \boldsymbol{m}_k^0 && k = 1,\dots,K && \text{(14b)}\\
&\bar{\boldsymbol{m}}_{\mathrm{out},k}(t) = u_k(t) A \phi_k(t) \rho_k^{\mathrm{l}}(t) \boldsymbol{\lambda}^\intercal \boldsymbol{x}(t) && k = 1,\dots,K && \text{(14c)}\\
&\bar{\boldsymbol{m}}_{\mathrm{in},k}(t) = \bar{\boldsymbol{m}}_{\mathrm{out},k-1}(t) && k = 2,\dots,K && \text{(14d)}\\
&\dot{\bar{\boldsymbol{m}}}_{\mathrm{in},1} = u_D A \rho_{\mathrm{in}}^{\mathrm{l}} \boldsymbol{\lambda}^\intercal \boldsymbol{x}_{\mathrm{in}} && && \text{(14e)}
\end{aligned}$$

where A is the cross-sectional area. u_k, ϕ_k, and ρ_k^{l} denote the liquid flow velocity, the porosity, and the liquid density of volume k, respectively. K is the number of volume elements resulting from discretization. Then, Eqs. (14) and (7b) - (7e) denote the full DAE to describe the reactive transport model that includes the equilibrium conditions (Eq. (7b)), the Fischer-Burmeister function (Eq. (7d)), and the volume constraint of each element and further constitutive equations (Eq. (7e)). In this case, we assume all liquid species to be present at all times, and solid species may disappear and reappear.

We consider a total distance of $d = 0.2\,\mathrm{m}$ with an equidistant volume length of $\Delta d = 0.01\,\mathrm{m}$. We set the time horizon to 1000 min and obtain simulation results illustrated in



(a) Calcite.

(b) Dolomite.

Figure 4: Molar concentrations of (a) calcite and (b) dolomite over the discretized distance at different times. Calcite first starts to dissolve in the brine feed, which enables dolomite formation. With further progress, dolomite starts to dissolve again in the brine solution. Quartz dissolution in the brine solution is negligible (not shown).

Fig. 4 showing the molar density of calcite and dolomite over the length of the porous rock. Due to the different presence of calcite and dolomite at different nodes and times, the composition of the brine solution changes as well with changing pH of the brine, as illustrated in Fig. 5. Initially, the solution is composed of water and in equilibrium with calcite. The brine feed imposes a pH reduction while dissolving calcite, lowering the pH to 4.5. The formation of dolomite results in a minor pH change, and only when both dolomite and calcite disappear the pH starts to further reduce to that of the brine feed.

This case study shows that the computation of reactive mass transfer is possible by employing an equation-oriented approach of discretizing the partial differential equation and integrating the semi-explicit DAE with a numerical integrator. We note, however, that the computation of the solution is expensive with integration times in the order of 1×10^3 s. The high computational cost stems from multiple nonlinear equation systems within the DAE that remain for the integrator to be solved in each time step after translation of the model. These nonlinear equation systems include both the equilibrium conditions and the volume constraints in each volume element. In comparison, the computational time to solve sequential approaches is in the order of 1×10^1 s [41]. Hence, in line with the literature on solving reactive transport models [15], we conclude that a sequential approach is more suitable for these types of problems.

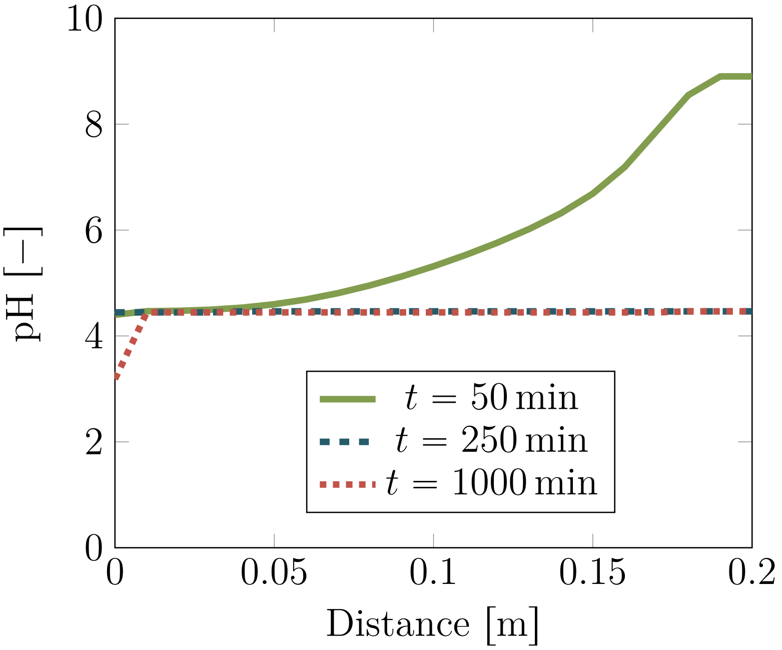


Figure 5: The change of present solid phases imposes a shift in pH over the distance, illustrated at different times.

5. Conclusion and outlook

In this work, we evaluated the equation-oriented approach to describe the transient behavior of a chemical system at equilibrium by embedding the equilibrium conditions as algebraic equations in the DAE formulation. We explain the implications of incorporating the complementarity conditions for the nonnegativity of chemical species as algebraic equations. The elimination of the nonnegativity conditions from the DAE assumes that these species are always present, and a smooth approximation of the remaining complementarity conditions results in a well-posed semi-explicit, index-1 DAE.

The object-oriented implementation of the DAE as multiple Modelica `Media` packages for the modeling and simulation of transient aqueous electrolyte systems enables its reusability for different applications and unit models. Additionally, the framework implements thermodynamic models in combination with the parameterization for numerous chemical species to compute the thermodynamic properties of gas, liquid, and solid phases relevant to aqueous electrolyte systems.

With the given framework, we performed exemplary case studies with different complexities to evaluate the applicability of the DAE formulation. The nonideality of thermodynamic models and the compositions of species changing by orders of magnitude can impose the Jacobian to be close to singularity, which makes the numerical integration costly. Integrating a model with many embedded equilibrium systems, e.g., due to discretized partial differential equations in reactive transport problems, still works fine. Yet, sequential approaches generally allow faster convergence. We found the equation-oriented

DAE formulation suitable for systems that incorporate a limited number of embedded equilibrium systems.

However, sequential approaches do not necessarily compute the transient behavior of the aqueous electrolyte system correctly, as the iterative solution of different sets of equations may result in a propagation of errors. Only in the case of sufficiently small time steps, the error becomes negligible, which is not always apparent from simulation results. Hence, for small-scale aqueous electrolyte systems, the advantage of employing the equation-oriented formulation used in this work lies in the exact solution of the DAE describing the transient behavior of an aqueous electrolyte system. This approach also enables the efficient modeling of unit models that incorporate further constitutive equations, e.g., volume constraints and energy balances. Additionally, the equation-oriented formulation allows the incorporation of control strategies and dynamic optimization of these systems. Future extensions of the framework may include additional thermodynamic models to account for solid mixtures, the incorporation of different databases, and the implementation of a medium containing gas, liquid, and solid phases.

Acknowledgments We thank the Federal Ministry of Education and Research (BMBF) for funding of the project CO2Min (033RO14B).

Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. M. Sahlodin, H. A. J. Watson, P. I. Barton, Nonsmooth model for dynamic simulation of phase changes, AIChE Journal 62 (2016) 3334–3351. doi:`10.1002/aic.15378`.

[2] T. Ploch, M. Glass, A. M. Bremen, R. Hannemann-Tamás, A. Mitsos, Modeling of dynamic systems with a variable number of phases in liquid-liquid equilibria, AIChE Journal 65 (2019) 571–581. doi:`10.1002/aic.16447`.

[3] L. T. Biegler, Nonlinear programming: Concepts, algorithms, and applications to chemical processes, Society for Industrial and Applied Mathematics (SIAM 3600 Market Street Floor 6 Philadelphia PA 19104), Philadelphia, Pa., 2010. URL: `http://epubs.siam.org/ebooks/siam/mos-siam_series_on_optimization/mo10`.

[4] C. I. Steefel, K. T. B. MacQuarrie, Chapter 2. Approaches to modeling of reactive transport in porous media, in: P. C. Lichtner, C. I. Steefel, E. H. Oelkers (Eds.), Reactive Transport in Porous Media, Reviews in Mineralogy & Geochemistry, De Gruyter, Berlin, 1996, pp. 83–130. doi:`10.1515/9781501509797-005`.

[5] A. I. Kakhu, C. C. Pantelides, Dynamic modelling of aqueous electrolyte systems, Computers & Chemical Engineering 27 (2003) 869–882. doi:`10.1016/S0098-1354(03)00002-4`.

[6] W. R. Smith, R. W. Missen, Chemical reaction equilibrium analysis: Theory and algorithms, A Wiley-Interscience Publication, Wiley, New York, 1982.

[7] A. H. Truesdell, B. F. Jones, WATEQ, a computer program for calculating chemical equilibria of natural waters, Jour. Research U.S. Geol. Survey 2 (1974). URL: `https://pubs.usgs.gov/journal/1974/vol2issue2/report.pdf#page=105`.

[8] T. J. Wolery, EQ3/6, a software package for geochemical modeling of aqueous systems: Package overview and installation guide (Version 7.0), 1992. URL: `https://inis.iaea.org/search/search.aspx?orig_q=rn:24046895`.

[9] D. L. Parkhurst, C. A. J. Appelo, Description of input and examples for PHREEQC version 3-A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations, volume 6 of *Techniques and Methods*, U.S. Geological Survey, 2013. URL: `http://pubs.usgs.gov/tm/06/a43`.

[10] A. M. Leal, D. A. Kulik, G. Kosakowski, M. O. Saar, Computational methods for reactive transport modeling: An extended law of mass-action, xLMA, method for multiphase equilibrium calculations, Advances in Water Resources 96 (2016) 405–422. doi:`10.1016/j.advwatres.2016.08.008`.

[11] D. A. Kulik, T. Wagner, S. V. Dmytrieva, G. Kosakowski, F. F. Hingerl, K. V. Chudnenko, U. R. Berner, GEM-Selektor geochemical modeling package: revised algorithm and GEMS3K numerical kernel for coupled simulation codes, Computational Geosciences (2012). doi:10.1007/s10596-012-9310-6.

[12] T. Wagner, D. A. Kulik, F. F. Hingerl, S. V. Dmytrieva, GEM-Selektor geochemical modeling package: TSolMod library and data interface for multicomponent phase models, The Canadian Mineralogist 50 (2012) 1173–1195. doi:10.3749/canmin.50.5.1173.

[13] A. M. M. Leal, Reaktoro: An open-source unified framework for modeling chemically reactive systems, 2015. URL: https://reaktoro.org.

[14] A. M. Leal, D. A. Kulik, G. Kosakowski, Computational methods for reactive transport modeling: A Gibbs energy minimization approach for multiphase equilibrium calculations, Advances in Water Resources 88 (2016) 231–240. doi:10.1016/j.advwatres.2015.11.021.

[15] G. T. Yeh, V. S. Tripathi, A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components, Water Resources Research 25 (1989) 93–108. doi:10.1029/WR025i001p00093.

[16] Modelica Association, Modelica, 2021. URL: https://www.modelica.org/.

[17] T. Ploch, E. v. Lieres, W. Wiechert, A. Mitsos, R. Hannemann-Tamás, Simulation of differential-algebraic equation systems with optimization criteria embedded in Modelica, Computers & Chemical Engineering 140 (2020) 106920. doi:10.1016/j.compchemeng.2020.106920.

[18] J. Nocedal, S. J. Wright (Eds.), Numerical Optimization, Springer-Verlag, New York, 1999. URL: http://dx.doi.org/10.1007/b98874. doi:10.1007/b98874.

[19] H. I. Moe, S. Hauan, K. M. Lien, T. Hertzberg, Dynamic model of a system with phase- and reaction equilibrium, Computers & Chemical Engineering 19 (1995) 513–518. doi:10.1016/0098-1354(95)87088-1.

[20] K. E. Brenan, S. L. Campbell, L. R. Petzold, Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, Society for Industrial and Applied Mathematics, 1995. doi:10.1137/1.9781611971224.

[21] O. Walz, C. Marks, J. Viell, A. Mitsos, Systematic approach for modeling reaction networks involving equilibrium and kinetically-limited reaction steps, Computers & Chemical Engineering 98 (2017) 143–153. doi:10.1016/j.compchemeng.2016.12.014.

[22] A. Kumar, P. D. Christofides, P. Daoutidis, Singular perturbation modeling of nonlinear processes with nonexplicit time-scale multiplicity, Chemical Engineering Science 53 (1998) 1491–1504. doi:10.1016/S0009-2509(98)00006-2.

[23] N. Vora, P. Daoutidis, Nonlinear model reduction of chemical reaction systems, AIChE Journal 47 (2001) 2320–2332. doi:10.1002/aic.690471016.

[24] P. Daoutidis, DAEs in Model Reduction of Chemical Processes: An Overview, in: A. Ilchmann, T. Reis (Eds.), Surveys in differential-algebraic equations, Differential-algebraic equations forum, DAE-F, Springer, Cham, 2015, pp. 69–102. doi:10.1007/978-3-319-11050-9{\textunderscore}2.

[25] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, Y. Zhang, On the formulation and theory of the Newton interior-point method for nonlinear programming, Journal of Optimization Theory and Applications 89 (1996) 507–541. doi:10.1007/BF02275347.

[26] A. Fischer, A special newton-type optimization method, Optimization 24 (1992) 269–284. doi:10.1080/02331939208843795.

[27] C. Kanzow, Some Noninterior Continuation Methods for Linear Complementarity Problems, SIAM Journal on Matrix Analysis and Applications 17 (1996) 851–868. doi:10.1137/S0895479894273134.

[28] A. Caspari, L. Lüken, P. Schäfer, Y. Vaupel, A. Mhamdi, L. T. Biegler, A. Mitsos, Dynamic optimization with complementarity constraints: Smoothing for direct shooting, Computers & Chemical Engineering 139 (2020) 106891. doi:10.1016/j.compchemeng.2020.106891.

[29] A. Zinser, L. Rihko-Struckmann, K. Sundmacher, Dynamic method for computation of chemical and phase equilibria, Computers & Chemical Engineering 89 (2016) 1–10. doi:10.1016/j.compchemeng.2016.02.014.

[30] A. M. Bremen, T. Ploch, A. Mhamdi, A. Mitsos, A mechanistic model of direct forsterite carbonation, Chemical Engineering Journal 404 (2021) 126480. doi:10.1016/j.cej.2020.126480.

[31] Modelica Association, Modelica Open-Source Libraries: ElectrolyteMedia, 2022. URL: https://modelica.org/libraries.html.

[32] D.-Y. Peng, D. B. Robinson, A New Two-Constant Equation of State, Industrial & Engineering Chemistry Fundamentals 15 (1976) 59–64. doi:10.1021/i160057a011.

[33] W. Wagner, H.-J. Kretzschmar, IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam, in: W. Wagner, H.-J. Kretzschmar (Eds.), International

Steam Tables, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 7–150. doi:10.1007/978-3-540-74234-0{\textunderscore}3.

[34] J. C. Tanger, H. C. Helgeson, Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures; revised equations of state for the standard partial molal properties of ions and electrolytes, American Journal of Science 288 (1988) 19–98. doi:10.2475/ajs.288.1.19.

[35] L. A. Bromley, Thermodynamic properties of strong electrolytes in aqueous solutions, AIChE Journal 19 (1973) 313–320. doi:10.1002/aic.690190216.

[36] K. S. Pitzer (Ed.), Activity coefficients in electrolyte solutions, 2. ed. ed., CRC Press, Boca Raton, 1991.

[37] T. J. B. Holland, R. Powell, An improved and extended internally consistent thermodynamic dataset for phases of petrological interest, involving a new equation of state for solids, Journal of Metamorphic Geology 29 (2011) 333–383. doi:10.1111/j.1525-1314.2010.00923.x.

[38] K. Zimmer, Y. Zhang, P. Lu, Y. Chen, G. Zhang, M. Dalkilic, C. Zhu, SUPCRTBL: A revised and extended thermodynamic dataset and software package of SUPCRT92, Computers & Geosciences 90 (2016) 97–111. doi:10.1016/j.cageo.2016.02.013.

[39] Dassault Systèmes AB, Dymola: Dynamic Modeling Laboratory, 2020. URL: https://www.3ds.com/products-services/catia/products/dymola/.

[40] A. C. Hindmarsh, ODEPACK, a systematized collection of ODE solvers, Scientific Computing (1983) 55–64. URL: https://ci.nii.ac.jp/naid/10008600225/.

[41] A. M. M. Leal, D. A. Kulik, M. O. Saar, Ultra-Fast Reactive Transport Simulations When Chemical Reactions Meet Machine Learning: Chemical Equilibrium, 2017. doi:10.48550/arXiv.1708.04825.