Calculating chemical speciation, pH, saturation index and calcium carbonate precipitation potential (CCPP) based on alkalinity and acidity using OpenModelica
Valerie Mehl and Klaus Johannsen

ABSTRACT
Of the most commonly used calcium carbonate (CaCO₃) saturation indices, the Calcium Carbonate Precipitation Potential (CCPP) most accurately describes CaCO₃ precipitation and dissolution. This paper illustrates how to write an OpenModelica programme that calculates the CCPP and other key water parameters starting from total alkalinity and CO₂-acidity. It describes the calculation steps of all components of the developed OpenModelica package and explains how the components work together. Underlying chemical principles follow the German standard DIN 38404-10 and are presented along with the OpenModelica code. The developed programme considers 14 different dissolved species, including complex species of carbonate, calcium, magnesium and sulphate. Data of the six phosphate-free example waters from DIN 38404-10 is compared to the results of the developed programme and the results of a programme which neglects complexation. The developed programme reaches required accuracy for four of the six CCPP values; with the other programme, requirements cannot be met. By using OpenModelica, the developed package can be readily integrated into a Modelica model of, for example, a drinking water treatment plant. There, it can help to monitor legal and technical requirements, or to assess the dissolution capacity if neutralizing filters containing CaCO₃ are employed for pH adjustment.

Key words | calcite dissolution capacity, calcium carbonate precipitation capacity (CCPC), ion pairs, modelling, software

INTRODUCTION
Calcium carbonate precipitation potential
The Calcium Carbonate Precipitation Potential (CCPP) is the amount of calcium carbonate (CaCO₃) that needs to precipitate or dissolve from water in order to reach equilibrium with CaCO₃. In equilibrium, water is saturated with CaCO₃ and the CCPP equals zero. If water is oversaturated with CaCO₃, the CCPP is positive; if it is undersaturated, the CCPP is negative (APHA/AWWA/WEF 2018).

Unlike other commonly used calcium carbonate saturation indices (such as the Saturation Index, the Relative Saturation and the Ryznar Index), the CCPP indicates both the tendency of a water to precipitate or dissolve CaCO₃ and the quantity that may be precipitated or dissolved. In drinking water treatment, knowledge about the ability of water to deposit CaCO₃ is often used in connection with the control of corrosion and scale formation in water conduits (Rossum & Merrill 1983; APHA/AWWA/WEF 2005).

In the current German Drinking Water Regulation (TrinkwV 2001), the parameter used to monitor calcium carbonate saturation is the Calcite Dissolution Capacity (Calcitlösekapazität) D_c. D_c has the same absolute value as the CCPP, but opposite sign; hence positive D_c values
represent CaCO$_3$-undersaturated water, while negative Dc values stand for CaCO$_3$-oversaturated water. According to the TrinkwV, Dc must not exceed 5 mg/L at the outlet of a water works (for a description of the rationale behind this limit value see Nissing & Johannsen (2003)). Even though no minimum limit value is defined, practice shows that, preferably, water close to saturation should be distributed, as extensive precipitation of CaCO$_3$ may cause disruptions in the distribution network and equipment (Nissing & Johannsen 2003).

The TrinkwV further specifies that Dc is to be determined following the calculation method set out in the German standard DIN 38404-10 if the pH at the outlet is below 7.7 (water with a pH $\leq$ 7.7 is considered to meet the limit value for Dc). For this purpose, DIN 38404-10 establishes chemical principles such as the reactions and thermodynamic constants that should be at the basis of any calculation. The newest version of DIN 38404-10 (DIN 38404-10:2012-22, hereafter referred to as DIN 38404-10) considers ion pairs and complex formation, including complexes with sulphate or phosphate (de Moel et al. 2013). The calculation of Dc starts out from data of the water chemistry analysis. To determine the reliability of this data, DIN 38404-10 lays down a procedure referred to as a plausibility test. To validate computer programmes, i.e. check whether their calculation of Dc is sufficiently accurate, DIN 38404-10 provides data sets of ten exemplary water samples (Stock & Johannsen 2013).

The developed programme

In the course of the present work, a computer programme which calculates the chemical speciation, pH, saturation index and eventually the Ccpp based on the chemical principles established in DIN 38404-10 has been developed. The programme has been created in the form of an OpenModelica package. OpenModelica is an open-source modelling and simulation environment developed by the Open Source Modelica Consortium (Open Source Modelica Consortium 2017; OpenModelica can be downloaded free of charge from openmodelica.org). It is based on the object-oriented, equation-based programming language Modelica, devised by the Modelica Association (Modelica Association 2017). Modelica is especially useful to model and connect processes of different natural and technical domains. Moreover, in Modelica, reuse of components and evolution of models is particularly convenient (Fritzson 2015). The developed package has been conceived to become part of an OpenModelica model which combines the chemical and physical processes inside a drinking water treatment plant. Therefore, OpenModelica is considered a suitable programming environment for this project.

The programme considers all aqueous species specified in DIN 38404-10 except from Ca(OH)$^+$, Mg(OH)$^+$, HSO$_4^-$ and the ones containing phosphate. Like the standard, it applies to ‘waters, that are designated for the distribution as drinking water’ (DIN 38404-10, p. 6). Due to its intended application, the input parameters of the developed programme differ slightly from the ones in DIN 38404-10 (further details in the following section).

This paper serves readers as a guideline to develop their own programme. It follows in large part the approach presented in Eberle & Donnert (1991), but focusses on the implementation with OpenModelica. For this purpose, the six components of the developed OpenModelica package will be explained in detail, while concentrating on how the main steps of calculating the Ccpp have been realised and how the different components work together. Where necessary, underlying chemical principles are elucidated. The thoroughly commented OpenModelica code of the entire package (as for example water 1) can be found in the appendix (available with the online version of this paper). From there, the code can be directly copied into the OpenModelica Connection Editor and run straightaway. In this project, OpenModelica version 1.9.6 was used.

To assess the accuracy of the developed programme, calculation results of six example waters from DIN 38404-10 will be compared to the values given in the standard and to those obtained with a programme based on a simplified model that does not consider ion pairs and complex formation.

METHODS

Input parameters of the developed programme and differences to DIN 38404-10

The input parameters of the developed programme are listed in Table 1. As mentioned earlier, the programme is
meant to be applied within some other larger model, for example of a drinking water treatment plant. For this reason, the developed programme starts out from the parameters ionic strength, total alkalinity and CO₂-acidity and not, like DIN 38404-10, from the pH, acid neutralising capacity with reference to pH 4.3, and base or acid neutralising capacity with reference to pH 8.2. Consequently, precision of the underlying data is also assumed to have been previously assessed. This particularly entails that the plausibility test set out in DIN 38404-10 is not within the scope of this programme. The programme also calculates the ionic strength (IOScal), but only to enable comparison with the input ionic strength (IOS). For calculation of the CCPP, IOS is used. The total concentrations of sodium, potassium, chloride or nitrate, respectively found to be in the range of 0.001 mg/L, thus far beneath the accuracy limit of 0.1 mg/L demanded in DIN 38404-10. Therefore, omission of Ca(OH)₂, Mg(OH)₂ and HSO₄ does not entail any significant losses in precision, but helps to considerably decrease code complexity and increase its readability. The neglect of phosphate is justified as its concentrations inside drinking water treatment plants are usually very small, especially compared to phosphate dosages that might be added later to inhibit corrosion. However, the developed programme can be extended to include the omitted species and reactions without any difficulties.

### Determining concentration-based equilibrium constants (function Constants)

The function Constants calculates the concentration-based equilibrium constants Kc in mmol/L at the temperature T. As input parameters, it requires the temperature T in °C and the ionic strength IOS in mmol/L. Output parameters are the array Kc containing the eleven concentration-based equilibrium constants and the activity coefficient fH of H⁺, which will be needed in the function EqConc to calculate the pH.

The function begins by calculating the decimal logarithm of the activity coefficients f of the considered species using the Debye–Hückel equation (DIN 38404-10):

$$\ln(f(i)) = -\frac{A \cdot z(i)^2 \sqrt{\text{IOS}/1000}}{1 + B \cdot g(i) \cdot \sqrt{\text{IOS}/1000}}$$  \hspace{1cm} (1)
with

\[ A = \frac{1824300}{(TK \cdot DK)^{1.5}} \]  

(2)

\[ B = \frac{50.248}{(TK \cdot DK)^{0.5}} \]  

(3)

\[ DK = 87.74 - 0.40008 \cdot T + 0.0009598 \cdot T^2 \]
\[ - 0.0000141 \cdot T^3 \]  

(4)

where IOS is the ionic strength of the water in mmol/L; \( z(i) \) is the charge of species \( i \); \( g(i) \) is the ion size parameter of species \( i \); \( DK \) is the dielectric constant of the water; \( TK \) is the temperature in K \( (TK = T + 273.15) \), and \( T \) is the temperature in °C.

All considered species, their index \( i \) and their ion size parameter can be found on the right side of Table 3. For uncharged species, \( \lg(f(i)) \) is zero. The explicit activity coefficient is only calculated for \( H^+ \), as it will be needed in the function EqConc.

The equilibrium constants \( K \) considered in the programme are listed in Table 2. They are defined based on the activities of the species involved in the respective reaction. However, for subsequent calculations it is convenient to use concentration-based constants \( Kc \). The conversion procedure will be demonstrated exemplary for \( [1] \). In the following, brackets \([\]\) signalize concentrations (besides identifying indices within arrays), while braces \( \{ \} \) symbolize activities.

\[ Kc[1] = \frac{[H^+][HCO_3^-]}{[CO_2]} \]  

(5)

The left part of the right side of Equation (7) equals the activity-based constant \( K[1] \) (cf. Table 2). Besides, \( CO_2 \) has no charge, therefore \( f(CO_2) = 1 \) and Equation (7) becomes:

\[ Kc[1] = \frac{K[1]}{f(H^+)} \cdot f(HCO_3^-) \]  

(8)

which is equal to:

\[ Kc[1] = 10^{\lg(K[1]) - \lg(f(H^+)) - \lg(f(HCO_3^-))} \]  

(9)

### Table 2

<table>
<thead>
<tr>
<th>Index</th>
<th>Activity-based equilibrium constant</th>
<th>( \lg(Kc) )</th>
<th>( \Delta H )</th>
<th>( C_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>([H^+][HCO_3^-]/[CO_2] )</td>
<td>-6.356</td>
<td>9,250</td>
<td>-330</td>
</tr>
<tr>
<td>2</td>
<td>([H^+][CO_2^2-]/[CO_3^2-] )</td>
<td>-10.329</td>
<td>14,950</td>
<td>-290</td>
</tr>
<tr>
<td>3</td>
<td>([Ca^{2+}][HCO_3^-]/[CaCO_3] )</td>
<td>-1.121</td>
<td>-7,950</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>([Ca^{2+}][CO_2^2-]/[CaCO_3] )</td>
<td>-3.200</td>
<td>-16,000</td>
<td>-430</td>
</tr>
<tr>
<td>5</td>
<td>([Ca^{2+}][SO_4^2-]/[CaSO_4] )</td>
<td>-2.310</td>
<td>-7,600</td>
<td>-170</td>
</tr>
<tr>
<td>6</td>
<td>([Mg^{2+}][HCO_3^-]/[MgHCO_3] )</td>
<td>-1.068</td>
<td>-7,250</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>([Mg^{2+}][CO_2^2-]/[MgCO_3] )</td>
<td>-2.947</td>
<td>-13,000</td>
<td>-70</td>
</tr>
<tr>
<td>8</td>
<td>([Mg^{2+}][SO_4^2-]/[MgSO_4] )</td>
<td>-2.263</td>
<td>-18,000</td>
<td>-10</td>
</tr>
<tr>
<td>9</td>
<td>([H^+][OH^-] )</td>
<td>-13.996</td>
<td>56,532</td>
<td>-197</td>
</tr>
<tr>
<td>10</td>
<td>([Ca^{2+}][CO_2^2-] )</td>
<td>-8.481</td>
<td>-10,000</td>
<td>-250</td>
</tr>
<tr>
<td>11</td>
<td>([H^+][H^+][CO_2^2-]/p(CO_2) )</td>
<td>-18.156</td>
<td>4,150</td>
<td>-460</td>
</tr>
</tbody>
</table>
The function begins by calculating the concentrations of $H^+$ and $OH^-$ from the handed over $pH$, activity coefficient $fH$ and equilibrium constant $Kc[9]$. Then the concentrations of the main constituents (constituents are the species that do not further dissolve under the examined conditions (Eberle & Donnert 1991)) $CO_3^{2-}, Ca^{2+}, Mg^{2+}$ and $SO_4^{2-}$ are calculated from the given total concentrations. The calculation will be presented taking the example of the carbonate ion $CO_3^{2-}$.

The total carbon concentration $tC$ is the sum of the concentrations of all species containing carbon and equals the sum of total alkalinity and $CO_2$-acidity (cf. Table 1):

\[
tC = [CO_3^{2-}] + [HCO_3^+] + [CO_2] + [CaCO_3] + [CaHCO_3] + [MgCO_3] + [MgHCO_3] + [MgSO_4]
\]

Using the concentration-based constants $Kc$, Equation (13) can also be expressed as follows:

\[
\]

\[
= [CO_3^{2-}] \cdot \left(1 + \frac{[H^+]}{Kc[2]} + \frac{[H^+]^2}{Kc[1]Kc[2]} + \frac{[Ca^{2+}]}{Kc[4]} + \frac{[Mg^{2+}][H^+]}{Kc[7]} + \frac{[Mg^{2+}][H^+]^2}{Kc[2]Kc[6]}ight)
\]

\[
= [CO_3^{2-}] \cdot S
\]
As $tC$ is given by Equation (14), $[CO_3^{2-}]$ can be calculated from $tC$ by:

$$[CO_3^{2-}] = \frac{tC}{S}$$  \hfill (18)

Similar relations can be derived for $Ca^{2+}$, $Mg^{2+}$ and $SO_4^{2-}$.

Since the constituent concentrations depend on one another, they must be calculated iteratively. In the programme, this is realised by a while-loop. As starting values for the sought concentrations, the respective total concentrations and the equilibrium constants as described before. The loop will be exited if:

$$\left| \frac{\sum \text{old values} - \sum \text{new values}}{\sum \text{new values}} \right| < 10^{-6}$$ \hfill (19)

If this is not the case, a new iteration step begins with the new values as starting values. Once $[CO_3^{2-}]$, $[Ca^{2+}]$, $[Mg^{2+}]$ and $[SO_4^{2-}]$ have been calculated with a sufficient precision, they are used together with the equilibrium concentrations of the remaining species.

As a last step, FC, $D$ and $IOScal$ are computed. FC and $D$ are necessary to determine the pH in the function $pH\_Alk\_Aci$; $IOScal$ is just calculated to enable comparison with the measured IOS. According to Eberle (p89), the relation between Alk and $tC$ can be expressed by the linear equation:

$$Alk = tC \cdot FC + D$$ \hfill (20)

with

$$FC = \frac{2 \cdot ([CO_3^{2-}] + [CaCO_3] + [MgCO_3])}{[FC]} + \frac{[HCO_3^{-}] + [CaHCO_3] + [MgHCO_3]}{[FC]}$$ \hfill (21)

$$D = [OH^-] - [H^+]$$ \hfill (22)

$IOScal$ is computed via:

$$IOS = 0.5 \cdot \sum c(i) \cdot z(i)^2$$ \hfill (23)

where $c(i)$ is the concentration and $z(i)$ the charge of the species $i$ (DIN 38404-10).

Determining pH and SI (functions $pH\_Alk\_Aci$ and $FunctSI$)

The function $pH\_Alk\_Aci$ primarily determines the equilibrium pH from the input water data. A flowchart illustrating the sequence of necessary operations is given in Figure 1. As input variables, $pH\_Alk\_Aci$ requires $T$, $IOS$, $Alk$, $Aci$, $tCa$, $tMg$, $tSO_4$, $tNa$, $tK$, $tCl$ and $tNO_3$. The output variables are the equilibrium pH ($pHcal$), $tC$, the equilibrium concentrations $ec$ of all considered species, $IOScal$ and $SI$. For the determination of the CCPP, the actual role of the function $pH\_Alk\_Aci$ is to provide the saturation index $SI$. As described later, the other output variables do not necessarily have to be considered in the main programme; however, they are useful to determine the programme’s accuracy and estimate the reliability of the resulting CCPP.

The function $pH\_Alk\_Aci$ calls the two previously described functions ($Constants$ and $EqConc$). As stated before, the function $EqConc$ determines the equilibrium concentrations at any given pH. However, in the function $pH\_Alk\_Aci$, the output of $EqConc$ is actually used to determine the equilibrium pH. This is carried out iteratively by adapting the input pH of $EqConc$ until equilibrium state is reached with sufficient precision. Before entering the pH iteration loop, the function $Constants$ needs to be called to provide the equilibrium constants $Kc$ for $EqConc$.

For the very first iteration step, the iteration pH ($pHit$) is assumed to be 7. Then the function $EqConc$ is called to determine $ec$, $FC$, $D$ and $IOScal$. The returned values of $FC$ and $D$ are immediately used to evaluate Equation (20). If, for the current $pHit$, the auxiliary variable zero is above 0, the current value of $tC \cdot FC - D$ is lower than $Alk$. In particular, the calculated $FC$ and $D$ are too low. Because
they both increase with increasing pH (and decrease with decreasing pH), the iteration pH of the next iteration step must be raised, which is realised by setting the lower boundary pH ($pH_{\text{min}}$) to $pH_{\text{Hit}}$. If, on the other hand, zero is beneath 0, the higher boundary pH ($pH_{\text{max}}$) is altered instead. This procedure is also known as the bisection method. For the (unlikely but possible) case that zero is exactly 0, $pH_{\text{Hit}}$ is the sought pH and both boundary pH values can be set to $pH_{\text{Hit}}$. After each iteration step, the absolute difference between upper and lower boundary pH is evaluated. If it reaches the desired precision (here $10^{-6}$), the while-loop is exited and the sought equilibrium pH ($pH_{\text{cal}}$) is set to $pH_{\text{Hit}}$ of the last iteration step. Note that, to enhance robustness, iteration has been limited to a pH range between 0 and 14. As this programme is developed for applications in drinking water treatment, this should not cause any problems.

Subsequently, the saturation index $SI$ is determined by calling the function $\text{FunctSI}$, which calculates $SI$ according to DIN 38404-10 via:

$$SI = \lg \left( \frac{[Ca^{2+}][CO_3^{2-}]}{K_c} \right)$$

An if-statement is added to the function to prevent misleading values of $SI$ in case concentrations of $Ca^{2+}$ or $CO_3^{2-}$ are zero.

**Determining the CCPP (function $CCPP_{\text{Alk_Aci}}$)**

The function $CCPP_{\text{Alk_Aci}}$ finally determines the CCPP of the examined water. For this purpose, it needs the same input variables as the function $pH_{\text{Alk_Aci}}$ ($T$, $IOS$, $Alk$, $Aci$, $tCa$, $tMg$, $tSO_4$, $tNa$, $tK$, $tCl$ and $tNO_3$). Aside from the CCPP, it also outputs $pH_{\text{C}}$, the pH of saturation with CaCO$_3$. Figure 2 shows the flowchart for $CCPP_{\text{Alk_Aci}}$.

There is no way to directly calculate the CCPP. Therefore, CaCO$_3$ precipitation has to be simulated by iteratively withdrawing virtual amounts of CaCO$_3$ from the water until saturation is reached. This virtual withdrawal is realised by altering given water parameters to the way they would react to precipitation of CaCO$_3$. For precipitation of $X$ mmol/L of CaCO$_3$, alterations are as follows:

- $IOS$ decreases by $4 \cdot X$, as both $CO_3^{2-}$ and $Ca^{2+}$ contribute to the ionic strength with two times their concentration (cf. Equation (23)).
- $Alk$ decreases by $2 \cdot X$, because $CO_3^{2-}$ contributes to it with two times its concentration (cf. Table 1).
- $Aci$ increases by $X$, because $CO_3^{2-}$ counteracts CO$_2$- acidity with one time its concentration (cf. Table 1), and
- $tCa$ decreases by $X$, as 1 mol of CaCO$_3$ contains the same amount of calcium.

The actual implementation of $CCPP_{\text{Alk_Aci}}$ is similar to $pH_{\text{Alk_Aci}}$. Iteration is again realised through a while-loop, but this time with $CCPP_{\text{it}}$ as iteration variable. Inside the loop, the function $pH_{\text{Alk_Aci}}$ is called. Here it is crucial to alter the input parameters using $CCPP_{\text{it}}$ as
described above. The returned saturation index \( SI_C \) is then evaluated in order to determine whether saturation has been reached sufficiently. If \( SI_C \) is above 0, the altered water is still oversaturated and the current \( CCPPit \) is lower than the sought \( CCPP \). Hence, \( CCPPit \) is raised for the next iteration step by setting the lower boundary \( CCPP \) (\( CCPP_{min} \)) to \( CCPPit \). If, on the other hand, \( SI_C \) is beneath 0, the higher boundary \( CCPP \) is adapted. For the (unlikely but possible) case that \( SI_C = 0 \), the exact value of the \( CCPP \) is found. The loop is exited if the difference between upper and lower boundary \( CCPP \) reaches or underpasses \( 10^{-6} \) mmol/L. Eventually, the iteration \( CCPP \) of the last iteration step is taken as the actual \( CCPP \) (in mmol/L) of the examined water.

Note that, to enhance robustness, overall limit values for the \( CCPP \) have been set. The upper limit has been set to \(-200 \) mg/L, as higher values are rarely found in drinking water treatment; yet it remains an arbitrary number.

**Main model (model MAIN_CCPP)**

The main model has to provide the input parameters listed in Table 1 in the specified units. Call the function \( pH_{Alk\_Aci} \) in the main model to obtain the calculated \( pH \) (\( pH_{calc} \)), the sum of the concentrations of carbonic species (dissolved inorganic carbon, \( tC \)), the equilibrium concentrations (\( ec \)) of the examined water, the calculated ionic strength (\( IOScal \)) and the saturation index (\( SI \)). This is not required to calculate the \( CCPP \), but provides a means to determine the accuracy of the programme when testing it and to assess the reliability of its results when employing it. To obtain the \( CCPP \) (in mmol/L), the function \( CCPP_{Alk\_Aci} \) must be called. This function also outputs the \( pH \) of calcium carbonate saturation (\( pH_{C} \)). To obtain the \( CCPP \) in mg/L, multiply with the molar mass of CaCO\(_3\) (here 100.09 g/mol are used). A scheme that illustrates how all described components are nested within the programme is shown in Figure 3.

**RESULTS AND DISCUSSION**

**Comparison to DIN 38404-10 and a simplified model**

To assess the accuracy of the developed programme, the \( pH_{C} \) and the \( CCPP \) were calculated for six example data sets of DIN 38404-10, and compared to the values given in the standard. DIN 38404-10 explicitly provides these data sets for validation of computer programmes. Data of the data sets is synthetically generated and represents analysis data of ten example waters, which mainly differ in temperature and ion content. As mentioned before, the developed programme does not consider any phosphate species. Contrary to this, the values for \( Alk \) and \( Aci \) of phosphate-containing waters given in the standard and required for programme execution do include these species; therefore, only the waters which do not contain phosphate (waters 1, 2, 3, 4, 5, and 6) were compared.

**Table 1**

<table>
<thead>
<tr>
<th>Water</th>
<th>( SI_C )</th>
<th>( pH_{calc} )</th>
<th>( pH_{C} )</th>
<th>( CCPP ) (mmol/L)</th>
<th>( CCPP ) (mg/L)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water 1</td>
<td>0.2</td>
<td>7.5</td>
<td>10.0</td>
<td>1.2</td>
<td>1200</td>
</tr>
<tr>
<td>Water 2</td>
<td>0.3</td>
<td>7.6</td>
<td>10.2</td>
<td>1.3</td>
<td>1230</td>
</tr>
<tr>
<td>Water 3</td>
<td>0.4</td>
<td>7.7</td>
<td>10.4</td>
<td>1.4</td>
<td>1260</td>
</tr>
</tbody>
</table>

The results for the page are as follows:

- **Input parameters**
  - CCPPmin = 2 mmol/L, CCPPmax = iCa
  - \( CCPP_{min} - CCPP_{max} = 10^{-5} \)

- **CCPPit = (CCPPmin + CCPPmax) / 2**

- **If \( SI_C \) > 0**
  - CCPPmax = CCPPit
  - CCPPmin = CCPPit

- **If \( SI_C \) < 0**
  - CCPPmax = CCPPit

- **Output:**
  - \( pH_{C} \), CCPP (mmol/L)

**Figure 2** | Flowchart for function CCPP_{Alk\_Aci}.
(4, 7, 9) are included in this study. Table 4 summarizes relevant data of these waters. While the temperature remains constant between 10 and 15°C for all regarded waters, the ionic strength ranges widely, with a minimum at 1.041 mmol/L (water 4) and a maximum at 15.167 mmol/L (water 3). With 3.50 mmol/L and 0.70 mmol/L, water 3 contains the highest total concentrations of calcium and magnesium respectively. The highest total sulphate concentration is 1.40 mmol/L and occurs in water 7. The lowest concentrations of these three constituents are contained in water 4, with 0.15 mmol/L for calcium, 0.05 mmol/L for magnesium and 0.05 mmol/L for sulphate. Regarding their CCPP, waters 2 and 7 can be characterised as slightly, and waters 1 and 3 as moderately, CaCO₃ dissolving. Water 4 is highly dissolving, whereas water 9 is the only one with a positive CaCO₃ precipitation potential.

Calculation results were furthermore compared to those obtained with a programme based on a simplified model. The simplified model neglects ion pairs and complex formation. Therefore, it assumes that all calcium is present as Ca²⁺ (i.e. $\text{Ca}^2+$) and only considers the equilibrium constants of the first and second dissociation of CO₂, the ion product of water, and the solubility product of calcite, which are equivalent to $K[1]$, $K[2]$, $K[9]$ and $K[10]$ in Table 2. In the simplified model, they are calculated using the formulas provided by Ball & Nordstrom (1991). To calculate activity coefficients, the following simpler version of the Debye–Hückel equation is used:

$$\lg(f(i)) = \frac{-0.5 \cdot z(i)^2 \sqrt{\text{IOS}}/100}{1 + 1.4 \cdot \sqrt{\text{IOS}}/100}$$

and, when calculating SI, it is presumed that all alkalinity stems from HCO₃⁻.

Values and results of the three calculation methods are presented in Table 4. The comparison reveals that CCPP values calculated with the developed programme tend to be slightly too low; yet, four of the six calculated CCPP values meet the accuracy requirement of ±0.1 mg/L set out in DIN 38404-10. With neglect of ion pairs and complex formation, i.e. with the simplified programme, no CCPP value meets the requirement. Values of $p\text{Hcal}$ comply with the required accuracy of ±0.001 for five of the six waters if calculated using the developed programme; using the simplified programme, no $p\text{Hcal}$ value lies within limits of accuracy.

Results of the simplified programme are in accordance with findings described in APHA/AWWA/WEF (2005), which states that ‘[m]odels that do not consider [ion pairs] overestimate the amount of CaCO₃ that can be precipitated and underestimate the amount of CaCO₃ that can be dissolved’ (pp. 2–34). As can be seen in Table 4, this behaviour is especially pronounced for waters with high ion concentrations (waters 3 and 9). Correspondingly, calculations for waters with low ion concentrations (waters 2 and 4) give relatively good results.

Results of the developed programme deviate from DIN 38404-10 because, contrary to what is assumed in the function CCPP_Alk_Aci, the ionic strength does not precisely decrease by four times the precipitated amount of CaCO₃. When CaCO₃ precipitates, the water’s pH decreases, which affects speciation. Consequently, withdrawal of CaCO₃ not only results in lower concentrations of Ca²⁺ and CO₃²⁻, but also of other, lower charged species. Therefore, the ionic strength will generally decrease less than four times the precipitated amount of CaCO₃. To determine the exact decrease, it is necessary to additionally iterate over the ionic strength when calculating the CCPP in CCPP_Alk_Aci. However, within the additional iteration loop, IOScal has to serve as input parameter of pH_Alk_Aci. This is not desirable for this programme, as it shall also be applicable in situations in which not all constituent...
Table 4: Data of the six example waters and comparison of the pH of calcium carbonate saturation ($pH_C$) and the CCPP obtained with the developed programme (Progr.) to the values provided in DIN 38404-10 (DIN), and the results of a programme based on a simplified model (Simpl.)

<table>
<thead>
<tr>
<th>Input parameters</th>
<th>Unit</th>
<th>Water 1</th>
<th>Water 2</th>
<th>Water 3</th>
<th>Water 4</th>
<th>Water 7</th>
<th>Water 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>°C</td>
<td>10.0</td>
<td>15.0</td>
<td>10.0</td>
<td>10.0</td>
<td>12.0</td>
<td>15.0</td>
</tr>
<tr>
<td>ION</td>
<td>mmol/L</td>
<td>5.362</td>
<td>3.125</td>
<td>15.167</td>
<td>1.041</td>
<td>7.173</td>
<td>8.886</td>
</tr>
<tr>
<td>$AB$</td>
<td>mmol/L</td>
<td>2.445</td>
<td>1.577</td>
<td>5.392</td>
<td>0.168</td>
<td>1.092</td>
<td>4.194</td>
</tr>
<tr>
<td>$Ac$</td>
<td>mmol/L</td>
<td>0.292</td>
<td>0.054</td>
<td>1.333</td>
<td>1.207</td>
<td>0.067</td>
<td>0.478</td>
</tr>
<tr>
<td>$tCa$</td>
<td>mmol/L</td>
<td>1.40</td>
<td>0.75</td>
<td>3.50</td>
<td>0.15</td>
<td>1.30</td>
<td>2.65</td>
</tr>
<tr>
<td>$tMg$</td>
<td>mmol/L</td>
<td>0.23</td>
<td>0.10</td>
<td>0.70</td>
<td>0.05</td>
<td>0.25</td>
<td>0.20</td>
</tr>
<tr>
<td>$tSO_4$</td>
<td>mmol/L</td>
<td>0.38</td>
<td>0.15</td>
<td>1.20</td>
<td>0.05</td>
<td>1.40</td>
<td>0.55</td>
</tr>
<tr>
<td>$tNa$</td>
<td>mmol/L</td>
<td>0.30</td>
<td>0.40</td>
<td>2.30</td>
<td>0.30</td>
<td>1.60</td>
<td>0.30</td>
</tr>
<tr>
<td>$tK$</td>
<td>mmol/L</td>
<td>0.05</td>
<td>0.10</td>
<td>0.30</td>
<td>0.10</td>
<td>0.15</td>
<td>0.06</td>
</tr>
<tr>
<td>$tCl$</td>
<td>mmol/L</td>
<td>0.25</td>
<td>0.30</td>
<td>2.70</td>
<td>0.34</td>
<td>0.85</td>
<td>0.75</td>
</tr>
<tr>
<td>$tNO_3$</td>
<td>mmol/L</td>
<td>0.15</td>
<td>0.03</td>
<td>0.50</td>
<td>0.18</td>
<td>0.10</td>
<td>0.05</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Deviation from DIN</td>
<td></td>
<td>-0.001</td>
<td>-0.026</td>
<td>-0.021</td>
<td>-0.021</td>
<td>-0.001</td>
<td>-0.052</td>
<td>-0.014</td>
<td>-0.005</td>
<td>-0.001</td>
<td>-0.001</td>
<td>-0.001</td>
<td>-0.001</td>
</tr>
<tr>
<td>CCPP</td>
<td>mg/L</td>
<td>-14.5</td>
<td>-14.6</td>
<td>-12.9</td>
<td>-3.2</td>
<td>-3.2</td>
<td>-2.5</td>
<td>-16.3</td>
<td>-16.6</td>
<td>-6.5</td>
<td>-111.1</td>
<td>-111.6</td>
<td>-110.6</td>
</tr>
<tr>
<td>Deviation from DIN</td>
<td>mg/L</td>
<td>-0.1</td>
<td>-1.9</td>
<td>-0.0</td>
<td>0.7</td>
<td>-0.3</td>
<td>9.8</td>
<td>-0.5</td>
<td>0.5</td>
<td>0.0</td>
<td>+0.9</td>
<td>0.0</td>
<td>+0.9</td>
</tr>
</tbody>
</table>

*Input parameters that are not considered in the simplified model

- 0.026 Deviations that surpass the maximum deviation specified in DIN 38404-10.
concentrations listed in Table 1 are known, but IOS is assessable, for example through the electrical conductivity.

**CONCLUSIONS**

This work has shown how to programme calculation of species concentrations, pH, CaCO₃ saturation and the CCPP using OpenModelica. The advantage of using OpenModelica is that the developed package can easily be integrated into the model of, for example, a drinking water treatment plant, where it can be employed, for example, to estimate whether water leaving the plant meets the required CCPP, or to assess how much CaCO₃ dissolves if neutralizing filters containing calcium carbonate are employed to adjust the pH of the water.

The described approach adheres closely to the principles established in DIN 38404-10 and, as proven by comparison to six example waters, its results for the CCPP of phosphate-free waters nearly reach the degree of accuracy stipulated by the standard – a degree that cannot be met when neglecting ion pairs and complex formation.

**REFERENCES**


DIN 38404-10 2012 *Deutsche Einheitsverfahren zur Wasser-, Abwasser- und Schlammuntersuchung – Physikalische und physikalisch-chemische Stoffkenngrößen (Gruppe C) – Teil 10: Berechnung der Calcitsättigung eines Wassers (C 10)* (German Standard Methods for the Examination of Water, Waste Water and Sludge – Physical and Physico-Chemical Parameters (Group C) – Part 10: Calculation of the Calcite Saturation of Water (C 10)). Deutsches Institut für Normung e.V., Berlin, Germany.


First received 18 June 2017; accepted in revised form 8 November 2017. Available online 29 November 2017.