Cubic Plus Association

Ailo Aasen

December 5, 2016

1 Introduction to CPA

CPA is an equation of state which is suitable for modeling associating mixtures, i.e. where hydrogen bonds occur. CPA stands for Cubic Plus Association, which refers to the fact that association is modeled as an additive contribution to the Helmholtz energy of a cubic equation of state. If one uses SRK as the underlying cubic equation of state – which is the most common choice – then one in other words has

$$A^{CPA} = A^{ideal} + A^{SRK} + A^{assoc}. \tag{1}$$

1.1 The association contribution to Helmholtz energy

Each associating molecule i is assigned association sites A_i, B_i, ... For example, water is usually modeled as having four associating sites: each hydrogen atom and each 'free' electron valence pair in oxygen is a site. A site is called bonded if it is involved in a hydrogen bond. The association contribution for a mixture is modeled by

$$\frac{A^{assoc}}{RT} = \sum_i n_i \sum_{A_i} \left(\ln X_{A_i} - \frac{X_{A_i}}{2} + \frac{1}{2} \right) \tag{2}$$

where X_{A_i} is the fraction of molecules *not* bonded at site A_i, and is given by the nonlinear equation

$$X_{A_i} = \frac{1}{1 + (1/V) \sum_j n_j \sum_{B_j} X_{B_j} \Delta^{A_i B_j}}. \tag{3}$$

Here $\Delta^{A_i B_j}$ is called the bond association strength, and is in CPA given by

$$\Delta^{A_i B_j}(T, V, \mathbf{n}) = g(V, \mathbf{n}) \cdot [\exp(\epsilon^{A_i B_j}/RT) - 1] b_{ij} \beta^{A_i B_j}, \tag{4}$$

where the radial distribution function g is given by[1]

$$g(V, \mathbf{n}) = \frac{1}{1 - 0.475 B(\mathbf{n})/V}, \tag{5}$$

and $B(\mathbf{n}) = \sum_i n_i b_i$. The quantities b_i are the familiar co-volume parameters of the cubic EoS. The strength of the association is modeled using a square-well potential, and the quantities $\epsilon^{A_i B_j}$ and $\beta^{A_i B_j}$ represent the well depth and width, respectively.

1.2 Mixing rules

When the CPA EoS is used for mixtures, the conventional mixing rules are applied for cubic part of the equation, namely

$$a = \sum_i \sum_j x_i x_j a_{ij}, \qquad a_{ij} = \sqrt{a_i a_j}(1 - k_{ij}), \tag{6}$$

$$b = \sum_i x_i b_i, \tag{7}$$

[1] There are several variants of g; this one from simplified CPA (sCPA).

where $x_i = n_i/n$ is the molfraction of component i in the mixture. Note that even for mixtures containing associating components, the interaction parameters k_{ij} given here are the only adjustable parameters. Several alpha-formulation have been used, although the classic formulation is the most common choice. See also Section 8.1.

There are two recognized sets of mixing rules for the association parameters:

CR-1 mixing rules for association parameters

$$\epsilon^{A_iB_j} = \frac{\epsilon^{A_iB_i} + \epsilon^{A_jB_j}}{2}, \qquad \beta^{A_iB_j} = \sqrt{\beta^{A_iB_i}\beta^{A_jB_j}}. \tag{8}$$

Elliot's combining rule for association parameters

This combining rule gives directly an expression for $\Delta^{A_iB_j}$:

$$\Delta^{A_iB_j} = \sqrt{\Delta^{A_iB_i}\Delta^{A_jB_j}}. \tag{9}$$

Mixtures with solvation CPA can be applied to mixtures with one self-associating component and one non-self-asssociating compound, but where there is cross-association – solvation – between the two compounds. Examples are the acid-gas mixtures H2O–CO2 and H2O–H2S. Since CO2–H2O is an especially important binary for CCS applications, modeling solvation will be an important task for ThermoPack.

One approach to modeling cross-interaction between a self-associating component i and a non self-associating component j is to set $\Delta^{A_iB_j} = s_{ij}\Delta^{A_i}$, where s_{ij} is a constant determined by fitting the model to experimental data. In general, s_{ij} is temperature-dependent. With this method, no self-association is modeled for the non-self-associating component, thus preserving this qualitative physical aspect.

Another method is discussed by Kontogeorgis [6], who suggests the so-called modified CR-1 combining rule. The procedure is to use the same expression for $\Delta^{A_iB_j}$, but with

$$\epsilon^{A_iB_j} = \frac{\epsilon_{assoc}}{2}, \qquad \beta^{A_iB_j} \text{ (fitted)}. \tag{10}$$

Also for this method, a non-self-associating component has no self-association.

The third approach is to simply model the solvating component as an associating molecule. This approach sort of disregards the underlying physics, but has been found to give good results. Some experimentation may be needed to figure out the appropriate association scheme.

1.3 CPA compared to other SAFT equations

Although it's not necessary for the remainder, we briefly mention general SAFT equations. CPA is just one in a host of equations of state which model associating components, and that are classified as SAFT equations. SAFT stands for Statistical Associating Fluid Theory, and is – unlike multiparameter equations of state – a physically based framework for modeling associating mixtures. Although developed in the late 1980s and early 1990s, there is still much work being published on SAFT.

Fortunately, all of the SAFT variants use essentially the same expression for A^{assoc}, the difference lying in the bond associating strength $\Delta^{A_iB_j}$. As an example, we will give the $\Delta^{A_iB_j}$ as it appears in the model PC-SAFT, namely

$$\Delta^{A_iB_j} = g_{ij} \cdot [\exp(\epsilon^{A_iB_j}/kT) - 1](\sigma_{ij}^3\kappa^{A_iB_j}). \tag{11}$$

The radial distribution function of the hard-sphere fluid is

$$g_{ij} = \frac{1}{1-\zeta_3} + \left(\frac{d_id_j}{d_i+d_j}\right)\frac{3\zeta_2}{(1-\zeta_3)^2} + \left(\frac{d_id_j}{d_i+d_j}\right)^2\frac{2\zeta_2^2}{(1-\zeta_3)^2}. \tag{12}$$

where

$$\zeta_n = \frac{\pi}{6}\rho\sum_i x_im_id_i^n, \qquad n = 0,1,2,3, \tag{13}$$

and

$$d_i = \sigma_i\left[1 - 0.12\exp\left(-3\frac{\epsilon_i}{kT}\right)\right]. \tag{14}$$

Note, in particular, the in PC-SAFT, the radial distribution function is temperature dependent. Mixing rules for the PC-SAFT parameters:

$$\sigma_{ij} = \frac{1}{2}(\sigma_i + \sigma_j) \tag{15}$$

$$\epsilon_{ij} = \sqrt{\epsilon_i \epsilon_j}(1 - k_{ij}). \tag{16}$$

(Note that $\epsilon^{A_i B_j}$ and ϵ_{ij} are completely different parameters.)

The biggest strength of PC-SAFT compared to CPA, is that the non-association part is theoretically justified. This advantage becomes especially clear when modeling polymers, which PC-SAFT is tailored to be able to handle. For molecules having a chain-like structure, PC-SAFT indeed outperforms CPA. On the other hand, PC-SAFT has a more complicated non-association part, making it harder to implement (especially in ThermoPack, for which everything is built around cubic equations of state), and also more demanding in terms of computation time. Although only CPA will be implemented at this point, significant portions of the code can be recycled if one at one point in the future decides to implement other SAFT variants.

1.4 The one-strength association schemes

Huang and Radosz (1990) were the first to publish a SAFT variant with extensive parameter lists. They also classified association schemes, which has become the standard for later work on SAFT. Ideally, one should have detailed, independent data from spectroscopy for the associating strength for each site-site interaction. The scheme of Huang and Radosz reduces the number of parameters to be fitted for each interaction $A_i B_i$ to just one. To illustrate this one-strength association scheme, let us consider the water molecule, with two positively polarized hydrogen atoms (sites A and B), and two free electron pairs (sites C and D). Huang and Radosz classifies water according to the so-called 4C scheme:

$$\begin{aligned} &\Delta^{AA} = \Delta^{AB} = \Delta^{BB} = \Delta^{CC} = \Delta^{CD} = \Delta^{DD} = 0 \\ &\Delta^{AC} = \Delta^{AD} = \Delta^{BC} = \Delta^{BD} \neq 0. \end{aligned}$$

Although this is the most physically appropriate one-strength association scheme for water, various reasons (e.g. limited experimental data, or a desire for reduced complextiy), it is often modeled according to the 3B scheme:

$$\begin{aligned} &\Delta^{AA} = \Delta^{AB} = \Delta^{BB} = \Delta^{CC} = 0 \\ &\Delta^{AC} = \Delta^{BC} \neq 0. \end{aligned}$$

Of course, in the 3B scheme the physical interpretation of the association sites is not so clear, but in the end they are just fitted parameters.

In the end, the big advantage with the one-strength associating scheme, is that one just have to give the two numbers β^{AB} and ϵ^{AB}, in addition to the particular one-strength association scheme, to fully describe the association of an associating component. (Of course, the three SRK parameters come in addition to this.)

When modeling cross-association, there is only a contribution between the sites having the opposite polarity.

2 The Q function and its relation to F^{assoc}

Define the function

$$Q(\mathbf{n}, T, V, \mathbf{X}) = \sum_i \sum_{A_i} n_i \left(\ln X_{A_i} - X_{A_i} + 1\right) - \frac{1}{2V} \sum_{i,j} \sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j} \Delta^{A_i B_j}. \tag{17}$$

Here $\Delta^{A_i B_j} = \Delta^{A_i B_j}(T, V, \mathbf{n})$ is the bond association strength. If $\mathbf{X}$ solves the equations

$$\left(\frac{\partial Q}{\partial \mathbf{X}}\right)(T, V, \mathbf{n}, X) = \mathbf{0}, \quad \text{i.e.} \quad \frac{1}{X_{A_i}} - 1 - \frac{1}{V} \sum_j \sum_{B_j} n_j X_{B_j} \Delta^{A_i B_j} = 0 \quad \forall \ X_{A_i}. \tag{18}$$

then the resulting solution $\mathbf{X} = \mathbf{X}(T, V, \mathbf{n})$ is such that[2]

$$F^{\text{assoc}}(T, V, \mathbf{n}) = Q(T, V, \mathbf{n}, \mathbf{X}(T, V, \mathbf{n})). \tag{19}$$

We now clarify the notation used below in the expressions for the derivatives. Given a differential operator ∂, we will in the following use ∂Q_{sp} to mean $(\partial Q)(T, V, \mathbf{n}, \mathbf{X}(T, V, \mathbf{n}))$. For example, $\left(\frac{\partial Q_{sp}}{\partial V}\right) = \left(\frac{\partial Q}{\partial V}\right)(T, V, \mathbf{n}, \mathbf{X}(T, V, \mathbf{n}))$. Moreover, to avoid subscripting every partial derivative to show which variables are fixed, we agree once and for all that F^{assoc} has $(T, V, \mathbf{n})$ as independent variables, while Q has $(T, V, \mathbf{n}, \mathbf{X})$ as independent variables. The equality (19) can also be stated as $F^{\text{assoc}} = Q|_{\mathbf{X}=\mathbf{X}(T,V,\mathbf{n})}$.

3 First-order derivatives of F^{assoc}

We find that

$$\begin{aligned} \left(\frac{\partial F^{\text{assoc}}}{\partial V}\right) &= \left(\frac{\partial Q_{sp}}{\partial V}\right) + \sum_i \sum_{A_i} \left(\frac{\partial Q_{sp}}{\partial X_{A_i}}\right)\left(\frac{\partial X_{A_i}}{\partial V}\right) \\ &= \left(\frac{\partial Q_{sp}}{\partial V}\right), \end{aligned}$$

since $\left(\frac{\partial Q_{sp}}{\partial X_{A_i}}\right) = 0$. Similarly, we have

$$\left(\frac{\partial F^{\text{assoc}}}{\partial T}\right) = \left(\frac{\partial Q_{sp}}{\partial T}\right) \quad \text{and} \quad \left(\frac{\partial F^{\text{assoc}}}{\partial n_k}\right) = \left(\frac{\partial Q_{sp}}{\partial n_k}\right).$$

3.1 Volume derivative

$$\begin{aligned} \left(\frac{\partial F^{\text{assoc}}}{\partial V}\right) &= \left(\frac{\partial Q_{sp}}{\partial V}\right) \\ &= \frac{1}{2V} \sum_{i,j} \sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j} \left[\frac{\Delta^{A_iB_j}}{V} - \left(\frac{\partial \Delta^{A_iB_j}}{\partial V}\right)\right]. \end{aligned} \tag{20}$$

3.2 Temperature derivative

$$\begin{aligned} \left(\frac{\partial F^{\text{assoc}}}{\partial T}\right) &= \left(\frac{\partial Q_{sp}}{\partial T}\right) \\ &= -\frac{1}{2V} \sum_{i,j} \sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j} \left(\frac{\partial \Delta^{A_iB_j}}{\partial T}\right). \end{aligned} \tag{21}$$

3.3 Composition derivative

$$\left(\frac{\partial F^{\text{assoc}}}{\partial n_k}\right) = \left(\frac{\partial Q_{sp}}{\partial n_k}\right)$$

$$= \sum_{A_k} (\ln X_{A_k} - X_{A_k} + 1) - \frac{1}{V} \sum_j \sum_{A_k,B_j} n_j X_{A_k} X_{B_j} \Delta^{A_kB_j} \tag{22}$$

$$- \frac{1}{2V} \sum_{i,j} \sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j} \left(\frac{\partial \Delta^{A_iB_j}}{\partial n_k}\right) \tag{23}$$

$$= \sum_{A_k} \ln X_{A_k} - \frac{1}{2V} \sum_{i,j} \sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j} \left(\frac{\partial \Delta^{A_iB_j}}{\partial n_k}\right). \tag{24}$$

[2] Where $F^{\text{assoc}}(T, V, \mathbf{n}) = A^R(T, V, \mathbf{n})/RT$ and A^R is the association contribution to the residual Helmholtz energy.

4 Second-order derivatives of F^{assoc}

Let the variables ζ_1, ζ_2 each equal one of the scalar variables in $(T, V, \mathbf{n})$. Recalling that

$$\left(\frac{\partial F^{assoc}}{\partial \zeta_1}\right)(T, V, \mathbf{n}) = \left(\frac{\partial Q}{\partial \zeta_1}\right)(T, V, \mathbf{n}, \mathbf{X}(T, V, \mathbf{n})),$$

we get

$$\left(\frac{\partial^2 F^{\text{assoc}}}{\partial \zeta_2 \partial \zeta_1}\right) = \frac{\partial}{\partial \zeta_2}\left(\frac{\partial F^{\text{assoc}}}{\partial \zeta_1}\right) \tag{25}$$

$$= \left(\frac{\partial^2 Q_{sp}}{\partial \zeta_2 \partial \zeta_1}\right) + \left(\frac{\partial^2 Q_{sp}}{\partial \zeta_1 \partial \mathbf{X}}\right)\left(\frac{\partial \mathbf{X}}{\partial \zeta_2}\right). \tag{26}$$

We once again stress the meaning of our notation: in the first term of (26), $\mathbf{X}$ is to be treated as a constant when the cross-derivative is taken. Now, the expression (26) involves the derivative $\partial \mathbf{X}/\partial \zeta_2$. To find this derivative, we differentiate the defining relation for $X(T, V, \mathbf{n})$, namely $\left(\frac{\partial Q}{\partial \mathbf{X}}\right) = \mathbf{0}$. Doing this (and taking care to transpose vectors correctly), we get

$$\mathbf{0} = \frac{\partial}{\partial \zeta_2}\left(\frac{\partial Q_{sp}}{\partial \mathbf{X}}\right) = \left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X} \partial \zeta_2}\right) + \left(\frac{\partial \mathbf{X}}{\partial \zeta_2}\right)^t \left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X}^2}\right), \tag{27}$$

yielding

$$\left(\frac{\partial \mathbf{X}}{\partial \zeta_2}\right) = -\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X}^2}\right)^{-1}\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X} \partial \zeta_2}\right)^t. \tag{28}$$

In conclusion, the formula for the second derivative is obtained by combining (26) and (28):

$$\left(\frac{\partial^2 F^{\text{assoc}}}{\partial \zeta_2 \partial \zeta_1}\right) = \left(\frac{\partial^2 Q_{sp}}{\partial \zeta_2 \partial \zeta_1}\right) - \left(\frac{\partial^2 Q_{sp}}{\partial \zeta_1 \partial \mathbf{X}}\right)\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X}^2}\right)^{-1}\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X} \partial \zeta_2}\right)^t. \tag{29}$$

Or, if one prefers summation notation:

$$\left(\frac{\partial^2 F^{\text{assoc}}}{\partial \zeta_2 \partial \zeta_1}\right) = \left(\frac{\partial^2 Q_{sp}}{\partial \zeta_2 \partial \zeta_1}\right) - \sum_{i,j}\sum_{A_i, B_j}\left(\frac{\partial^2 Q_{sp}}{\partial \zeta_1 \partial X_{A_i}}\right)\left(\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X}^2}\right)^{-1}\right)_{ij}\left(\frac{\partial^2 Q_{sp}}{\partial X_{B_j} \partial \zeta_2}\right). \tag{30}$$

4.1 Formulas for $\left(\frac{\partial^2 Q_{sp}}{\partial \zeta_2 \partial \zeta_1}\right)$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T^2}\right) = -\frac{1}{2V}\sum_{i,j}\sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial T^2}\right).$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T \partial V}\right) = \frac{1}{2V}\sum_{i,j}\sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j}\left[\frac{1}{V}\left(\frac{\partial \Delta^{A_i B_j}}{\partial T}\right) - \left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial T \partial V}\right)\right]$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T \partial n_k}\right) = -\frac{1}{V}\sum_{j}\sum_{A_k, B_j} n_j X_{A_k} X_{B_j}\left(\frac{\partial \Delta^{A_k B_j}}{\partial T}\right) - \frac{1}{2V}\sum_{i,j}\sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial T \partial n_k}\right)$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right) = \frac{1}{2V}\sum_{i,j}\sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j}\left[-\frac{2\Delta^{A_i B_j}}{V^2} + \frac{2}{V}\left(\frac{\partial \Delta^{A_i B_j}}{\partial V}\right) - \left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial V^2}\right)\right]$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial V \partial n_k}\right) = \sum_j \sum_{A_k,B_j} n_j X_{A_k} X_{B_j} \left[\frac{\Delta^{A_k B_j}}{V^2} - \frac{1}{V}\left(\frac{\partial \Delta^{A_k B_j}}{\partial V}\right)\right] + \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left[\frac{1}{V}\left(\frac{\partial \Delta^{A_i B_j}}{\partial n_k}\right) - \left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial V \partial n_k}\right)\right].$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial n_l \partial n_k}\right) = -\frac{1}{V}\sum_{A_k,B_l} X_{A_k} X_{B_l} \Delta^{A_k B_l} - \frac{1}{V}\sum_j \sum_{A_l,B_j} n_j X_{A_l} X_{B_j}\left(\frac{\partial \Delta^{A_l B_j}}{\partial n_k}\right) - \frac{1}{V}\sum_j \sum_{A_k,B_j} n_j X_{A_k} X_{B_j}\left(\frac{\partial \Delta^{A_k B_j}}{\partial n_l}\right) - \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_i B_j}}{\partial n_l \partial n_k}\right)$$

These derivatives are all found by performing one more differentiation on the first-order derivatives we found in section 3. However, when taking an additional derivative of $\left(\frac{\partial Q_{sp}}{\partial n_k}\right)$, we have to take care to differentiate the expressions on (22) and (23), and not the simplified expression (24). This is because the supscript sp means that $X(T, V, \mathbf{n})$ should be substituted in *after* all the derivatives have been performed.

4.2 Formulas for $\left(\frac{\partial^2 Q_{sp}}{\partial X_{A_i} \partial \zeta_1}\right)$

We have

$$\left(\frac{\partial Q}{\partial X_{A_i}}\right) = \frac{n_i}{X_{A_i}} - n_i - \frac{n_i}{V}\sum_j \sum_{B_j} n_j X_{B_j} \Delta^{A_i B_j}, \tag{31}$$

and thus

$$\left(\frac{\partial^2 Q_{sp}}{\partial T \partial X_{A_i}}\right) = -\frac{n_i}{V}\sum_j \sum_{B_j} n_j X_{B_j}\left(\frac{\partial \Delta^{A_i B_j}}{\partial T}\right)$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial V \partial X_{A_i}}\right) = n_i \sum_j \sum_{B_j} n_j X_{B_j}\left[\frac{1}{V^2}\Delta^{A_i B_j} - \frac{1}{V}\left(\frac{\partial \Delta^{A_i B_j}}{\partial V}\right)\right]$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial n_l \partial X_{A_i}}\right) = -\frac{n_i}{V}\sum_{B_l} X_{B_l} \Delta^{A_i B_l} - \frac{1}{V}\sum_j \sum_{B_j} n_j X_{B_j} n_i \left(\frac{\partial \Delta^{A_i B_j}}{\partial n_l}\right).$$

4.3 Solving for $\left(\frac{\partial \mathbf{X}}{\partial V}\right)$ and $\left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right)$ simultaneously

The derivatives $\left(\frac{\partial \mathbf{X}}{\partial V}\right)$ and $\left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right)$ are needed in the Newton iteration when solving for volume given pressure, temperature and composition. When both of these are needed, one wants to first solve for $\left(\frac{\partial \mathbf{X}}{\partial V}\right)$ from (28), and then use (26) to find $\left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right)$, and therefore a dedicated routine for this has been implemented. To obtain $\left(\frac{\partial \mathbf{X}}{\partial V}\right)$, we solve the linear system

$$\left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X}^2}\right)\left(\frac{\partial \mathbf{X}}{\partial V}\right) = -\left(\frac{\partial^2 Q_{sp}}{\partial V \partial \mathbf{X}}\right)^t. \tag{32}$$

where[3]

$$\left(\frac{\partial^2 Q}{\partial X_{A_i} \partial X_{B_j}}\right) = -\frac{n_i}{X_{A_i}^2}\delta_{A_i B_j} - \frac{n_i n_j}{V}\Delta^{A_i B_j}. \tag{33}$$

Having found this derivative, we find $\left(\frac{\partial P}{\partial V}\right)$ from (26):

$$\left(\frac{\partial^2 F^{\text{assoc}}}{\partial V^2}\right) = \left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right) + \left(\frac{\partial^2 Q_{sp}}{\partial \mathbf{X} \partial V}\right)\left(\frac{\partial \mathbf{X}}{\partial V}\right). \tag{34}$$

[3] Note that only the diagonal of $\left(\frac{\partial^2 Q}{\partial X_{A_i} \partial X_{B_j}}\right)$ is dependent on $\mathbf{X}$.

5 Derivatives when using the CPA-form of $\Delta^{A_iB_j}(T, V, \mathbf{n})$

We now specialize to a specific functional form of $\Delta^{A_iB_j}(T, V, \mathbf{n})$, namely that used in the CPA-model. It is given by

$$\Delta^{A_iB_j}(T, V, \mathbf{n}) = g(V, \mathbf{n}) \cdot [\exp(\epsilon^{A_iB_j}/RT) - 1]b_{ij}\beta^{A_iB_j}, \tag{35}$$

where $\epsilon^{A_iB_j}$, b_{ij} and $\beta^{A_iB_j}$ are constants. The first derivatives are thus given by

$$\left(\frac{\partial \Delta^{A_iB_j}}{\partial T}\right) = -\frac{\epsilon^{A_iB_j}}{RT^2} g(V, \mathbf{n}) \exp(\epsilon^{A_iB_j}/RT) b_{ij}\beta^{A_iB_j} \tag{36}$$

$$\left(\frac{\partial \Delta^{A_iB_j}}{\partial V}\right) = \left(\frac{\partial g(V, \mathbf{n})}{\partial V}\right) \cdot [\exp(\epsilon^{A_iB_j}/RT) - 1]b_{ij}\beta^{A_iB_j} = \left(\frac{\partial \ln g(V, \mathbf{n})}{\partial V}\right) \Delta^{A_iB_j} \tag{37}$$

$$\left(\frac{\partial \Delta^{A_iB_j}}{\partial n_k}\right) = \left(\frac{\partial g(V, \mathbf{n})}{\partial n_k}\right) \cdot [\exp(\epsilon^{A_iB_j}/RT) - 1]b_{ij}\beta^{A_iB_j} = \left(\frac{\partial \ln g(V, \mathbf{n})}{\partial n_k}\right) \Delta^{A_iB_j} \tag{38}$$

while the second derivatives are given by

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial T^2}\right) = g(V, \mathbf{n}) \left(2 + \frac{\epsilon^{A_iB_j}}{RT}\right) b_{ij}\beta^{A_iB_j} \frac{\epsilon^{A_iB_j}}{RT^3} \exp(\epsilon^{A_iB_j}/RT) \tag{39}$$

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial V \partial T}\right) = -\frac{\epsilon^{A_iB_j}}{RT^2} g(V, \mathbf{n}) \exp(\epsilon^{A_iB_j}/RT) b_{ij}\beta^{A_iB_j} \left(\frac{\partial g(V, \mathbf{n})}{\partial V}\right) \tag{40}$$

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial n_l \partial T}\right) = -\frac{\epsilon^{A_iB_j}}{RT^2} g(V, \mathbf{n}) \exp(\epsilon^{A_iB_j}/RT) b_{ij}\beta^{A_iB_j} \left(\frac{\partial g(V, \mathbf{n})}{\partial n_l}\right) \tag{41}$$

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial V^2}\right) = \left(\frac{\partial^2 g(V, \mathbf{n})}{\partial V^2}\right) \frac{\Delta^{A_iB_j}}{g(V, \mathbf{n})} \tag{42}$$

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial n_l \partial V}\right) = \left(\frac{\partial^2 g(V, \mathbf{n})}{\partial n_l \partial V}\right) \frac{\Delta^{A_iB_j}}{g(V, \mathbf{n})} \tag{43}$$

$$\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial n_l \partial n_k}\right) = \left(\frac{\partial^2 g(V, \mathbf{n})}{\partial n_l \partial n_k}\right) \frac{\Delta^{A_iB_j}}{g(V, \mathbf{n})} \tag{44}$$

Derivatives of $g(V, \mathbf{n})$

In all common variants of CPA, the radial distribution function takes the special functional form $g(V, \mathbf{n}) = g(\eta)$, where η is the adimensional number

$$\eta = B(\mathbf{n})/4V = b\rho/4, \quad \text{where} \quad B(\mathbf{n}) = nb = \sum_i n_i b_i.$$

Note that $\mathbf{n}$ here includes all components, also the non-associating ones. Two common variants for $g(\eta)$, along with their first and second derivatives, are given by

$$g(\eta) = \frac{1}{1 - 1.9\eta}, \qquad g'(\eta) = \frac{1.9}{(1 - 1.9\eta)^2}, \qquad g''(\eta) = \frac{2 \cdot 1.9^2}{(1 - 1.9\eta)^3} \qquad (\textbf{sCPA})$$

$$g(\eta) = \frac{1 - \eta/2}{(1 - \eta)^3}, \qquad g'(\eta) = \frac{2.5 - \eta}{(1 - \eta)^4}, \qquad g''(\eta) = \frac{3\eta - 9}{(1 - \eta)^5} \qquad (\textbf{original})$$

Its derivatives with respect to variables ζ_1, ζ_2, each being a member of $(T, \mathbf{n})$, are given by

$$\left(\frac{\partial g}{\partial \zeta_1}\right) = g'(\eta) \left(\frac{\partial \eta}{\partial \zeta_1}\right), \qquad \left(\frac{\partial^2 g}{\partial \zeta_1 \partial \zeta_2}\right) = g''(\eta) \left(\frac{\partial \eta}{\partial \zeta_1}\right) \left(\frac{\partial \eta}{\partial \zeta_2}\right) + g'(\eta) \left(\frac{\partial^2 \eta}{\partial \zeta_1 \partial \zeta_2}\right), \tag{45}$$

and

$$\left(\frac{\partial \eta}{\partial V}\right) = -\frac{B(\mathbf{n})}{4V^2} \qquad \left(\frac{\partial \eta}{\partial n_k}\right) = \frac{b_k}{4V}, \tag{46}$$

$$\left(\frac{\partial^2 \eta}{\partial V^2}\right) = \frac{B(\mathbf{n})}{2V^3} \qquad \left(\frac{\partial^2 \eta}{\partial n_l \partial V}\right) = -\frac{b_k}{4V^2} \qquad \left(\frac{\partial^2 \eta}{\partial n_k \partial n_l}\right) = 0. \tag{47}$$

5.1 Simplified CPA Derivatives

Simplified first derivatives

$$\begin{aligned}\left(\frac{\partial F^{\text{assoc}}}{\partial V}\right) =&\frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left[\frac{\Delta^{A_iB_j}}{V}-\left(\frac{\partial \Delta^{A_iB_j}}{\partial V}\right)\right] && (48)\\ =&\frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\Delta^{A_iB_j}\left[\frac{1}{V}-\left(\frac{\partial \ln g}{\partial V}\right)\right] && (49)\\ =&\frac{1}{2}\left[\frac{1}{V}-\left(\frac{\partial \ln g}{\partial V}\right)\right]\sum_{i}\sum_{A_i} n_i(1-X_{A_i}) && (50)\end{aligned}$$

$$\left(\frac{\partial F^{\text{assoc}}}{\partial T}\right) = -\frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial \Delta^{A_iB_j}}{\partial T}\right). \tag{51}$$

$$\begin{aligned}\left(\frac{\partial F^{\text{assoc}}}{\partial n_k}\right) =&\sum_{A_k}\ln X_{A_k} - \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial \Delta^{A_iB_j}}{\partial n_k}\right) && (52)\\ =&\sum_{A_k}\ln X_{A_k} - \left(\frac{\partial \ln g}{\partial n_k}\right)\frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\Delta^{A_iB_j} && (53)\\ =&\sum_{A_k}\ln X_{A_k} - \frac{1}{2}\left(\frac{\partial \ln g}{\partial n_k}\right)\sum_{i}\sum_{A_i} n_i(1-X_{A_i}). && (54)\end{aligned}$$

Simplified formulas for $\left(\frac{\partial^2 Q_{sp}}{\partial X_{A_i}\partial \zeta_2}\right)$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T\partial X_{A_i}}\right) = -\frac{1}{V}\sum_{j}\sum_{B_j} n_i n_j X_{B_j}\left(\frac{\partial \Delta^{A_iB_j}}{\partial T}\right)$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial V\partial X_{A_i}}\right) =&\sum_{j}\sum_{B_j} n_i n_j X_{B_j}\left[\frac{1}{V^2}\Delta^{A_iB_j}-\frac{1}{V}\left(\frac{\partial \Delta^{A_iB_j}}{\partial V}\right)\right]\\ =&\left[\frac{1}{V}-\left(\frac{\partial \ln g}{\partial V}\right)\right]\frac{1}{V}\sum_{j}\sum_{B_j} n_i n_j X_{B_j}\Delta^{A_iB_j}\\ =&\left[\frac{1}{V}-\left(\frac{\partial \ln g}{\partial V}\right)\right]\left(\frac{1}{X_{A_i}}-1\right)\end{aligned}$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial n_l\partial X_{A_i}}\right) =&-\frac{1}{V}\sum_{B_l} n_i X_{B_l}\Delta^{A_iB_l} - \frac{1}{V}\sum_{j}\sum_{B_j} n_j X_{B_j}\left[\delta_{il}\Delta^{A_iB_j} + n_i\left(\frac{\partial \Delta^{A_iB_j}}{\partial n_l}\right)\right]\\ =&-\frac{1}{V}\sum_{B_l} n_i X_{B_l}\Delta^{A_iB_l} - \left[\delta_{il} + n_i\left(\frac{\partial g}{\partial n_l}\right)\right]\frac{1}{V}\sum_{j}\sum_{B_j} n_j X_{B_j}\Delta^{A_iB_j}\\ =&-\frac{1}{V}\sum_{B_l} n_i X_{B_l}\Delta^{A_iB_l} - \left[\delta_{il} + n_i\left(\frac{\partial g}{\partial n_l}\right)\right]\left(\frac{1}{X_{A_i}}-1\right)\end{aligned}$$

Simplified formulas for $\left(\frac{\partial^2 Q_{sp}}{\partial \zeta_1\partial \zeta_2}\right)$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T^2}\right) = -\frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial T^2}\right).$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial V^2}\right) &= \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left[-\frac{2\Delta^{A_iB_j}}{V^2} + \frac{2}{V}\left(\frac{\partial \Delta^{A_iB_j}}{\partial V}\right) - \left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial V^2}\right)\right] \\ &= \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\Delta^{A_iB_j}\left[-\frac{2}{V^2} + \frac{2}{V}\left(\frac{\partial \ln g(V,\mathbf{n})}{\partial V}\right) - \left(\frac{\partial^2 g(V,\mathbf{n})}{\partial V^2}\right)\frac{1}{g(V,\mathbf{n})}\right] \\ &= \frac{1}{2}\sum_{i}\sum_{A_i} n_i\left(\frac{1}{X_{A_i}} - 1\right)\left[-\frac{2}{V^2} + \frac{2}{V}\left(\frac{\partial \ln g(V,\mathbf{n})}{\partial V}\right) - \left(\frac{\partial^2 g(V,\mathbf{n})}{\partial V^2}\right)\frac{1}{g(V,\mathbf{n})}\right]\end{aligned}$$

$$\left(\frac{\partial^2 Q_{sp}}{\partial T \partial V}\right) = \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left[\frac{1}{V}\left(\frac{\partial \Delta^{A_iB_j}}{\partial T}\right) - \left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial T \partial V}\right)\right]$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial T \partial n_k}\right) =& \frac{1}{V}\sum_{j}\sum_{A_k,B_j} n_j X_{A_k} X_{B_j}\left(\frac{\partial \Delta^{A_kB_j}}{\partial T}\right) \\ &- \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial T \partial n_k}\right)\end{aligned}$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial V \partial n_k}\right) =& \sum_{j}\sum_{A_k,B_j} n_j X_{A_k} X_{B_j}\left[\frac{\Delta^{A_kB_j}}{V^2} - \frac{1}{V}\left(\frac{\partial \Delta^{A_kB_j}}{\partial V}\right)\right] \\ &+ \sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left[\frac{1}{V^2}\left(\frac{\partial \Delta^{A_iB_j}}{\partial n_k}\right) - \frac{1}{2V}\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial V \partial n_k}\right)\right].\end{aligned}$$

$$\begin{aligned}\left(\frac{\partial^2 Q_{sp}}{\partial n_l \partial n_k}\right) =& -\frac{1}{V}\sum_{A_k,B_l} X_{A_k} X_{B_l}\Delta^{A_kB_l} - \frac{1}{V}\sum_{j}\sum_{A_i,B_j} n_j X_{A_l} X_{B_j}\left(\frac{\partial \Delta^{A_lB_j}}{\partial n_k}\right) \\ &- \frac{1}{V}\sum_{j}\sum_{A_i,B_j} n_j X_{A_k} X_{B_j}\left(\frac{\partial \Delta^{A_kB_j}}{\partial n_l}\right) - \frac{1}{2V}\sum_{i,j}\sum_{A_i,B_j} n_i n_j X_{A_i} X_{B_j}\left(\frac{\partial^2 \Delta^{A_iB_j}}{\partial n_l \partial n_k}\right)\end{aligned}$$

6 Efficient implementation of the association contribution

We now describe an efficient implementation of the association contribution, which is outlined in the papers by Michelsen [3] and [4].

Equation of state models with an association contribution are computationally expensive as they have to solve an internal chemical equilibrium problem. However, using that the association contribution to the Helmholtz energy can be found from a certain minimization procedure, will be used to simplify the calculation of properties like pressure and chemical potentials, together with their derivatives with respect to temperature, volume and composition.

6.1 Base equations

In the SAFT and CPA models, the association contribution to the mixture Helmholtz energy is found from

$$\frac{A^{\text{assoc}}}{RT} = \sum_{i} n_i \sum_{A_i}\left(\ln X_{A_i} - \frac{1}{2}X_{A_i} + \frac{1}{2}\right). \tag{55}$$

Here, A and B index bonding sites on a given molecule, and X_{A_i} denotes the fraction of A-sites on molecule i that do NOT form bonds with other active sites. These site fractions are given implicitly by the nonlinear equations

$$X_{A_i} = \frac{1}{1 + \sum_j \sum_{B_j} x_j X_{B_j} \Delta^{A_i B_j}}, \tag{56}$$

where $\Delta^{A_i B_j}$ is the association strength between site A on molecule i and site B on molecule j, and depends on T, V and $\mathbf{n}$, but NOT on the fraction of sites that form bonds. Differentiation of (55) yields derived properties (i.e. the contribution from the association term to the derived properties), e.g.

$$\frac{P^{\text{assoc}}}{RT} = -\frac{\partial}{\partial V}\left(\frac{A^{\text{assoc}}}{RT}\right) = \sum_i n_i \sum_{A_i} \left(\frac{1}{X_{A_i}} - \frac{1}{2}\right) \frac{\partial X_{A_i}}{\partial V}. \tag{57}$$

We will now see, however, that by taking advantage of a certain minimization problem, there is a computationally cheaper expression for the association pressure.

Now, define Q by

$$Q(\mathbf{n}, T, V, \mathbf{X}(\mathbf{n}, T, V)) = \sum_i \sum_{A_i} n_i \left(\ln X_{A_i} - X_{A_i} + 1\right) - \frac{1}{2V} \sum_{i,j} \sum_{A_i, B_j} n_i n_j X_{A_i} X_{B_j} \Delta^{A_i B_j}. \tag{58}$$

The reason for introducing the Q function, is that the association contribution of SAFT and CPA to the reduced, residual Helmholtz energy, equals the value of Q at a stationary point with respect to the site fractions $\mathbf{X}$; i.e. at a point where the gradient of Q with respect to the $\mathbf{X}$-coordinates vanishes. This is shown in Michelsen [3], in addition to where Q comes from in the first place. The association contribution to pressure is thus found from

$$\frac{P^{\text{assoc}}}{RT} = -\frac{\partial Q_{sp}}{\partial V},$$

where Q_{sp} is a stationary point with respect to the $\mathbf{X}$-variables. To find the derivative on the right hand side, we use the chain rule:

$$\frac{\partial Q_{sp}}{\partial V} = \left(\frac{\partial Q}{\partial V}\right)_{\mathbf{X}} + \sum_i \sum_{A_i} \left(\frac{\partial Q}{\partial X_{A_i}}\right)_V \frac{\partial X_{A_i}}{\partial V}$$

but since the derivatives with respect to X_{A_i} are 0 at the stationary point, we get $P^{\text{assoc}} = -RT\left(\frac{\partial Q}{\partial V}\right)_{\mathbf{X}}$. When this is differentiated out, we get

$$P^{assoc} = -\frac{RT}{2V}\left(1 - V\left(\frac{\partial \ln g}{\partial V}\right)\right) \sum_i \sum_{A_i} n_i (1 - X_{A_i}). \tag{59}$$

Although the notation we have used up to now is the traditional one, it is unwieldy to use in what follows. We therefore follow Michelsen [4] and use the following notation

- S is the total number of different sites for all molecules
- The totality of association sites on all molecules are indexed sequentially, $k = 1, 2, \ldots, S$
- m_k is the total number of moles of molecules that host a given site k
- $K_{kl} = K_{lk} = m_l m_k \Delta^{lk}/V$

Using this notation, we can write

$$Q(\mathbf{X}, \mathbf{m}) = \sum_{k=1}^{S} m_k (\ln X_k - X_k + 1) - \frac{1}{2} \sum_{k=1}^{S} \sum_{l=1}^{S} K_{kl} X_k X_l. \tag{60}$$

When $Q = Q(\mathbf{X}, \mathbf{m})$ is written in this form, its $\mathbf{X}$-derivatives are given by

$$g_k := \left(\frac{\partial Q}{\partial X_k}\right) = m_k \left(\frac{1}{X_k} - 1\right) - \sum_{l=1}^{S} K_{kl} X_l, \tag{61}$$

while its Hessian matrix with respect to **X** is

$$H_{kl} := \left(\frac{\partial^2 Q}{\partial X_k \partial X_l}\right) = -\frac{m_k}{X_k^2}\delta_{kl} - K_{kl}. \tag{62}$$

Observe that, apart from the diagonal, the Hessian matrix is independent of **X**. The first step in determining **X** is to compute **K** from the component parameters and the Δ-function.

6.2 Solution procedure for X

An attractive way to solve for **X** is formulating it as a maximation procedure. More precisely, one utilizes the fact that for a given T, V and $\mathbf{n}$, the correct value of **X** is the value for which Q is maximized. The maximization is unconstrained, the maximum unique, and global convergence can be assured. Michelsen [4] suggests the following quasi-Newton iteration scheme:

$$\hat{\mathbf{H}}\mathbf{\Delta X} + \mathbf{g} = 0, \tag{63}$$

where $\hat{\mathbf{H}}$ is a modified Hessian matrix with the following properties

(i) It is negative definite for all **X**

(ii) $\hat{\mathbf{H}} \to \mathbf{H}$ as **X** approaches the solution

Property (i) ensures that $\mathbf{\Delta X}$ is an ascent *direction*, because it has a positive projection along the gradient: $\mathbf{g}^T(-\hat{\mathbf{H}}^{-1}\mathbf{g}) > 0$. However, overstepping is still a possibility, and in that case one can use a linesearch method, or, as we will do, simply bisect the step until an increase in Q is obtained. Property (ii) ensures quadratic convergence.

The modification is performed as follows: From equations (61) and (62), we see that the diagonal contribution of the hessian can be written as

$$\frac{m_k}{X_k^2} = \frac{1}{X_k}\frac{m_k}{X_k} = \frac{1}{X_k}\left(m_k + \sum_{l=1}^{S} K_{kl}X_l + g_k\right).$$

To get the modified hessian **H** we simply drop the gradient contribution, giving

$$\hat{H}_{kl} := \left(\frac{\partial^2 Q}{\partial X_k \partial X_l}\right) = -\frac{1}{X_k}\left(m_k + \sum_{l=1}^{S} K_{kl}X_l\right)\delta_{kl} - K_{kl}. \tag{64}$$

That $\hat{H}$ fulfills property (i) is shown in Michelsen [4], while property (ii) is obvious since $\mathbf{g} = 0$ at the solution. In conclusion, Michelsen [4] suggests the following approach for solving for **X**:

(1) Choose an initial estimate of **X**.

(2) Calculate $\mathbf{\Delta X}$ from equation (63).

(3) Set $\mathbf{X}^{new} = \max(\mathbf{X}^{old} + \mathbf{\Delta X}, 0.2\mathbf{X}^{old})$, denying more than 80% reduction in any component.

(4) Test that $\mathbf{X}^{new} > 0$, and that the objective function is increased: $Q(\mathbf{X}^{new}) > Q(\mathbf{X}^{old})$.

(5) If (4) is violated, set $\mathbf{\Delta X} = \frac{1}{2}\mathbf{\Delta X}$ and repeat from step (3).

(6) Check for convergence. If not converged, set $\mathbf{X}^{old} = \mathbf{X}^{new}$ and repeat from step (2).

We have also implemented a back-up procedure if the maximization approach (1)-(6) should fail to converge in a few iterations. The back-up is the method of **damped successive substitutions**. This iteration scheme is defined as follows:

$$\mathbf{X}^{(n+1)} = (1-\omega)\mathbf{f}(\mathbf{X}^{(n)}) + \omega\mathbf{X}^{(n)}, \quad \text{where} \quad f_k(\mathbf{X}^{(n)}) := \frac{m_k}{m_k + \sum_{l=1}^{S} K_{kl}X_l^{(n)}}. \tag{65}$$

Although the damping parameter ω in principle can be tailored to the specific **f**, we will follow Michelsen [4] and set $\omega = 0.2$. With this terminology, the back-up procedure can be described as follows

(1') Set all elements $X_k = 0.2$ as the initial estimate.

(2') Perform five iterations of successive substitutions with damping factor $\omega = 0.2$.

(3') Use the second-order approach above to converge the equations to desired accuracy.

6.3 Solution procedure for molar volume v

When solving for volume, a twofold nested calculation loop is required. The molar volume v is adjusted in the outer loop, while the association equations must be solved for the matrix X corresponding to the assumed volume in the inner loop. Having found X in the inner loop, we will calculate not only the pressure contribution from the association term $P^{asc} = -RT\left(\frac{\partial Q}{\partial V}\right)$, but also the two derivatives

$$\left(\frac{\partial \mathbf{X}}{\partial V}\right) \quad \text{and} \quad \left(\frac{\partial P^{asc}}{\partial V}\right). \tag{66}$$

The derivative $\left(\frac{\partial \mathbf{X}}{\partial V}\right)$ is found from differentiating $\mathbf{g}(\mathbf{X}(V), V) = 0$, which by the chain rule yields

$$\mathbf{H}\left(\frac{\partial \mathbf{X}}{\partial V}\right) + \left(\frac{\partial \mathbf{g}}{\partial V}\right) = 0. \tag{67}$$

Here $\mathbf{H}$ is already found when solving for $\mathbf{X}$, seeing as $\mathbf{H} = \hat{\mathbf{H}} - \text{diag}(\mathbf{g_k}/\mathbf{X_k})_{\mathbf{k=1}}^{\mathbf{n}}$. The derivative $\left(\frac{\partial P^{asc}}{\partial V}\right)$ is then found from

$$-\frac{1}{RT}\left(\frac{\partial P^{asc}}{\partial V}\right) = \left(\frac{\partial^2 Q}{\partial V^2}\right)_{\mathbf{X}} + \left(\frac{\partial \mathbf{g}}{\partial V}\right)^T \left(\frac{\partial \mathbf{X}}{\partial V}\right). \tag{68}$$

The derivative of the association pressure is used in the outer loop to solve for volume using a Newton-based method, and the volume derivative of $\mathbf{X}$, which is obtained as a byproduced in the calculation of $\left(\frac{\partial P^{asc}}{\partial V}\right)$, is used to create initial estimates (step (1) above) for the inner solution loop for $\mathbf{X}$. When a correction ΔV has been determined from the Newton iteration in the outer loop, we use, as an initial estimate for the inner loop,

$$\mathbf{X}^{(n+1)} = \mathbf{X}^{(n)} + \Delta V\left(\frac{\partial \mathbf{X}}{\partial V}\right). \tag{69}$$

The tolerance for accepting an inner loop solution in the volume iteration is set fairly loose, and consequently only a single inner-loop iteration is necessary in most cases.

Iterating on the reduced density ζ

Michelsen also offers the following suggestions for a CPA volume solver. First, a robust volume iteration should use the reduced density $\zeta = b/v$ as the independent variable. Choose, as the equation to be solved to 0, not $\zeta \mapsto P(\zeta) - P^{spec}$, but as $F(\zeta) = (1-\zeta)(P(\zeta) - P^{spec})$.

1. Initialization. For the liquid phase, $\zeta = 0.99$ is to be used as initial estimate. For the vapor phase, use $\zeta = b/(b + (RT/P))$. The initial limits should be set to $\zeta_{min} = 0$, $\zeta_{max} = 1$.

2. At step k, calculate a new value according to Newton's method:

$$\zeta_{new} = \zeta_k - \frac{(1-\zeta_k)(P(\zeta_k) - P^{spec})}{P^{spec} - P + (1-\zeta_k)\left(\frac{\partial P}{\partial V}\right)_{T,\mathbf{n}} V_k/\zeta_k}. \tag{70}$$

3. If $\zeta_{min} < \zeta_{new} < \zeta_{max}$, take $\zeta_{k+1} = \zeta_{new}$. Otherwise, take $\zeta_{k+1} = (\zeta_{min} + \zeta_{new})/2$.

4. If $F(\zeta_{k+1}) > 0$, set $\zeta_{max} = \zeta_{new}$; otherwise, set $\zeta_{min} = \zeta_{new}$.

5. Continue until convergence.

7 Testing the CPA code

7.1 Mathematical consistency

We have implemented unittests for all the derivatives of all the functions occuring in the CPA implementation. Specifically, we have tested the following analytical derivatives against their numerical counterpart using finite differences:

- $g(V, \mathbf{n})$, first and second derivatives;

- $Q(T, V, \mathbf{n}, \mathbf{X})$, first and second derivatives;
- $\Delta^{A_i B_j}(T, V, \mathbf{n})$, first and second derivatives;
- $\mathbf{X}(T, V, \mathbf{n})$, first derivatives;
- $F(T, V, \mathbf{n})$, first and second derivatives;
- $P(T, V, \mathbf{n})$, first and second derivatives;
- $Z(T, V, \mathbf{n})$, first derivatives;
- $S^R(T, P, n)$, first derivatives;
- $G^R(T, P, n)$, first derivatives;
- $H^R(T, P, n)$, first derivatives;
- $\ln(\phi)(T, P, n)$, first derivatives.

7.2 Thermodynamic consistency

We have set up the supertest cpa_consistency, which is passed with reasonable tolerances. Thus, all of ThermoPack's implemented thermodynamic identities are fulfilled to a satisfactory accuracy.

7.3 Testing physical predictions

We tested what SRK-CPA gave as the liquid density of water at 277 K, and compared it with SRK and PR.

- SRK-CPA: 1019 kg/m^3.
- SRK: 766 kg/m^3.
- PR: 859 kg/m^3.

We thus see that the error of SRK-CPA (1.9 %) is about one order of magnitude smaller than that of SRK (23.4 %) and PR (14.1 %).

7.4 Reduction to cubic equation

Although the CPA code does not handle pure components without association, we can trick the code to do so by assigning it an arbitrary association scheme, but with association parameters $\epsilon = \beta = 0$. By assigning a_0, b and c_1 to be what they are in the SRK equation, SRK-CPA and SRK should be equivalent. We have verified that the two EoS indeed seem to give the same results, by computing the pressure at given (T, V), and solving for the volume at given (T, P).

8 Overview of the CPA code

8.1 Cubic equations of state with fitted parameters

The usual formulation of PR and SRK for pure components is given by

$$P = \frac{RT}{v - b} - \frac{a_0 \alpha(T)}{(v + \delta_1 b)(v + \delta_2 b)},$$

with

$$a_0 = \Omega_a \frac{(RT_c)^2}{P_c}, \quad b = \Omega_b \frac{RT_c}{P_c}, \quad \alpha(T) = \left(1 + c_1(1 - \sqrt{T/T_c})\right)^2, \quad c_1 = m(\omega). \tag{71}$$

The constants $\delta_1, \delta_2, \Omega_a, \Omega_b$ and the function m are inherent to the equation of state. The critical temperature T_c, critical pressure P_c and acentric factor ω are properties of the component, and are needed to capture the peculiarities of the component. The specific forms of a_0 and b result from demanding that $dP/dV = d^2P/dV^2 = 0$ at $(T, P) = (T_c, P_c)$, while the function m was devised to fit the vapor pressure data of hydrocarbons.

Instead of using the formulas (71) to compute a_0, b and c_1, one can also simply fit them (e.g. using liquid volume and vapor pressure data). This is usually necessary when the cubic term is not the only term in the full equation of state (as in e.g. CPA), simply because the formulas (71) were derived under the assumption that the cubic contribution is the *only* contribution to pressure. Another approach is of course to retain the expressions (71), and to fit any parameters in the non-cubic terms accordingly. This latter approach does not seem reasonable, seeing as the first approach will always give a fit which is as least as good as the latter, and because the latter approach (71) essentially forces the cubic equation to try to do something it wasn't designed to do (e.g. correlate vapor pressure of self-associating compounds).

Kontogeorgis and Folas [6] have compiled fitted parameters for a_0, b and c_1 for use in the SRK-CPA equation. Their database covers both self-associating and non self-associating components. The parameters for some of the most common components have been recorded in ThermoPack, in the module cpa_parameters.

It should also be mentioned that although the classic alpha formulation $\alpha(T) = (1+c_1((1-\sqrt{T/T_c}))^2$ is most common, there is nothing in the way for using other alpha formulations, such as e.g. the well-known variants by Twu or Mathias.

8.2 The Helmholtz energy in ThermoPack

One needs to be careful when adding the contributions F^{cb} and F^{assoc} to get the total reduced residual Helmholtz energy F. In general we have that the reduced residual Helmholtz energy can be written as $F(T, V, \mathbf{n}) = k \cdot \tilde{F}(T, 1000V/k, \mathbf{n}/k)$, where k is a constant, which happens to equal $\sum_i n_i$. ThermoPack's cbhelm-module gives the derivatives of $\tilde{F}$. Thus, to get the cubic contribution F^{cb} to the total reduced residual Helmholtz energy, we need to use the following formulas:

$$F^{cb} = k \cdot \tilde{F} \tag{72}$$

$$F^{cb}_T = k \cdot \tilde{F}_T \tag{73}$$

$$F^{cb}_V = 1000 \cdot \tilde{F}_V \tag{74}$$

$$F^{cb}_n = \tilde{F}_n \tag{75}$$

$$F^{cb}_{TT} = k \cdot \tilde{F}_{TT} \tag{76}$$

$$F^{cb}_{TV} = 1000 \cdot \tilde{F}_{VT} \tag{77}$$

$$F^{cb}_{Tn} = \tilde{F}_{nT} \tag{78}$$

$$F^{cb}_{VV} = 1000 \cdot 1000 \cdot \tilde{F}_{VV}/k \tag{79}$$

$$F^{cb}_{Vn} = 1000 \cdot \tilde{F}_{nV}/k \tag{80}$$

$$F^{cb}_{nn}(i,j) = \tilde{F}_{nn}/k. \tag{81}$$

8.3 New modules, and changes in existing modules

Below we list the routines we have modified in order to implement CPA, ordered by module. This may be helpful if one is working in ThermoPack and sees some CPA-related code one doesn't understand.

cpa

cpa_set_scheme_and_fitted_parameters: This routine has different behavior depending on whether the mixture solvates or not. If it is not solvating, it does the following:

If the component is not in the CPA database, we use the standard (SRK or PR) values already

If the component is in the CPA database, the parameters in the database are used

cpa_init: Allocates memory, retrieves the relevant parameters from the module cpa_parameters, and finally uses the parameters together with the inputted mixing rules to initialize the global variables in the cpa-module. Note that cpa_init is called after selectEOS in init_thermoPack.

The routines in this module can be separated into two categories: back-end routines and front-end routines. The back-end routines sometimes assume that other routines have been called prior to calling it, a typical example being a routine that takes in X_k and (T,V,n), and assumes that *solve_for_X_k* has been called so that the inputted X_k is consistent with (T,V,n). The dependencies of a back-end routine are given before the code for the routine. The front-end routines have no such dependencies.

eoslibinit

init_thermopack: The cpa model is used if eosstr(1:3)=”CPA”.*SelectEOS* is called before *cpa_init*, since the global cbeos instance has to be initialized before *cpa_init* is called. Moreover, *cpa_init* is only called for the first component in the cbeos-vector, since we don’t want to recalculate the global parameters in cpa.f90 over and over. For the other components, we call *cpa_set_scheme_and_fitted_parameters* (which involves some removable overhead since some things are done ncbeos times, but this isn’t critical since it is an initialization routine).

cpa_parameters

A record of pure-component fitted parameters, as well as binary interaction coefficients. Note that the parameters a_0, b and ω are often replaced by fitted parameters.

As of now, the module cpa_parameters is in the file cpa_parameters.f90, but eventually it should probably be moved to tpinputdb.f90, where all other parameters are stored.

We point out that some common associating components (e.g. ethanol) are not stored in the module compdatadb, and therefore ThermoPack can not be initialized with these components.

The c_1 parameter is used only when ThermoPack is initialized with the classic alpha formulation. The fitted parameters a_0 and b, if they exist in the database, are per now always used for CPA. However, if one wants to use the “standard” (SRK or PR) values for these parameters in terms of critical properties, then one can simply iterate through the global comp-vector, set a_0 and b equal to zero, and call the routine *cbCalcMixtureParams*. Using the fitted parameters if they exist seems like reasonable default behavior.

compdata

Added the three parameters b_cpa, a0_cpa and c1_cpa in the gendata struct (which is the struct holding component data). These are initially set to zero, and then possibly – depending on the options the user chooses and whether fitted values exist in the database – updated to their fitted values in cpa_parameters.

eosdata

Added the integer parameters cpaSRK and cpaPR, which will be set to eosidx in the global eoscubic-instance cbeos. Also added the integer parameter cbMixClassicCPA, which will be set to mruleidx in cbeos. The reason we need an own CPA-indicator for the classical (van der Waals) mixing rule, is that we need to retrieve the interaction parameters from the CPA_parameters database, and not the eosdatadb database where the other interaction parameters are stored.

tpselect

selectEOS: Initialize cbeos%eosidx to the integer parameter cpaSRK if eosstr=’CPA-SRK’. Similarly for CPA-PR. Also cbeos%mruleidx is set to cbMixClassicCPA.

tpSelectInteractionParameters: If cbeos%mruleidx=cbMixClassicCPA, retrieve interaction parameters from the cpa_parameters database if they exist. If not, use the interaction parameters from the usual eosdatadb database.

copyFromDB: Called from *SelectComp*, this helper function now also sets component%b_cpa=0.0, component%a0_cpa=0.0 and component%c1_cpa=0.0.

tpcbmix

cbCalcParameters: cpaSRK and cpaPR are assigned the same alpha, beta, delta and gamma as cbSRK and cbPR, respectively. However, (α, β, γ) are only used to compute $m(\omega) = \alpha + \beta\omega - \gamma\omega^2$ if c_1 isn't fitted (i.e. if it is set to 0.0 in the global eoscubic-instance cbeos).

cbCalcM: cpaSRK and cpaPR behaves the same way as cbSRK and cbPR, namely that cbeos%m1 and cbeos%m2 are calculated. The denominator in the attractive part of the cubic EoS is then m1 times m2.

cbCalcOmegaZc: This routine computes many single-component properties, amongst other a0 and b. Therefore we have to stipulate that, in the case of a CPA model, b_cpa and a0_cpa should be used.

cbCalcAmix: No significant changes; cbMixClassicCPA should behave the same way as cbMixClassic, i.e. call vanderWaalsMix.

cbAlpha

calcAlpha_classic_CPA: New routine. Does the same as calcAlpha_classic, except that instead of using $m(\omega) = \alpha + \beta\omega - \gamma\omega^2$, it uses the fitted parameter c_1 stored in the global comp-vector.

cbCalcAlphaTerm: If cbeos%eosidx equals cpaSRK or cpaPR, and if one uses the classical alpha correlation, then we should use the fitted parameter if it has been set, and the usual formulation α-formulation using $m(\omega)$ if not.

tpsingle

The only changes made were that the following functions also works with CPA: *TP_CalcZfac*, *TP_CalcEnthalpy*, *TP_CalcEntropy*, *TP_CalcFugacity*, *TP_CalcPressure*, *Gres.*

9 Adapting pure component parameters

There does not seem to be any readily available CPA-parameters for CO_2 modeled as an associating component. We have therefore fitted the five pure component parameters $a_0, b, c_1, \epsilon, \beta$ to PVT-points on CO_2's boiling point curve, using the association scheme 1 (i.e. modeling CO_2 as having one association site). The objective function minimized was

$$O(\Lambda) = \sum_i \left[\left(\frac{P_{bub}(T_i^{exp}; \Lambda) - P^{exp}}{P^{exp}} \right)^2 + \left(\frac{v_{liq}(T_i^{exp}, P_i;^{exp} \Lambda) - v_{liq}^{exp}}{v_{liq}^{exp}} \right)^2 \right],$$

where $\Lambda = (a_0, b, c_1, \epsilon, \beta)$ represents the adjustable parameters. According to [6] one should use PVT data on the boiling point curve ranging from reduced temperatures 0.5 to 0.95. Extending the reduced temperature to values close to one is of minor importance, since association models overpredict the critical temperature.

Note that we have chosen to minimize the squared deviations – not the squared *relative* deviations – since we want to weight more the deviations near the critical point (high pressures).

The resulting parameters have been stored in the parameter database, *cpa_parameters.f90.*

References

[1] Gross J., Sadowski G. Perturbed-Chain SAFT: An Equation of State Based on a Perturbation Theory for Chain Molecules. *Industrial and Engineering Chemistry Research* **2001**, 40, 1244–1260.

[2] Kontogeorgis, Georgis M. and Folas, Georgios K., Thermodynamic Models for Industrial Applications, Wiley 2010

[3] Michelsen M.L., Hendriks E.M. Physical Properties from Association Models. *Fluid Phase Equilibria* **2001**, 180, 165–174.

[4] Michelsen M.L. Robust and Efficient Solution Procedures for Association Models. *Industrial and Engineering Chemistry Research* **2006**, 45, 8449–8453.

[5] Müller E.A., Gubbins K.E. Molecular-Based Equations of State for Associating Fluids: A Review of SAFT and Related Approaches. *Industrial and Engineering Chemistry Research* **2001**, 40, 2193–2211.

[6] Kontogeorgis, G.M., Folas, G.K. *Thermodynamic Models for Industrial Applications.* Wiley 2010.