Super-Ancillary Equations for Cubic Equations of State[†]

Ian H. Bell[*,‡] and Ulrich K. Deiters[¶]

‡*Applied Chemicals and Materials Division, National Institute of Standards and Technology, Boulder, CO 80305, USA*
¶*Ulrich K. Deiters - Institute of Physical Chemistry, University of Cologne, 50939 Köln, Germany*

E-mail: ian.bell@nist.gov

Abstract

Calculation of thermodynamic phase equilibrium is error-prone and can fail both near the critical point and at very low temperatures due to the limited precision available in double precision arithmetic. Most importantly, these calculations frequently represent a computational bottleneck. In this work we extend the "super-ancillary" equation approach developed for reference multiparameter equations of state to classical cubic equations of state (van der Waals, Redlich-Kwong-Soave, Peng-Robinson). Iterative calculations in double precision are replaced by non-iterative evaluation of pre-built Chebyshev expansions constructed with extended precision arithmetic. Exact solutions for the equation of state constants are given. The Chebyshev expansions are shown to reproduce the equation of state values to within nearly double precision (aside from in the very near vicinity of the critical point) and are more than 40 times faster to evaluate than the VLE calculations from the fastest computational library. In this way we further expand the domains in which iterative calculations for pure fluid phase equilibria may be rendered obsolete. A C++ header implementing these expansions (and with no external dependencies) is provided as supplemental information.

1 Introduction

The cubic equations of state (EOS), although based on an outdated concept, retain an important place in engineering practice because of their robustness, generally accurate predictions, CPU time savings, and the ease with which parameters can be determined based on generally well known fluid constants.

A very frequently needed operation is the calculation of vapor-liquid equilibrium, which can only be done iteratively, is rather time-consuming, and is prone to failure. Analytic limiting cases for zero temperature and at the critical point have been derived for the van der Waals EOS.[1] For reference EOS (e.g., those implemented in the REFPROP,[2] CoolProp[3] and TREND[4] libraries) it is common practice to provide not only the EOS itself (usually the Helmholtz energy as a function of temperature and density), but also "ancillary equations" that yield good initial values for the vapor pressure and the equilibrium phase densities. Recently[5] it became possible to generate "super-ancillary equations" that provide these data with approximately machine precision so

[†]Commercial equipment, instruments, or materials are identified only in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose.

that vapor-liquid iterations are no longer required, resulting in significant speedup. Here we provide super-ancillary equations for some of the most popular cubic EOS.

1.1 Vapor–liquid phase equilibrium

The conditions for equilibrium between two homogeneous phases (denoted by $'$ and $''$) of a pure substance are

$$\begin{aligned} p(V'_\mathrm{m}, T) &= p(V''_\mathrm{m}, T) \\ G'_\mathrm{m} &= G''_\mathrm{m} \end{aligned} \tag{1}$$

where p is the pressure, V_m is the molar volume, T is the temperature, and G_m is the molar Gibbs energy. This system of equations represents the equality of temperature, pressure, and molar Gibbs energy between phases. The last condition can be written as the residuum $r_G \equiv G''_\mathrm{m} - G'_\mathrm{m} = 0$ with

$$r_G = A''^\mathrm{r}_\mathrm{m} - A'^\mathrm{r}_\mathrm{m} - RT \ln\left(\frac{V''_\mathrm{m}}{V'_\mathrm{m}}\right) + p''V''_\mathrm{m} - p'V'_\mathrm{m}, \tag{2}$$

where $A^\mathrm{r}_\mathrm{m} = A_\mathrm{m}(T,\rho) - A^\mathrm{ig}_\mathrm{m}(T,\rho)$ denotes the residual molar Helmholtz energy (A^ig_m is the molar Helmholtz energy of the ideal gas),

$$A^\mathrm{r}_\mathrm{m}(V_\mathrm{m}, T) = -\int_\infty^{V_\mathrm{m}} \left(p(V_\mathrm{m}, T) - \frac{RT}{V_\mathrm{m}}\right) \mathrm{d}V_\mathrm{m} \,. \tag{3}$$

Similarly, the residuum for pressure reads

$$r_p = p(V'_\mathrm{m}, T) - p(V''_\mathrm{m}, T) \tag{4}$$

and phase equilibrium is achieved when r_p and r_G are "close enough" to zero. The trivial solution ($V'_\mathrm{m} = V''_\mathrm{m}$) represents a mathematically valid solution to the system of equations, but this solution is not generally meaningful.

2 Cubic Equations of State

The oldest cubic equations of state, the van der Waals equation[6,7] and the Redlich–Kwong equation,[8] have only two substance-specific parameters. Therefore they strictly obey the corresponding-states principle, and it is possible to write them with reduced temperatures and densities in a universal form that applies to all substances.

The second generation of cubic equations modifies the temperature dependence of the attractive term, using one or more substance-specific parameters. The most popular equations of this type are the Redlich–Kwong–Soave equation[9] and the Peng–Robinson equation,[10,11] which use Pitzer's "acentric factor" as the third parameter. Particularly for the Peng–Robinson equation, however, variants with more precisely tuned temperature dependences exist. In the context of this work it should be noted that all these equations of state can also be transformed into a universal form. In this case, however, the reducing parameter for the temperature is temperature-dependent.

A third class of cubic equations of state uses one or more additional substance-dependent parameters to modify the density-dependence of the attractive term. The oldest of these is perhaps the Fuller equation.[12] More recent equations are the "generalized cubic equations", for instance the Trebble–Bishnoi–Salim equation[13] or the GEOS equation by Geană and Feroiu.[14] Such equations can yield superior representations of the thermodynamic properties of fluids. With unlucky combinations of parameters, however, they are more prone to produce unphysical behavior than simpler cubic equations of state. This is perhaps the reason why such equations are not as popular for mixture calculations as the simpler equations. In the context of this work it is important to observe that cubic equations of this third class cannot be transformed into a universal form.

2.1 The van der Waals Equation

The equation for the pressure[6,7] is

$$p = \frac{RT}{V_\mathrm{m} - b} - \frac{a}{V_\mathrm{m}^2} \,. \tag{5}$$

with the reduced state variables

$$\tilde{\rho} = \frac{b}{V_\mathrm{m}} \quad \text{and} \quad \tilde{T} = \frac{RTb}{a} = \frac{8}{27}\frac{T}{T_\mathrm{crit}} \tag{6}$$

this equation becomes

$$\tilde{p} \equiv \frac{pb^2}{a} = \frac{\tilde{\rho}\tilde{T}}{1-\tilde{\rho}} - \tilde{\rho}^2 \,. \tag{7}$$

The corresponding expression for the residual Helmholtz energy is

$$\tilde{A}^\mathrm{r}_\mathrm{m} \equiv \frac{A^\mathrm{r}_\mathrm{m} b}{a} = -\tilde{T}\ln(1-\tilde{\rho}) - \tilde{\rho} \,. \tag{8}$$

At the critical point, the simultaneous solution of the conditions

$$\left(\frac{\partial\tilde{p}}{\partial\tilde{\rho}}\right)_{\tilde{T}} = 0 \tag{9}$$

$$\left(\frac{\partial^2\tilde{p}}{\partial\tilde{\rho}^2}\right)_{\tilde{T}} = 0 \tag{10}$$

yields the critical temperature and critical density of

$$\tilde{T}_\mathrm{crit} = \frac{8}{27} \tag{11}$$

$$\tilde{\rho}_\mathrm{crit} = \frac{1}{3} \tag{12}$$

after rejecting non-physical solutions at zero temperature. Substitution for pressure yields $\tilde{p}_\mathrm{crit} = 1/27$. The critical compressibility factor is therefore

$$Z_\mathrm{crit} = \frac{\tilde{p}_\mathrm{crit}}{\tilde{\rho}_\mathrm{crit}\tilde{T}_\mathrm{crit}} = \frac{3}{8} \tag{13}$$

For the three EOS considered here (although this general form does not apply to all cubic EOS), the constants $a = a_\mathrm{c}\alpha$ and b can be written in a generalized form of

$$a_\mathrm{c} = \Omega_a \frac{R^2 T^2_\mathrm{crit}}{p_\mathrm{crit}} \tag{14}$$

$$b = \Omega_b \frac{RT_\mathrm{crit}}{p_\mathrm{crit}} \tag{15}$$

with the general solutions

$$\begin{aligned} \Omega_a &= \frac{Z_c\tilde{\rho}_\mathrm{crit}}{\tilde{T}_\mathrm{crit}} \\ \Omega_b &= Z_\mathrm{crit}\tilde{\rho}_\mathrm{crit} \end{aligned} \tag{16}$$

For the vdW EOS, the values are

$$\Omega_a = \frac{27}{64} \tag{17}$$

$$\Omega_b = \frac{1}{8} \tag{18}$$

$$\alpha = 1 \tag{19}$$

The reduced pressure and the reduced residual Helmholtz energy therefore do not depend on substance-specific variables, and this is also true for the conditions of phase equilibrium. After multiplication with b/a, the phase equilibrium condition from Eq. (2) can therefore be written with reduced variables only,

$$\tilde{A}^{\mathrm{r}\prime\prime}_\mathrm{m} - \tilde{A}^{\mathrm{r}\prime}_\mathrm{m} + \tilde{T}\ln\left(\frac{\tilde{\rho}''}{\tilde{\rho}'}\right) + \tilde{p}''\left(\frac{1}{\tilde{\rho}''} - \frac{1}{\tilde{\rho}'}\right) = 0 \tag{20}$$

Hence it is possible to compute universal functions for the vapor pressure $\tilde{p}(\tilde{T})$ as well as the densities $\tilde{\rho}'(\tilde{T})$ and $\tilde{\rho}''(\tilde{T})$.

For given temperature and pressure, densities are obtained as solutions of the cubic polynomial

$$\tilde{\rho}^3 - \tilde{\rho}^2 + (\tilde{T} + \tilde{p})\tilde{\rho} - \tilde{p} = 0 \tag{21}$$

and the densities along the spinodal (where $(\partial\tilde{p}/\partial\tilde{\rho})_{\tilde{T}} = 0$) for a given reduced temperature are obtained as the solutions of the cubic polynomial

$$-2\tilde{\rho}^3 + 4\tilde{\rho}^2 - 2\tilde{\rho} + \tilde{T} = 0 \tag{22}$$

in $\tilde{\rho}$ for a given reduced temperature $\tilde{T}$; these solutions may be real or complex, and the real solutions may be outside the range of validity of (0, 1) for $\tilde{\rho}$.

2.2 The Redlich–Kwong–Soave Equation

Soave improved the Redlich–Kwong equation by introducing a more sophisticated tempera-

ture dependence of the attraction term,[9]

$$p = \frac{RT}{V_\mathrm{m} - b} - \frac{a_\mathrm{c}\alpha(T)}{V_\mathrm{m}(V_\mathrm{m} + b)}$$
$$\text{with} \quad \alpha(T) = \left[1 + (\sum_{k=0}^{2} m_k \omega^k)\left(1 - \sqrt{\frac{T}{T_\mathrm{crit}}}\right)\right]^2 , \tag{23}$$

The m_k are a set of (universal) numerical constants, and ω is Pitzer's acentric factor, a substance-specific parameter. In analogy to the previous section, the working equations for the generation of universal phase equilibrium functions are then

$$\tilde{\rho} = \frac{b}{V_\mathrm{m}} \quad \text{and} \quad \tilde{T} = \frac{RTb}{a_\mathrm{c}\alpha(T)} , \tag{24}$$

$$\tilde{p} \equiv \frac{pb^2}{a_\mathrm{c}\alpha(T)} = \frac{\tilde{\rho}\tilde{T}}{1-\tilde{\rho}} - \frac{\tilde{\rho}^2}{1+\tilde{\rho}} \tag{25}$$

and

$$\tilde{A}^\mathrm{r}_\mathrm{m} \equiv \frac{A^\mathrm{r}_\mathrm{m} b}{a_\mathrm{c}\alpha(T)} = -\tilde{T}\ln(1-\tilde{\rho}) - \ln(1+\tilde{\rho}) , \tag{26}$$

and again Eq. (20) can serve as the equilibrium condition.

Even though the RKS EOS introduces a third substance-specific parameter (ω), the equation of state can be written as a two-parameter equation. In order to compute the phase equilibrium for a given temperature T one has to determine $\alpha(T)$ first, set up the reduced variables according to Eq. (24), solve the universal equilibrium problem Eq. (20), and convert the solution to the normal pressure and density units.

In recent years there have been many updates to the α function employed, here we highlight the works of Neau and co-authors,[15,16] the work of Le Guennec et al.,[17,18] and our previous work.[19] The equations developed in this work do not depend on the α function employed.

Solution of the criticality conditions from Eq. (9) and Eq. (10) yields the density at the critical point as

$$\tilde{\rho}_\mathrm{crit} = -1 + \sqrt[3]{2} \tag{27}$$
$$= 0.25992104989487316477... \tag{28}$$

after rejecting complex solutions and non physical solutions at zero temperature. Python code for this is in the supporting information. Solving for the critical temperature in terms of the critical density via Eq. (9) yields

$$\tilde{T}_\mathrm{crit} = \frac{\tilde{\rho}_\mathrm{crit}\left(\tilde{\rho}_\mathrm{crit}-1\right)^2\left(\tilde{\rho}_\mathrm{crit}+2\right)}{\left(\tilde{\rho}_\mathrm{crit}+1\right)^2} \tag{29}$$
$$= -6\sqrt[3]{2} + 3 + 3\cdot 2^{\frac{2}{3}} \tag{30}$$
$$= 0.20267685653535943565\ldots \tag{31}$$

and the critical pressure is given by

$$\tilde{p}_\mathrm{crit} = \frac{-8 - 5\sqrt[3]{2} + 9\cdot 2^{\frac{2}{3}}}{-2 + \sqrt[3]{2}} \tag{32}$$
$$= 0.017559993780021070047\ldots \tag{33}$$

Substitution yields $Z_\mathrm{crit} = 1/3$. coefficients can be given (following Eq. (16), and after cancellation) by the values

$$\Omega_b = \frac{\sqrt[3]{2} - 1}{3} \tag{34}$$
$$= 0.086640349964957721589\ldots \tag{35}$$
$$\Omega_a = \frac{1}{9(\sqrt[3]{2} - 1)} \tag{36}$$
$$= 0.42748023354034140439\ldots \tag{37}$$

Density is obtained from the roots of the cubic polynomial

$$\tilde{\rho}^3 + (\tilde{T} + \tilde{p} - 1)\tilde{\rho}^2 + \tilde{T}\tilde{\rho} - \tilde{p} = 0 \tag{38}$$

and the spinodal densities are obtained from solutions of the quartic polynomial

$$\tilde{\rho}^4 + (-\tilde{T} - 3)\tilde{\rho}^2 + (2 - 2\tilde{T})\tilde{\rho} - \tilde{T} = 0 \tag{39}$$

2.3 The Peng–Robinson equation

This equation[10] has a slightly more complicated density dependence of the attractive term,

$$p = \frac{RT}{V_\mathrm{m} - b} - \frac{a_\mathrm{c}\alpha(T)}{V_\mathrm{m}(V_\mathrm{m}+b) + b(V_\mathrm{m}-b)} . \tag{40}$$

The $\alpha(T)$ function is formally the same as for the Redlich–Kwong–Soave function, but contains different numerical constants. A later ver-

sion[11] uses an additional m_k coefficient. The definitions of the reduced parameters are the same as in the previous section (Eq. (24)), and the same discussion about the range of alpha functions available also applies.

The working equations for the generation of universal phase equilibrium functions are

$$\tilde{p} \equiv \frac{pb^2}{a_c\alpha(T)} = \frac{\tilde{\rho}\tilde{T}}{1-\tilde{\rho}} - \frac{\tilde{\rho}^2}{1+2\tilde{\rho}-\tilde{\rho}^2} \tag{41}$$

and

$$\tilde{A}^r_m \equiv \frac{A^r_m b}{a_c\alpha(T)} = -\tilde{T}\ln(1-\tilde{\rho}) - 2^{-3/2}\ln\left(\frac{1+(\sqrt{2}+1)\tilde{\rho}}{1-(\sqrt{2}-1)\tilde{\rho}}\right) . \tag{42}$$

Solution of the criticality conditions from Eq. (9) and Eq. (10) yields the density at the critical point of

$$\tilde{\rho}_{crit} = -\frac{1}{3} - \frac{2^{\frac{2}{3}}}{3\sqrt[3]{4+3\sqrt{2}}} + \frac{\sqrt[3]{2}\sqrt[3]{4+3\sqrt{2}}}{3} \tag{43}$$

$$= 0.25307658654159946227... \tag{44}$$

after rejecting complex solutions and non physical solutions at zero temperature. Python code for this is in the supporting information. Solving for the critical temperature in terms of the critical density via Eq. (9) yields

$$\tilde{T}_{crit} = \frac{2\tilde{\rho}_{crit}\left(\tilde{\rho}_{crit}-1\right)^2\left(\tilde{\rho}_{crit}+1\right)}{\left(\tilde{\rho}^2_{crit}-2\tilde{\rho}_{crit}-1\right)^2} \tag{45}$$

$$= 0.17014442007035030247... \tag{46}$$

Following through the analytic solutions for critical density $\tilde{\rho}_{crit}$ and temperature $\tilde{T}_{crit}$ yields solutions for the critical pressure and critical compressibility factor of

$$\tilde{p}_{crit} = -\frac{\tilde{\rho}^2_{crit}\left(\tilde{\rho}^2_{crit}+2\tilde{\rho}_{crit}-1\right)}{\left(\tilde{\rho}^2_{crit}-2\tilde{\rho}_{crit}-1\right)^2} \tag{47}$$

$$= 0.013236567878127214416... \tag{48}$$

$$Z_{crit} = -\frac{\tilde{\rho}^2_{crit}+2\tilde{\rho}_{crit}-1}{2\left(\tilde{\rho}_{crit}-1\right)^2\left(\tilde{\rho}_{crit}+1\right)} \tag{49}$$

$$= 0.30740130869870384801... \tag{50}$$

The leading coefficients are given (following Eq. (16)) by

$$\Omega_b = -\frac{\tilde{\rho}_{crit}\left(\tilde{\rho}^2_{crit}+2\tilde{\rho}_{crit}-1\right)}{2\left(\tilde{\rho}_{crit}-1\right)^2\left(\tilde{\rho}_{crit}+1\right)} \tag{51}$$

$$= 0.077796073903888455972... \tag{52}$$

$$\Omega_a = -\frac{\left(\tilde{\rho}^2_{crit}-2\tilde{\rho}_{crit}-1\right)^2\left(\tilde{\rho}^2_{crit}+2\tilde{\rho}_{crit}-1\right)}{4\left(\tilde{\rho}_{crit}-1\right)^4\left(\tilde{\rho}_{crit}+1\right)^2} \tag{53}$$

$$= 0.45723552892138218938... \tag{54}$$

Density is obtained from the cubic polynomial

$$(\tilde{T}+\tilde{p}-1)\tilde{\rho}^3+(-2\tilde{T}-3\tilde{p}+1)\tilde{\rho}^2+(\tilde{p}-\tilde{T})\tilde{\rho}+\tilde{p}=0 \tag{55}$$

and the spinodal densities are obtained from solutions of

$$(\tilde{T}-2)\tilde{\rho}^4+(2-4\tilde{T})\tilde{\rho}^3+(2\tilde{T}+2)\tilde{\rho}^2+(4\tilde{T}-2)\tilde{\rho}+\tilde{T}=0 \tag{56}$$

2.4 Volume translation

Péneloux et al.[20] observed that adding a constant to the volume, e.g., for the Peng–Robinson equation

$$p = \frac{RT}{V_m+c-b} - \frac{a_c\alpha(T)}{(V_m+c)(V_m+c+b)+b(V_m+c-b)} , \tag{57}$$

does not change the vapor pressure, as the "translation" c cancels in the equilibrium condition Eq. (2). Consequently, the universal equilibrium functions for the Peng–Robinson equation can also be used for the "volume-translated Peng–Robinson equation" that underlies the

so-called VTPR group contribution model;[21,22] one merely has to add the volume translation at the end of the calculation. Volume translation of this kind is not without its challenges as some properties are unevenly shifted by the volume translation.[23,24]

3 Super-ancillary equations

3.1 Robust VLE algorithm

As we note below, there are many algorithms in use to solve the phase equilibrium problem for cubic equations of state. In this work we need an algorithm that is as robust as possible, efficiency being a secondary consideration. Therefore, this section lays out an algorithm that has been found to converge for every set of valid inputs, so long as sufficient numerical precision is available. The problem is constructed as a one-dimensional bounded root-finding problem, with a nested polynomial root-finding problem for the liquid density. Exceptionally reliable algorithms are available for each sub-problem.

The VLE algorithm is based upon first finding acceptable bounds for the vapor phase density, and iterating on the vapor phase density to find the suitable liquid phase density. The bounds on the solution for the vapor phase density are obtained from the following steps:

1. Find spinodal densities for the given temperature. These densities are obtained as the roots of a polynomial, and only real roots in the domain (0,1) are acceptable solutions.

2. The largest spinodal density in (0, $\tilde{\rho}_{\rm crit}$) is the vapor density upper bound

3. Lower bound for vapor density is the maximum of i) zero and ii) the vapor density associated with the pressure associated with the higher density spinodal solution

This approach is shown graphically in Fig. 1, the upper panel is for a relatively high temperature, while the lower panel is for a relatively low temperature. For the low reduced temperature, the vapor density paired with the pressure of the liquid spinodal solution is very negative, resulting in a negative (non-physical) lower bound for the vapor density, so a value just above zero (on the order of the denormal limit in the floating point mathematics) can be used as the lower bound. As the reduced temperature approaches 1.0, the numerical challenges in finding these limits increase, but in extended precision the solution for the limits is still possible arbitrarily close to the critical point (for $\widetilde{T}_{\rm crit} - \widetilde{T} \ll 10^{-16}$: the limit of double precision arithmetic)

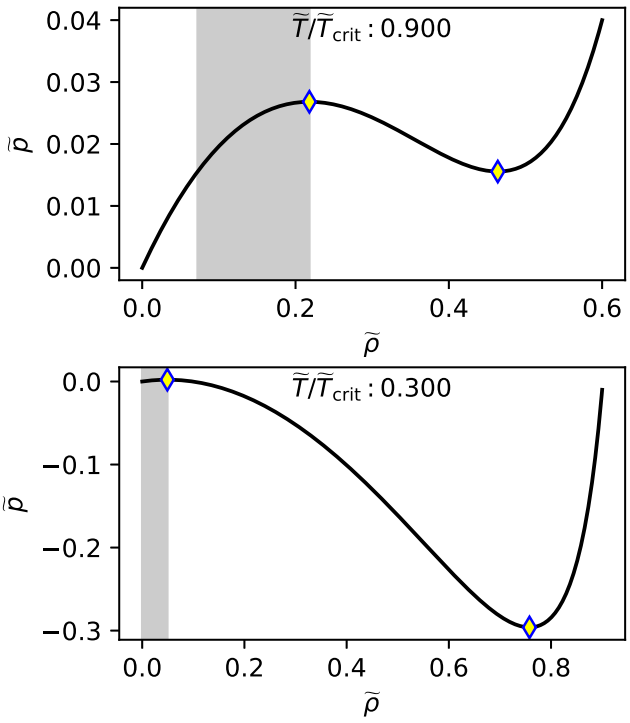

Figure 1: Schematic depiction of the bounds used for the one-dimensional solver for the vapor-liquid equilibrium at high (upper panel) and low (lower panel) reduced temperature. The vdW EOS is used here. The grey region depicts the range of densities to be searched for the equilibrium, and the diamonds are the spinodal densities in the region (0,1)

Solving the phase equilibrium problem turns into a bounded rootfinding problem. For a given value of vapor density, the pressure is calculated, and then the liquid density is obtained for the given pressure. The residuum for the

solver is the phase equilibrium residuum defined by Eq. (2).

Brent's method[25] was used for the outer solver and the C++ implementation of this method was taken from the CoolProp library[3] and further modified to allow for extended precision mathematics. Thus the VLE solution can be made arbitrarily precise depending on the numerical precision in use: the greater the numerical precision, the more accurately the phase equilibrium conditions may be solved, and the closer to the critical point the solution is able to converge.

The results of this robust approach for solving for the VLE conditions of the three cubic EOS are presented in Fig. 2. All three EOS show relatively similar qualitative behavior, and each can be represented in the generalized form described above in terms of $\tilde{T}$, $\tilde{p}$, and $\tilde{\rho}$, invoking no fluid-specific parameters.

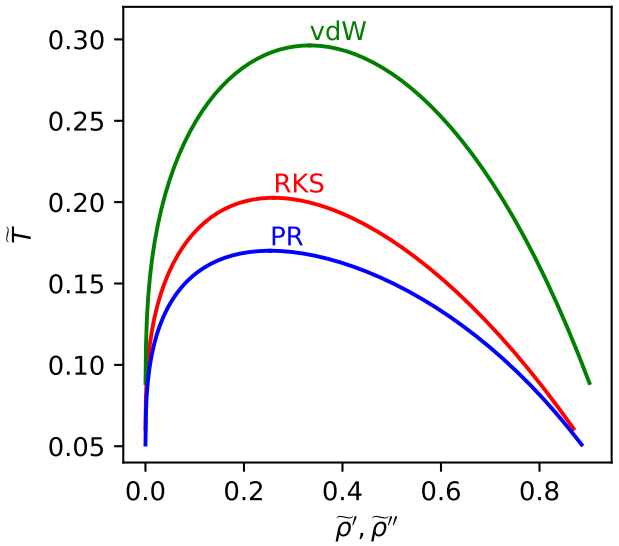

Figure 2: General solutions for phase equilibria for the three cubic EOS (vdW: van der Waals, RKS: Redlich-Kwong-Soave, PR: Peng-Robinson) considered in this work

An important detail that has been intentionally avoided in the above discussion is numerical precision. In the above calculations, more than 16 digits of precision have been used for all the calculations. The calculations in Python have utilized the *sympy* library with the *mpmath* extended precision library to obtain the values to any desired level of numerical precision. Constants have been given to more than 16 digits of precision, which is the practical limit of most computational routines.

When it comes to iteration and rootfinding, double precision does not allow for sufficient precision. Following the same approach as the previous paper,[5] the iterative calculations were carried out in extended precision. This is especially important for the polynomial rootfinding needed to obtain the spinodal solutions. The `boost::multiprecision` library was used for the extended precision calculations, as can be explored in the code in the supporting information. The addition of extended precision analysis to a templated C++ numerical library involves only a few compile-time type checks.

3.2 Construction of Chebyshev polynomials

Now that it is possible to obtain VLE solutions for the standard cubic EOS to any level of precision, it is necessary to make those results accessible to end users and software developers in a convenient and efficient formulation.

The use of orthogonal polynomials to approximate well-behaved and continuous numerical functions has a long pedigree. Recently, some mathematical functions have been implemented in terms of orthogonal polynomials in standard numerical libraries. The Chebyshev expansions have found extensive use as basis functions for approximation[26,27] and rootfinding.[26,28,29]

In essence: given the function values at carefully selected values of the independent variable (the Chebyshev-Lobatto nodes) in a bounded domain (usually [-1,1]), an excellent representation of the function of interest can be constructed. Evaluation of the approximation can be nearly as efficient as a monomial of the form $y = \sum_i c_i x^i$, though the numerical conditioning of Chebyshev expansions allows for high-order Chebyshev expansions at a level impossible to achieve with monomial expansions.

A C++ library (with Python wrappers) has been written by one of the authors[30] that allows for the evaluation and construction of Chebyshev expansions with a user-friendly interface. While construction of the expansions can be

done either via the Python interface or the low-level C++ code, in this case we have decided to use the C++ code for the improved computational efficiency. Doing so is not without its challenges, as ensuring that the versions of the libraries are mutually compatible is non-trivial. Here we have used version 1.6 of ChebTools[30] (which uses Eigen version 3.3.4) and boost version 1.71.0 for the `boost::multiprecision` library.

Rather than construct an extremely high-order Chebyshev expansion (degrees of 100 or 1000 pose no numerical challenge for a Chebyshev expansion), multiple Chebyshev expansions of lower order can be generated over domains that span the full domain of interest. The advantage of this approach is that the refinement can occur where it is needed, rather than uniformly over the entire domain. Evaluation of a single value from a set of low-order expansions can be much faster than a single large expansion; binary search to find the right low-order expansion plus its evaluation is significantly faster than evaluation of a high-order expansion with the same accuracy. To start, the entire domain is spanned by a single expansion, and the domain is then recursively subdivided as needed into smaller sub-domains. This process is fully automated in the `dyadic_splitting` function of ChebTools version 1.6.

The C++ code used to do the fitting is included in the supplemental information of this paper. A further Python script was used to parse the results and generate a C++ header file with no dependencies aside from the standard library of C++. Calling this file outside C++ can be achieved with a C++ to C interface, compiling the files into a shared library for convenient end use in higher-level languages. The scripts needed for that are in the `makeso` folder in the supporting information. The C++ interface exposes the function

```
double supercubic(int EOS, int prop, double Ttilde);
```

where `EOS` is an integer flag to select the equation of state, and `prop` a flag to select the property. A negative number is returned from these functions if the input is out of range. The returned value will always be in tilde-scaled form. The user is required to scale the inputs into tilde-scaled form prior to calling the functions and then unscale the outputs afterwards. Much of the scaling and unscaling work can be precalculated. $\tilde{T}$ is proportional to T, so scaling represents a single multiplication. The same is true for unscaling $\tilde{p}$ to p. Evaluation of α is still required for both the scaling and unscaling steps, which involves a few more floating point operations.

4 Results

4.1 Accuracy

The previous paper[5] provides a detailed discussion of the challenges in resolving the near-critical behavior of the EOS with Chebyshev expansions and discusses topics like rate of convergence of the expansions. Here we will simply focus on the results of the process, demonstrating that the set of Chebyshev expansions provides an excellent representation of the VLE obtained from the EOS.

The expansions were generated according to the following parameters in the `superancillary.cpp` file

- Degree of expansion is 18
- Temperature range of $[0.1\tilde{T}_{\rm crit}, \tilde{T}_{\rm crit}]$
- Tolerance for splitting is 10^{-13} based upon a 3-element norm

This resulted in roughly 24 expansions for density (and 6 for pressure) and per EOS, thus approximately 50 expansions per EOS. If each coefficient (19 per expansion) is an 8 byte double then the expansions represents on the order of 20,000 bytes of binary data for the set of three EOS.

In order to justify the claim that these formulations have made iterative evaluation of phase equilibria of pure fluids with the classic cubic EOS obsolete, calculations of the phase equilibria with 100 digits of precision (the extremely precise reference data) are compared with the results of the Chebyshev expansions. The errors were calculated at the midpoints between

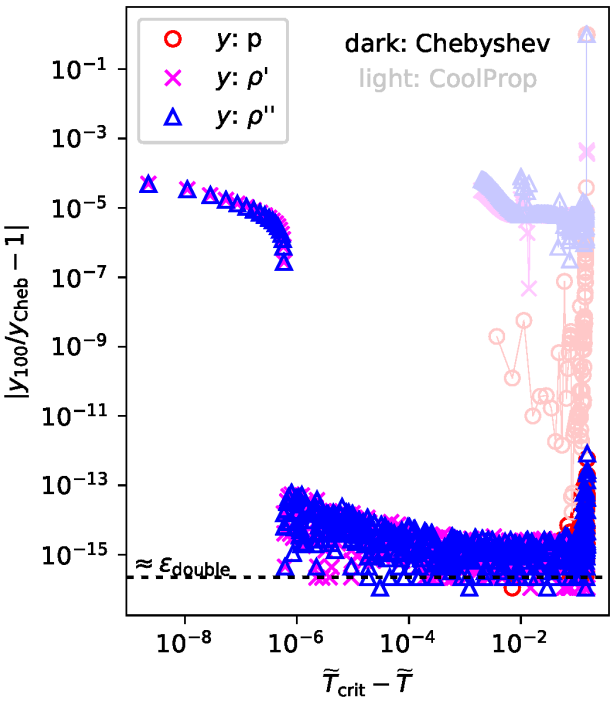

Figure 3: Absolute value of relative fractional errors between iterative calculations in 100 digits of precision y_{100} and the Chebyshev expansion at the midpoints between the nodes of the Chebyshev expansions y_{Cheb} for the Peng-Robinson EOS. For the CoolProp results (in a light color), the value of y_{Cheb} in the deviation is replaced with the result of the iterative calculation.

the Chebyshev-Lobatto nodes. These locations represent a good proxy for the worst-case errors of the approximation. The results of these comparisons are in Fig. 3. For all three properties, and all three cubic equations, the relative error from the Chebyshev expansion is on the order of numerical precision for $\widetilde{T} < \widetilde{T}_{\mathrm{crit}} - 10^{-6}$. The deviations do not quite reach the epsilon of approximately 2.2×10^{-16} in double precision, but this representation is nearly as good as could be expected in double precision. In any case, these deviations are much tighter than could be achieved for iterative calculations in double precision.

Also shown in the figure are the results from iterative calculations for the EOS from the CoolProp library, in this case a modified version with the sole change that the values of Ω_a and Ω_b are given to numerical precision rather than the truncated values.[31] The convergence criterion used in double precision results in densities that are in error by more than 0.1 % as compared with the extended precision calculations with 100 digits of precision as the critical point is approached. While these deviations are adequate for engineering purposes, they are approximately 10 orders of magnitude worse than what can be achieved with the Chebyshev expansions fit to the extended precision results. An equivalent result is obtained for the RKS and van der Waals EOS, as shown in the appendix.

4.2 Alternative critical scaling

As described in our previous article,[32] the orthobaric (saturated liquid and vapor) densities can be approximated by a power law in the vicinity of the critical point,

$$\tilde{\rho}^{\alpha} - \tilde{\rho}_{\mathrm{crit}} = \pm B_{\rho} \left(\frac{\tilde{T}_{\mathrm{crit}} - \tilde{T}}{\tilde{T}_{\mathrm{crit}}} \right)^{\beta}, \qquad (58)$$

where B_{ρ} is a so-called critical amplitude and $\alpha = ',''$. The critical exponent β is 0.5 for all analytical equations of state. Consequently, the

scaled density

$$\hat{\rho}^\alpha \equiv (\tilde{\rho}^\alpha - \tilde{\rho}_{\rm crit})\sqrt{1-\frac{\tilde{T}}{\tilde{T}_{\rm crit}}} \approx \pm B_\rho \left(1-\frac{\tilde{T}}{\tilde{T}_{\rm crit}}\right) \tag{59}$$

is linear w.r.t. temperature for $\tilde{T} \to \tilde{T}_{\rm crit}$ (see Fig. 4).

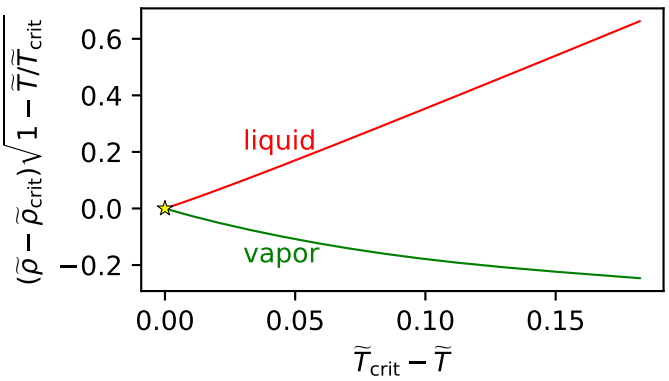

Figure 4: Orthobaric densities of the RKS EOS in the near critical region with the critical scaling from Eq. (59) applied, vs. the distance from the reduced critical temperature. The star indicates the location of the critical point in these reduced coordinates.

Hence $\hat{\rho}^\alpha(\tilde{T})$ no longer exhibits an infinite slope at the critical temperature and can be approximated with a smaller set of Chebyshev polynomials. This saves CPU time when searching for the desired expansion; on the other hand, the reverse transformation from $\hat{\rho}$ to $\tilde{\rho}$,

$$\tilde{\rho}^\alpha = \tilde{\rho}_{\rm crit} + \hat{\rho}^\alpha \left(1-\frac{\tilde{T}}{\tilde{T}_{\rm crit}}\right)^{-1/2} \tag{60}$$

is computationally more costly due to the need for the square-root function. Another disadvantage of Eq. (60) is that, at low temperatures, it yields the vapor density as the difference of two almost equal terms, and this leads to significant round-off errors that prevent the use of this scaling approach. A computationally safe way to represent vapor densities can be obtained with the transformation

$$\hat{\rho}^\alpha \equiv \left(\frac{\tilde{\rho}^\alpha \tilde{p}_{\rm crit}}{\tilde{p}} - \tilde{\rho}_{\rm crit}\right)\left(1-\frac{\tilde{T}}{\tilde{T}_{\rm crit}}\right)^{1/2} \tag{61}$$

The reverse transformation is

$$\tilde{\rho}^\alpha = \left(\frac{\hat{\rho}^\alpha}{\left(1-\frac{\tilde{T}}{\tilde{T}_{\rm crit}}\right)^{1/2}} + \tilde{\rho}_{\rm crit}\right)\frac{\tilde{p}}{\tilde{p}_{\rm crit}} \tag{62}$$

A downside of this improved approach is that the vapor pressure $\tilde{p}$ is needed for the evaluation of the scaled orthobaric densities, which doubles the evaluation time for the saturation density, although if the pressure is also needed (as is usually the case), the vapor pressure calculation is not duplicative.

The conclusion therefore is the same as for the multiparameter EOS:[5] creativity with alternative scalings that result in more linear functions is not obviously superior to the naïve approach of directly developing Chebyshev expansions of the unscaled property of interest.

4.3 CPU Time and Reliability

The computational speed of several implementations were tested against the Chebyshev expansions. The candidates were the iterative routines in REFPROP, the iterative routines in CoolProp, the iterative routines in *ThermoC*, and the robust iterative routines from the above, in double precision. In each case, argon was selected as the chemical species. Each implementation has a different convergence criterion on the phase equilibrium calculation, which makes direct comparison impossible. Nonetheless, the comparison gives an approximate picture of the relative computational advantage of this approach. All the calculations were carried out on a windows 10 machine, and the *ThermoC* calculations were carried out inside a docker container with Ubuntu 20.04 as its base image. The machine uses an Intel Core i7-8700 CPU @ 3.20GHz and can step up to 4.3 GHz under load.

The relative timing of the different approaches is laid out in Table 1. Random shuffling of the inputs is used to ensure that the processor branch prediction will be defeated as often as possible to obtain a fair comparison. The routines in CoolProp are the fastest of the calculations that use double precision arithmetic.

CoolProp carries out the vapor-liquid equilibrium by estimating the vapor pressure based upon the acentric factor of the fluid (see the SI of Ref. 19) and then iterating to find the equilibrium pressure; the approach here in extended precision is significantly more reliable (though much slower). For the Chebyshev expansions the input temperature (in Kelvin) is scaled, the calculation is carried out, and then the result is unscaled. The time breakdown for the Chebyshev expansions is very roughly: 25 nanoseconds to evaluate the expansion, 10 nanoseconds to find the right expansion, and the remainder is time to scale and unscale the input and output quantities and additional overhead. Timing code is included in the supporting information.

Table 1: Timing of calculations of saturated liquid density for 10^7 linearly spaced then shuffled temperatures in $[0.6T_{\rm crit}, 0.9T_{\rm crit}]$. The relative speedup of the Chebyshev expansions uses the Chebyshev expansions as baseline.

method	avg. μs/call	speedup
this work (Cheb)	0.053	
CoolProp	2.17	48.4×
REFPROP	5.01	112×
ThermoC	9.89	220×
this work (double)	215	4800×

Another relevant question is how close to the critical point the iterative calculations are able to succeed. The Chebyshev expansion approach cannot fail, only its accuracy decreases on approaching the critical point. On the other hand, the iterative calculations may fail below the critical point because the density solutions for the spinodal disappear for some temperature less than the critical temperature as a consequence of the limited numerical precision. The closest that the VLE calculations are able to succeed in double precision for each of the three EOS is approximately 10^{-3} below the critical temperature in tilde-scaled units.

A final question pertains to the very low temperature behavior of the Chebyshev expansions. As a consequence of the use of interval subdivision, the lowest temperature behavior can still be properly captured, even though the pressures and vapor densities are more than 10 orders of magnitude smaller than their respective values at the critical point. Figure 5 shows the values obtained for the Peng-Robinson EOS on a very fine grid over the range of the lowest temperature expansion, demonstrating that none of the predicted values go negative, and that the curve is smooth.

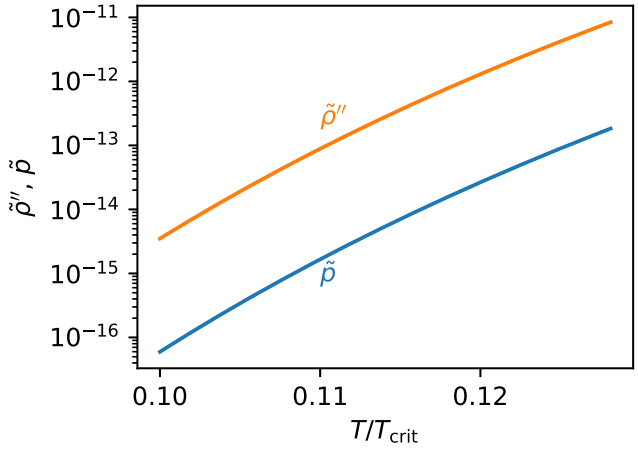

Figure 5: Values of saturated vapor density and pressure of the Peng-Robinson EOS calculated from the Chebyshev expansions at very low reduced temperatures.

5 Conclusions

The saturation phase boundary from cubic EOS for pure fluids can be very accurately represented by Chebyshev expansions. These expansions a) are at least an order of magnitude faster to evaluate than the iterative calculation of phase equilibrium, even in double precision arithmetic b) provide a representation of the phase equilibrium that is several orders of magnitude more precise than the iterative calculation c) are able to provide quite accurate values even very close to the critical point. Therefore, for these standard cubic EOS, there is no longer a need to carry out iterative calculations of phase equilibria for pure fluids. In order to also improve the accuracy of the cubic EOS themselves, we have calculated exact solutions for the coefficients of the cubic equations of state as opposed to the truncated values used so often in the literature.

The approach presented here is particularly

well suited to the "simple" cubic equations of state because they can be written in a common reduced form for all fluids. In principle any equation of state can be handled in the same way, with the commensurate simultaneous increase in computational efficiency and precision.

Supporting Information Available

In order to ensure reproducibility of the results, the supporting information available at https://doi.org/10.18434/mds2-2394 includes in a single zip file

- The C++ code used to build the expansions in the file `superancillary.cpp`
- The C++ header with the coefficients injected (no dependencies other than on the `vector` header of C++) in the file `makeso/cubicsuperancillary.cpp`
- The scripts used to inject the expansions into the header and test the header
- The jupyter notebook used to generate the constants

Readers are invited to contact the corresponding author for clarification of any aspect of the paper, or if any errors are identified.

A Deviations for other EOS

Figures 6 and 7 present the deviations for the van der Waals and RKS EOS, respectively.

B Validation

In order to ensure the correct implementation, the check values in Table 2 were obtained from the test code in the supporting information.

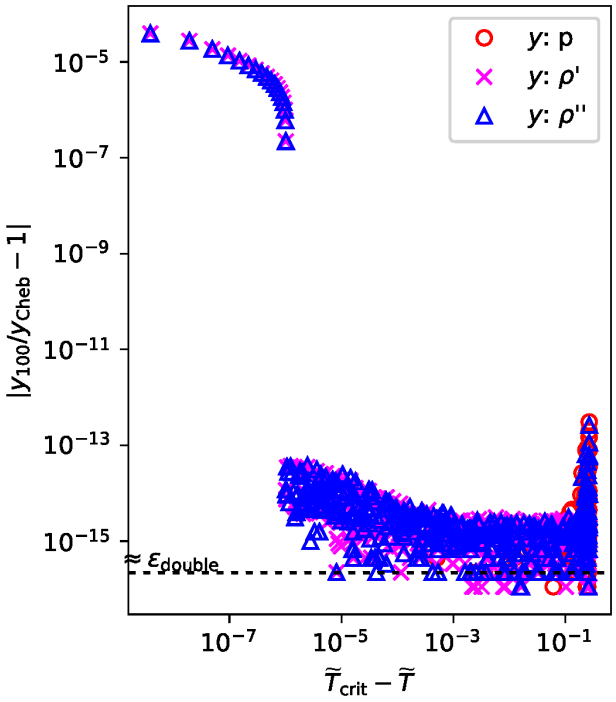

Figure 6: Absolute value of relative fractional errors between iterative calculations in 100 digits of precision y_{100} and the Chebyshev expansion at the midpoints between the nodes of the Chebyshev expansions y_{Cheb} for the van der Waals EOS.

Table 2: Validation values for each of the implementations at $\tilde{T} = 0.125$

EOS	$\tilde{p}$	$\tilde{\rho}'$	$\tilde{\rho}''$
vdW	0.0002958543239347111	0.8536251284168529	0.002407389267319304
PR	0.003034198868923775	0.6394564580846998	0.03023195086998487
RKS	0.001736846506201768	0.6976615743280177	0.01555500889873714

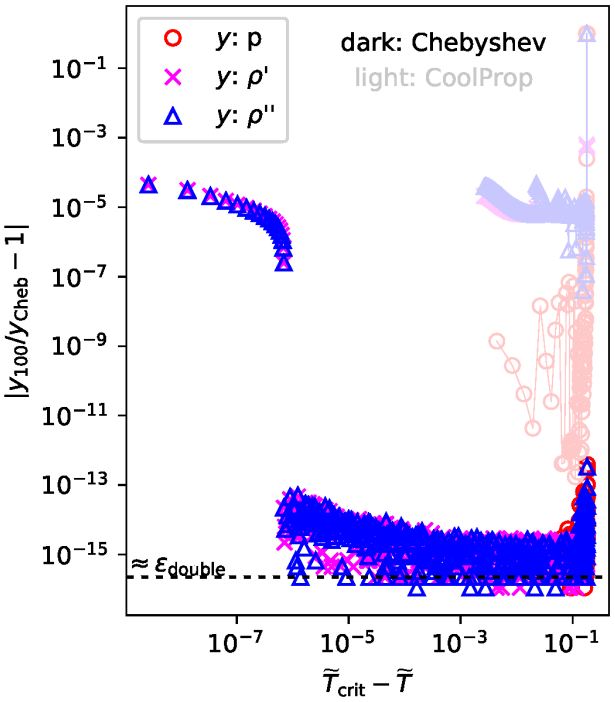

Figure 7: Absolute value of relative fractional errors between iterative calculations in 100 digits of precision y_{100} and the Chebyshev expansion at the midpoints between the nodes of the Chebyshev expansions y_{Cheb} for the Redlich-Kwong-Soave EOS. For the CoolProp results (in a light color), the value of y_{Cheb} in the deviation is replaced with the result of the iterative calculation.

References

(1) Berberan-Santos, M. N.; Bodunov, E. N.; Pogliani, L. The van der Waals equation: analytical and approximate solutions. *J. Math. Chem.* **2007**, *43*, 1437–1457, DOI: 10.1007/s10910-007-9272-4.

(2) Lemmon, E. W.; Bell, I. H.; Huber, M. L.; McLinden, M. O. NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties-REFPROP, Version 10.0, National Institute of Standards and Technology. http://www.nist.gov/srd/nist23.cfm, 2018.

(3) Bell, I. H.; Wronski, J.; Quoilin, S.; Lemort, V. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. *Ind. Eng. Chem. Res.* **2014**, *53*, 2498–2508, DOI: 10.1021/ie4033999.

(4) Span, R.; Beckmüller, R.; Hielscher, S.; Jäger, A.; Mickoleit, E.; Neumann, T.; Pohl, S.; Semrau, B.; Thol, M. TREND. Thermodynamic Reference and Engineering Data 5.0. 2020; http://www.thermo.ruhr-uni-bochum.de/.

(5) Bell, I. H.; Alpert, B. K. Efficient and Precise Representation of Pure Fluid Phase Equilibria with Chebyshev Expansions. *Int. J. Thermophys.* **2021**, *42*, 75, DOI: 10.1007/s10765-021-02824-x.

(6) van der Waals, J. D. Over de continuiteit van den gas- en vloeistoftoestand. Ph.D. thesis, Universiteit Leiden, The Netherlands, 1873.

(7) van der Waals, J. D. *On the Continuity of the Gaseous and Liquid States (with an introduction by J. S. Rowlinson)*; Studies in Statistical Mechanics; North-Holland, Amsterdam, 1988; Vol. XIV.

(8) Redlich, O.; Kwong, J. N. S. On the thermodynamics of solutions. V. An equation of state—fugacities of gaseous solutions. *Chem. Reviews* **1949**, *44*, 233–244.

(9) Soave, G. Equilibrium constants from a modified Redlich–Kwong equation of state. *Chem. Eng. Sci.* **1972**, *27*, 1197–1203.

(10) Peng, D. Y.; Robinson, D. B. A new two-constant equation of state. *Ind. Eng. Chem. Fundam.* **1976**, *15*, 59–64, DOI: 10.1021/i160057a011.

(11) Robinson, D. B.; Peng, D. Y. The characterization of the heptanes and heavier fractions for the GPA Peng–Robinson program. *GPA Research Report* **1978**, *RR-28*, 1–36.

(12) Fuller, G. G. Modified Redlich–Kwong–Soave equation of state capable of representing the liquid state. *Ind. Eng. Chem. Fundam.* **1976**, *15*, 254–257, DOI: 10.1021/i160060a005.

(13) Trebble, M. A.; Bishnoi, P. R. Development of a new four-parameter cubic equation of state. *Fluid Phase Equilib.* **1987**, *35*, 1–18, DOI: 10.1016/0378-3812(87)80001-8.

(14) Geană, D.; Feroiu, V. *Ecuaţii de stare*; Editura Tehnică, Bucarest, 2000.

(15) Neau, E.; Hernández-Garduza, O.; Escandell, J.; Nicolas, C.; Raspo, I. The Soave, Twu and Boston–Mathias alpha functions in cubic equations of state. Part I. Theoretical analysis of their variations according to temperature. *Fluid Phase Equilib.* **2009**, *276*, 87–93, DOI: 10.1016/j.fluid.2008.09.023.

(16) Neau, E.; Raspo, I.; Escandell, J.; Nicolas, C.; Hernández-Garduza, O. The Soave, Twu and Boston–Mathias alpha functions in cubic equations of state. Part II. Modeling of thermodynamic properties of pure compounds. *Fluid Phase Equilib.* **2009**, *276*, 156–164, DOI: 10.1016/j.fluid.2008.10.010.

(17) Le Guennec, Y.; Lasala, S.; Privat, R.; Jaubert, J. N. A consistency test for alpha-functions of cubic equations of state. *Fluid Phase Equilib.* **2016**, *427*, 513–538, DOI: 10.1016/j.fluid.2016.07.026.

(18) Le Guennec, Y.; Privat, R.; Jaubert, J.-N. Development of the translated-consistent tc-PR and tc-RK cubic equations of state for a safe and accurate prediction of volumetric, energetic and saturation properties of pure compounds in the sub- and super-critical domains. *Fluid Phase Equilib.* **2016**, *429*, 301–312, DOI: 10.1016/j.fluid.2016.09.003.

(19) Bell, I. H.; Satyro, M.; Lemmon, E. W. Consistent Twu Parameters for More than 2500 Pure Fluids from Critically Evaluated Experimental Data. *J. Chem. Eng. Data* **2018**, *63*, 2402–2409, DOI: 10.1021/acs.jced.7b00967.

(20) Péneloux, A.; Rauzy, E.; Fréze, R. A consistent correction for Redlich–Kwong–Soave volumes. *Fluid Phase Equilib.* **1982**, *8*, 7–23, DOI: https://doi.org/10.1016/0378-3812(82)80002-2.

(21) Ahlers, J.; Gmehling, J. Development of an universal group contribution equation of state. I: Prediction of liquid densities for pure compounds with a volume translated Peng–Robinson equation of state. *Fluid Phase Equilib.* **2001**, *191*, 177–188, DOI: 10.1016/S0378-3812(01)00626-4.

(22) Schmid, B.; Gmehling, J. Revised parameters and typical results of the VTPR group contribution equation of state. *Fluid Phase Equilib.* **2006**, *317*, 110–126, DOI: 10.1016/j.fluid.2012.01.006.

(23) Jaubert, J.-N.; Privat, R.; Guennec, Y. L.; Coniglio, L. Note on the properties altered by application of a Péneloux–type volume translation to an equation of state. *Fluid Phase Equilib.* **2016**, *419*, 88–95, DOI: 10.1016/j.fluid.2016.03.012.

(24) Privat, R.; Jaubert, J.-N.; Guennec, Y. L. Incorporation of a volume translation in an equation of state for fluid mixtures: which combining rule? which effect on properties of mixing? *Fluid Phase Equilib.* **2016**, *427*, 414–420, DOI: 10.1016/j.fluid.2016.07.035.

(25) Brent, R. *Algorithms for Minimization without Derivatives*; Prentice-Hall, Englewood Cliffs, NJ, 1973; Chapter 4.

(26) Trefethen, L. N. *Approximation Theory and Approximation Practice, Extended Edition*; SIAM, Philadelphia, USA, 2020.

(27) Driscoll, T. A.; Hale, N.; Trefethen, L. N. Chebfun Guide. 2014.

(28) Bell, I. H.; Alpert, B. K. Exceptionally reliable density-solving algorithms for multiparameter mixture models from Chebyshev expansion rootfinding. *Fluid Phase Equilib.* **2018**, *476B*, 89–102, DOI: 10.1016/j.fluid.2018.04.026.

(29) Boyd, J. P. Finding the Zeros of a Univariate Equation: Proxy Rootfinders, Chebyshev Interpolation, and the Companion Matrix. *SIAM Rev.* **2013**, *55*, 375–396, DOI: 10.1137/110838297.

(30) Bell, I. H.; Alpert, B. K.; Bouck, L. ChebTools: C++11 (and Python) tools for working with Chebyshev expansions. *J. Open Source Soft.* **2018**, *3*, 569, DOI: 10.21105/joss.00569.

(31) Bell, I. H.; Jäger, A. Helmholtz Energy Transformations of Common Cubic Equations of State for Use with Pure Fluids and Mixtures. *J. Res. Nat. Inst. Stand. Technol.* **2016**, *121*, 238, DOI: 10.6028/jres.121.011.

(32) Deiters, U. K.; Bell, I. H. Unphysical Critical Curves of Binary Mixtures Predicted with GERG Models. *Int. J. Thermophys.* **2020**, *41*, DOI: 10.1007/s10765-020-02743-3.

TOC Graphic