The Thermodynamic Models in PYroMat

Christopher R. Martin

March 2, 2021

1 Introduction

As of PYroMatversion 2.1.0, there are three general model formulations used to evaluate the thermodynamic properties of gases and multiphase (liquid-vapor) substances. Each pure substance is referred to as a "species." For ideal gases, calculating the properties of mixtures is almost trivial, but when inter-molecular forces rise to relevance, this task becomes difficult enough to require special treatment with a unique mixture model.

1.1 Ideal Gases

In ideal gas models, inter-molecular forces are unimportant, so only the molecule's variable specific heat needs to be resolved in order to construct an entire thermodynamic model. There are two classes in PYroMatfor the two commonly used standards: the IG1 class manages the Shomate equation of state, and IG2 manages the so-called NASA polynomials equation of state. In either case, constant-pressure specific heat, c_p, is constructed purely as a function of temperature. These are addressed in the first two sections below.

Given an explicit analytical formulation for $c_p(T)$, it is practical to express enthalpy

$$\begin{aligned} h(T) &= \int_{T_0}^{T} c_p(\tau)\mathrm{d}\tau \\ &= \int c_p(T)\mathrm{d}T + h_0. \end{aligned} \tag{1}$$

There are many ways one could decide on a value for the reference temperature, T_0, and the value of enthalpy there, h_0, would seem to be arbitrary. That is not so, however, when one is dealing with chemical reactions. These values must account for the energy released or absorbed in chemical reactions. As a result, T_0 and h_0, must be explicitly provided with the model, and the models must be constructed with the same conventions.

Entropy is calculated from its definition with the benefit of the ideal gas equation of state,

$$\begin{aligned} \mathrm{d}s &= \frac{\mathrm{d}h}{T} - \frac{v\mathrm{d}p}{T} \\ s(T,p) &= s_0 + \int \frac{c_p(T)}{T}\mathrm{d}T - R\ln\left(\frac{p}{p_0}\right) \end{aligned} \tag{2}$$

1.2 Multi-phase Properties

At states when inter-molecular forces are important, a far more general formulation is needed. For engineering applications, it is popular to refer to the state of a substance in terms of temperature and pressure, but this is entirely inadequate when phase transitions make sharp discontinuities possible. This is especially true when meta-stable states make it possible for multiple properties to be defined at the same T,p pair. Formulations with respect to (T, p) pairs are not practical in general.

2 IG1: The Shomate Equation

PYroMat's ig1 class is built on the Shomate equation, which is the basis of NIST's gas phase thermophysical property database.

$$t = \frac{T}{1000\mathrm{K}} c_p(t) \quad = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + \frac{c_4}{t} \tag{3}$$

It may be obvious from inspecting the equation that there is little attempt to formulate the

3 Polynomials of two variables

In the evaluaiton of empirical thermodynamic properties, there is often the problem of evaluating polynomial expansions on two variables. These expansions are typically of the form

$$p(x, y) = \sum_{i,j} c_{i,j} x^i y^j \tag{4}$$

where a and b are real coefficients such that i and j are integer indices.

3.1 Modifying polynomials for non-integer and negative powers

Fractional and negative exponents are also possible within this framework if we were to accept input values X, and Y, and adjust them according to pre-exponentials a and b,

$$x = X^a \tag{5}$$

$$y = Y^b. \tag{6}$$

The new polynomial formed is

$$p(X, Y) = \sum_{i,j} c_{i,j} X^{ai} Y^{bj}. \tag{7}$$

For example, were $a = 0.25$, then the expansion on X could proceed in fractions of 4 even though the evaluation algorithm we use is purely expressed in integer exponents. It is relatively computationally inexpensive to apply these exponentials prior to the code's execution. It would be far more costly to apply them to each term.

Similarly, negative exponentials can be achieved by using post-exponential terms

$$P(X, Y) = X^{\alpha} Y^{\beta} p(X, Y). \tag{8}$$

When α or β are non-zero, the effective exponents are all shifted positively or negatively by a single multiplication operation.

3.2 Efficient evaluation of the polynomial

The PYroMat polynomial evaluation algorithm is an expansion with purely integer exponents.

$$p(x, y) = \sum_{i,j} c_{i,j} x^i y^j \tag{9}$$

However, evaluating each term individually requires two expensive calls to a `pow` function and two floating point multiplications.

The widely accepted method for evaluating a polynomial of one variable is to construct a recursive expansion

$$q(y) = c_0 + y(c_1 + y(c_2 + y(\ldots \tag{10}$$

If there are n coefficients, then this amounts to only m multiplications with no `pow` calls. In order to extend this algorithm to two variables, more elegant notation will be helpful. If we name the intermediate value calculated in the process of these recursions q, then a polynomial with n terms implies the series

$$q_n = c_n \tag{11}$$
$$q_j(y) = c_j + y\, q_{j+1}(y) \tag{12}$$
$$q_0(y) = q(y). \tag{13}$$

This is a series beginning with q_n, and proceeding backwards through the values of j to q_0, which is the final value for $q(y)$. In practice, there is no need to keep the old values of q, so a single register may be used to hold the latest value.

How can this be extended to a polynomial of two variables? We may consider the polynomials to be nested; the evaluation of a polynomial on Y determines the individual coefficients for a polynomial on X.

$$p(x,y) = \sum_i q_i(y) x^i \tag{14}$$
$$q_i(y) = \sum_j c_{i,j} y^j \tag{15}$$

We only need a minor modification to the intermediate values for the x polynomial since there will be a separate expansion for each value of i. If there are n j terms,

$$q_{i,n}(y) = c_{n,j} \tag{16a}$$
$$q_{i,j}(y) = c_{i,j} + y\, q_{i+1,j}(y) \tag{16b}$$
$$q_{i,0}(y) = q_i(y). \tag{16c}$$

If there are m x terms,

$$p_m(x) = q_m(x) \tag{17a}$$
$$p_i(x,y) = q_i(x) + y\, p_{i+1}(x,y) \tag{17b}$$
$$p_0(x,y) = p(x,y). \tag{17c}$$

3.3 Efficient evaluation of derivatives

The partial derivatives of the polynomial can be efficiently evaluated along with the polynomial itself. To relax the already cumbersome notation, the functional dependencies (y) and (x,y) will be dropped. For the purpose of thermodynamic property evaluation, the first two derivatives will suffice.

Let us begin with the simpler task of calculating the derivatives of q_j.

$$q_{i,n|y} = 0 \tag{18a}$$
$$q_{i,j|y} = q_{i+1,j} + y\, q_{i+1,j|y} \tag{18b}$$
$$q_{i,0|y} = q_{i|y} \tag{18c}$$

$$q_{i,n|yy} = 0 \tag{19a}$$
$$q_{i,j|yy} = 2q_{i,j+1|y} + y\, q_{i,j+1|yy} \tag{19b}$$
$$q_{i,0|yy} = q_{i|yy} \tag{19c}$$

The derivatives on p are constructed somewhat differently because they can be in both x and y. Beginning with y,

$$p_{n|y} = 0 \tag{20a}$$
$$p_{j|y} = q_{i|y} + x\, p_{j+1|y} \tag{20b}$$
$$p_{0|y} = p_y \tag{20c}$$

$$p_{m|yy} = 0 \tag{21a}$$
$$p_{i|yy} = q_{i|yy} + x\, q_{i+1|yy} \tag{21b}$$
$$p_{0|yy} = p_{yy} \tag{21c}$$