

ME 370B
Energy Systems II:
Modeling and Advanced Concepts

Chris Edwards
Winter 2023

Project #7: Helmholtz Mixture Props: Bubble, Dew, & Flash

Individual & Group assignment, due as a PDF via Canvas, Thursday, Mar. 2, 11 PM

There isn't much more we need to say beyond that which is in the *More than you ever wanted to know...* handout. If you haven't read that already, please do so before starting in here.

In this project we will use the state-of-the-art Lemmon-Jacobsen O₂-N₂-Ar Helmholtz mixture model to build a basic understanding of how to handle mixtures of real fluids such as air. In order to get you up and running, we have posted a get-started script *Pr_Isotherms_c* which will generate a basic P - ρ isotherm plot in the vicinity of the vapor dome for air as a ternary mixture. Note that the location of the spinodal points has been included as a series of black dots on this plot. The reason is for clarity—if you connect these together to draw out the spinodal lines you will see that these shoot off in wild directions as inflections occur in the fundamental relation surface for the mixture. The culprit is the fundamental relation for nitrogen (Span et al., 2000). This is the state-of-the-art surface for nitrogen, and therefore the standard for mixtures involving nitrogen. The fact that it has terribly misbehaved spinodals is something that we simply have to deal with—the priority in making such fits is the accuracy of the resulting surface, not the convenience of those programming it. We will just have to cope with the resulting numerical gymnastics as we go along.

As you did last week, use the *Pr_Isotherms_c* script as a way to get started writing your own scripts as well as a way to get a look at the state surface. In particular, look for where the spinodals (or lack thereof) are likely to cause problems for numerics. Remember that the fundamental relation we will be using is valid for any mixture of the three component gases. As such, if you change the composition specification in the file from air to any other mixture and run the file, you will be looking at a projection of the P - ρ - T surface for that mixture, whatever it may be, including the limiting cases of each pure component. Run this script to make sure you have things functioning, and use it to view the isotherms for binary (engineering) air, ternary air, and other N₂-O₂-Ar mixtures (by varying c).

The Objectives

- (1) to explore how state-of-the-art mixture properties are implemented in software using fundamental relations in the Helmholtz representation and the Lemmon-Jacobsen approach,
- (2) to further your experience of what it takes to develop tools of the kind contained in Cantera,
- (3) to develop a deeper understanding of the role of chemical potential in phase equilibrium,
- (4) to position you for the discussion of air distillation that is yet to come, and
- (5) to see why less-secure, fast numerical methods may be needed to provide solutions in situations where simpler, more-secure methods cannot be implemented.

The Assignment

This assignment has three parts. The first is individual work but hopefully with lots of collaboration. The second and third parts are group work, so divide and conquer as you see fit.

- (1) 40 Points Individual work and plots required. Write a pair of scripts that use the (Gibbs) ideal solution model for the complementary (unknown) phase to generate approximate answers for the bubble and dew points of a specified mixture at a specified temperature. Since the specified-phase composition and temperature are known (e.g., air as a vapor in a dew-point calculation) the chemical potential of each of its components is known once a candidate pressure (the iterating variable) is specified (see *mui_icrT* and the new *rl_cTP* and *rv_cTP* functions). Similarly, the chemical potential of each *pure component fluid* can be obtained at this temperature and pressure (see *mu_irT*, *rl_iTP*, *rv_iTP*). Consistent with the ideal solution model, use the Amagat model (partial volumes) to calculate the density of the complementary phase.

By definition, the difference between the chemical potential of a pure component and its chemical potential in an ideal solution is $\bar{R}T \ln x_i$. As such, if the actual chemical potential and the pure-component chemical potential of a species are both known, the mole fraction required in the ideal-solution phase (to balance the chemical potentials) can be found. If the sum of these mole fractions is not unity, the pressure is incorrect. Raising or lowering the pressure in your iterative solution procedure then provides a means to preferentially raise or lower the chemical potential in the unknown phase with respect to the specified phase. Stated another way, by adjusting the pressure, the chemical potentials can be shifted until the mole fractions of the complementary phase sum to unity and you have a physically viable solution. (Other methods of closing the pressure iteration can also be devised and are just fine. This is simply the most common.)

Exercise your functions by plotting the dew and bubble points they return over the range of temperatures where it is possible to construct an ideal solution. (Determining the conditions under which this is possible is part of what you are figuring out.) Specifically, make three plots: temperature vs. density, pressure vs. density, and composition of the complementary phase vs. complementary-phase density for your ideal bubble and dew lines (and their complementary phase). Although your functions should be capable of dealing with any binary or ternary composition, use ternary air as defined in the *Pr_Isotherms_c* script as your working fluid.

Test your ideal functions by using them to provide a starting guess for the *Dew_cT* and *Bubble_cT* routines. If your functions are working correctly, the solutions they produce should be close enough to allow these routines to converge on the exact (as per the mixture fundamental relation) values. Add curves for the exact solution to each of your plots so that you can compare the results from the ideal-solution model to those from the fundamental relation.

On each plot show the dew line as the left side of the dome and the bubble line as the right side. (Since these are “variable-x” vs. density plots, the left side is vapor and the right is liquid.) Use an alternative color or line type to show the complementary phase on the same plot. Add the locations of the critical point, maxcondenbar, and maxcondentherm of air to each plot. Use the supplied procedure *Pr_Inflection_c* to find the numerical *P-ρ* inflection point for air and add it to the plot too.

- (2a) 20 Points We would like to use your ideal functions to provide starting points for *Dew_cT* and *Bubble_cT* over the full range of conditions up to near-critical temperature. This can be done by using a combination of direct ideal-solution starting points in the range where they are possible, and extrapolation of ideal-solution values into the high-temperature range. Two functions have been supplied for this: *Dew_Start_cT* and *Bubble_Start_cT*.

In order to use these functions, you will need to take one of your group's functions, name it *Ideal_Dew_cT* or *Ideal_Bubble_cT*, and modify it as required so as to supply the return values in the same order as in the *Dew_cT* and *Bubble_cT* functions. You will also need to have it return a saturation pressure value of zero under conditions where formation of an ideal solution is not possible. (The *Dew_Start_cT* and *Bubble_Start_cT* functions use this as a flag to determine whether to use the ideal-solution values or whether to construct an extrapolation.) Confirm that the functions are working correctly by checking to see that the *Dew_cT* and *Bubble_cT* functions are now capable of starting autonomously (by calling the start-up functions internally) at any temperature up to the near-critical region (~98% of critical temperature). If you are not familiar with the variable argument input operations of Matlab, now would be a good time to look into these (type *help varargin*).

Make what I might call a *Vapor_Dome_c* function that returns the T - ρ locus of the vapor dome for any composition (including the pure components). It should be based on solutions that are exact up to the near-critical region on both sides of the dome, but then splines over the top between the vapor and dew lines. This can be accomplished by using Matlab's *interp1* function to generate a spline through data that you generate using the *Bubble_cT* and *Dew_cT* functions.

Have your function return a pair of matched arrays (T and ρ) that span the complete vapor dome from the lowest temperature (use 60 K) on the dew curve (and therefore lowest density) to the lowest temperature (again 60 K) on the bubble curve (and therefore the highest density). Use a sufficient number of *Bubble_cT* and *Dew_cT* solutions on each side of the dome to guarantee a well-behaved fit across the entire region (you will eventually want 10-15 solutions on each side to avoid spurious oscillations, but start with a smaller number to speed debugging along). Do not include the P - ρ inflection point in the data used to generate the spline—the region at the top of the dome passes above this point—it is *not* the critical point for a mixture. Write your routine in such a way that it can treat arbitrary binary or ternary mixtures of oxygen, nitrogen, and argon. For the special case where a ternary mixture of air is considered, include the (known) critical-point as part of the data used for fitting. (In the general case, the critical point is not known a priori, and therefore it cannot be included.) How close does the spline fit come to passing through the maxcondentherm and maxcondenbar for air? (No response needed.)

Use your function to generate the same three plots as in Part 1, but now comparing the actual solutions used to generate the spline (as discrete points) with the spline curves across the top of the dome. Do this for ternary air and show the critical point, maxcondentherm, maxcondenbar, and P - ρ inflection points on the plots as well.

Write the T - ρ vapor dome arrays to disk using the Matlab *save* function. We will need them for Part 2b below but do not wish to re-compute them.

- (2b) 10 Points Write a script that loads your vapor dome arrays from disk and uses them to make other versions of the vapor dome. Since the arrays give the dome in Helmholtz variables, they can be used to calculate additional arrays for any property we wish. Use this fact to generate and plot vapor domes for the following diagrams: T - s , h - s , T - h , and P - h (semilogy). Please include the maxcondentherm, maxcondenbar, critical, and P - ρ inflection points on each plot. (Since these plots are relatively simple, you might consider putting them in a compact 2x2 array using the *subplot* command.)

- (3a) 15 Points Phase-change processes for multicomponent mixtures invariably involve changes in composition. In order to visualize these, plotting in a three-dimensional space can be very helpful (at least for binary mixtures). In order to get you started, we have provided a function *Tsx_Plot3* that will create a Matlab *plot3* representation of the T - s - x surface of the binary oxygen/nitrogen system (x is the mole fraction of nitrogen). Run this script to see how this space looks, and modify it a bit (for example, by removing unneeded isobars for clarity) to see how you might be able to use it for visualization. Use the viewpoint control in the Matlab figure window to rotate it around to see various views of the space: general 3-D, T - s in 2-D, T - x in 2-D, etc. (Note that this file loads the data that comprise the space from disk. If you do not have the *Tsx_Data.mat* file in the directory, the script will not run.)

Use the supplied *Flash_zTP* function to show what happens during isobaric cooling of engineering air from 300 K, 1 bar to 60 K. Include composition tie lines (lines linking complementary compositions) at a few (at least four) temperature intervals under the dome. Show the complete cooling process from 300 to 60 K as a continuous, smooth process path in the T - s - x space created above. Since this process involves only the 1 bar isobar surface, eliminate all of the other isobars from the plot. (Hint: Don't ask for flash calculations right at the bubble and dew boundaries. Although it is possible to check for these in *Flash_zTP*, it really slows the function down. Use the bubble and dew functions instead.)

- (3b) 15 Points Use the *Flash_zTP* function as a starting point from which to build a *Flash_zhP* function. Do this by implementing an outer Newton-Raphson iteration loop around the *Flash_zTP* function. In order to do this you will want to study how the *Flash_zTP* function finds its solution and use that as a basis to choose an appropriate approach. Note that the *Flash_zTP* function is capable of generating its own starting point by using bubble and dew calculations from which to interpolate starting estimates. Your function should also be capable of supplying its own starting values.

Demonstrate your function by using it to show the state points for a Linde dual-pressure air liquefaction system on a T - s - x diagram: Start with air at ambient conditions, compress it isothermally to 150 bar and then cool it at constant pressure to 150 K. Flash the fluid (via throttle) from this state into a receiver (tank) at 5 bar. Return the vapor from that receiver to ambient temperature by isobaric heat transfer, and flash (throttle) the liquid to another receiver maintained at a pressure of 1 bar. (The liquid that results from the second flash is the product and is sometimes referred to as *oxygen-enriched liquid air*.) Return the vapor from that flash to ambient temperature by isobaric heat transfer. Show only the state points described above, connected by straight, dashed lines (smooth process paths are not needed for this illustration).

The Write-Up

Only the plots are required. As always, please provide a brief statement of how much time was spent on each part of the analysis and what you thought were the most/least useful aspects. Venting about frustrations encountered in working through this are not only acceptable, they are highly desirable since they let us know issues that we might not otherwise discover.

The Deliverables

- (1) Individual plots required.

T - ρ plot for ternary air with ideal solution complementary phase over the range possible and for a real solution over the same range. Show the dew line as the left side of the dome and the bubble line as the right. Use an alternative color or line type to show the complementary phase on the same plot. Add the critical point, maxcondenbar, and maxcondentherm, and P - ρ inflection point.

P - ρ plot with same conditions.

Mole fraction- ρ plot with same conditions (but no critical point, etc.).

- (2) Same three plots as above, but comparing the actual solutions used to generate the spline (as discrete points) with the spline curves across the top of the dome.

T - s , h - s , T - h , and P - h (*semilogy*) plots of the spline fit to the vapor dome. Please include the maxcondentherm, maxcondenbar, critical, and P - ρ inflection points.

- (3) T - s - x 3-D plot showing isobaric cooling of binary air from 300 K to 60 K. Show at least four tie-lines under the dome. Show only the 1 bar isobar surface for clarity.

T - s - x 3-D plot showing Linde dual-pressure liquefaction. Show only the state points connected by straight, dashed lines. Show only the 1 and 5 bar isobar surfaces for clarity.