Compressibility Factor Calculator for Gases

Erick Garcia, Brittney Valerio, and Riley Shuping

CHE 348

Dr. Lea Hildebrandt Ruiz

**1. What does the authors of the code tell you about how the code works?**

The purpose of the code is to approximate the compressibility factor, Z, for a specific gas, given a value or range of temperatures and pressures. The compressibility factor is calculated based off of the equation of state known as Van der Waals. The first process establishes the function and requests three inputs when the function is used: the gas name as a string, temperature(s) in Kelvin, and pressure(s) in Bar. In this code the author, Chad Green, starts off by assigning the function inputs to specific variables. Next, Green uses the “switch” built-in MatLab function in order for the user to obtain critical information for the specific gas they want to evaluate. Within each gas case, critical temperature and critical pressure values to initialized. These values are obtained from the Gas Encyclopedia provided by Air Liquide according to the author. In the event of a quantum gas, the author includes a required Newton’s correction to the critical temperature and critical pressure. He replaces the critical values that were originally obtained from Air Liquide with the adjusted values. Then, Green calculated the reduced temperature(s) and reduced pressure(s) by dividing the pressure and temperature input by the critical values of the gas. A reasonable range for the values of the compressibility factor are initialized. Green then uses the “if” function to determine in the lengths of the temperature and pressure inputs match. If the are not the same length, then a warning message will be produced. Two other “if” functions are used to determine if the reduced temperature and reduced temperature need to be fixed using Brute-force. Finally, Green creates a row vector for the value of Z before the loop begins. In order to calculate the actual values of the compressibility factor the author uses a “for” loop. The for loop will calculate different values of the Van der Waals equation for each input of temperature and pressure. The compressibility factor, Z, will be calculated by using the interpolation function “interp1q” and three input variables: the vectors for Van der Waals equation, reasonable Z values, and the number 1.

**2. References**

Air Liquide. (2016). *Gas Encyclopedia Air Liquide*. Retrieved from Air Liquide: https://encyclopedia.airliquide.com/

**3. What are the potential sources of error in your final solution? Include information on the numerical methods used, the granularity of the problem and significant digits.**

One potential source of error is due to the numerical function we use to obtain the compressibility factor. We compared the code to another program which uses a third order “cubic” approximation of the compressibility factor using Kapuno’s algorithm (named compressibilityexact). Compared 7 to our program’s numerical method, which uses 1-D interpolation (compressibilitynonexact), R. Kapuno’s algorithm is of a higher order. In class we have learned that higher order equations contain a smaller numerical error. Therefore, we can conclude that the algorithm described by R. Kapuno in the book “Programming for Chemical Engineers Using C, C++, and MATLAB” has smaller errors that simple 1-D interpolation.

|  |  |
| --- | --- |
| **1a) Methane (452 K and 8 Bar)** | **1b) Neon (452 K and 8 Bar)** |
| **2a) Carbon Dioxide ( 253 K and 2 Bar)** | **2b) Carbon Dioxide ( 600 K and 6 Bar)** |

Figure 1: Tests to determine whether there’s an error in the 1-D interpolation compared to Kupno’s algorithm. The first test(1) compares two different elements at the same condition. The second test (2) compares the same element at two different conditions.

In two tests performed, which are displayed above, we sought to understand the percentage error between R. Kapuno’s algorithm and the 1-D interpolation function included in MatLab. First, we compare two different elements at the same conditions. Then, we compare the same element at different conditions. In both tests, different errors were obtained. By analyzing our tests, we are able to not only confirm the presence of an error in 1-D interpolation, but we are also able to conclude that the error depends both on substance and conditions.

Another potential source of error is the data in the original program obtained from Air Liquide. Although the source is noted as a very credible one, the Tc and Pc values for each element are reported in 3-5 significant figures. As the Tc and Pc values are rounded, important information regarding the properties of the elements are rounded. If more significant figures were reported, your results would be more and more accurate. It is also important to shine light on the fact that these values for Tc and Pc are tabulated. This means that they are obtained based off experimental data. Even if we added several significant figures, our calculations still would have a small error.

Finally, a potential source of error is the Van der Waals Equation of State used to calculate the compressibility factor. While it is an improvement from the ideal gas law above critical temperature. it does not properly estimate compressibility factors at or below critical temperature. Also the model breaks down at high pressures and does not reflect any deviation from ideal state. That provides a source of error for the actual compressibility factor in those isotherms and isobars. There exists other equations where the compressibility factor can be calculated such as the virial equation. The virial equation is like a Taylor Series expansion equation in the sense that we can expand to infinite amount of terms and obtain a very accurate answer. We could also chose to expand the virial equation only N terms so that there would be an error proportional to (1/N)N+1.

**4. Make a plot of your choice by editing the code and plot the variation in the dependent variable as a function of the independent variable. Provide information on the original and modified plots.**

We added a new gas called ethane or C2H6 with a critical temperature of 305.4 K and a critical pressure of 48.839 bars. In order to enhance the code, we modified the code so that there’s a plot with three different temperatures. These three different temperatures cross the critical temperature across a range of temperatures above and below the critical pressure. The three different temperatures are below, approximately at, and above critical temperature. All of these are plotted against the compressibility factor, Z, which tells us the deviation from an ideal case so that we may observe where the model fails. The original plot uses a temperature range with a constant pressure to plot it against the compressibility factor. We are not able to see how the gas behaves as we change pressure. Therefore, the modified graph allows us to further analyze the behavior of the gas before, at, and after critical conditions. The parameters were changed in our modified code to be pressure versus compressibility factor at three different temperatures. Refer to the modified code named “compressmod”.

**5. Add labels, title, legend, etc. to the plot to make it self explanatory. If using numerical methods, use at least 2 different time steps.**

|  |  |
| --- | --- |
| **1a) Original Code** | **1b) Original Output** |
| **2a) Modified Code**  **code222.PNG** | **2b) Modified Output**  **Capture.PNG** |

Figure 2: The figure above depicts the difference between the original code and plot provide by the author and the code we wrote to enhance the plot. The modified code contains a legend and is more self-explanatory.

**6. Based on the courses you have taken so far and this course, interpret the variation in the above plot based on your chemical engineering knowledge. You can use chemical engineering equations for this, from other references.**

C2H6 has a critical temperature of 305.4 K and a critical pressure of 48.839 bars. Based off of the plot shown above, the 100K line, which is below the critical temperature is linear for the most part. Between ~5 bars and 100 bars, the compressibility factor is below 1. When the pressure is at the critical point, the compressibility factor is approximately 0.5. In the temperature plot of 305K, we can see a concave curve that starts at a compressibility factor value of 1 at a pressure of 0 bars and does down to a compressibility factor of about 0.4 at the critical pressure. Once this temperature curve reaches the critical pressure, the graph starts moving in a positive slope. The slope of the 305K line after critical pressure is less than the slope of the 100K line after the critical pressure. The line representing temperature at 700 K, which is above critical temperature, is linear for the most part. Above 300 bars, the 700 K line begins to move away from a compressibility factor of 1 and has a small positive slope.

Liquid-gas interface interactions have an effect on the compressibility of a gas, along with temperature and pressure. Based off the plot above, with lower temperatures, the compressibility factor is increased. When a gas that is maintained at a lower temperature is subjected to low pressures, the molecules interact more like a liquid. The attractive forces dominate in this region. But as pressure increases, the repulsive forces tend to dominate. This differences between the attractive and repulsive forces appear more strongly in gases with lower kinetic energy (lower temperature). As temperature is increased, the compressibility of the gas is decreased due to the higher kinetic energy and increased repulsive forces felt between molecules. At sufficiently high temperatures, the compressibility approaches one - which corresponds to that of an ideal gas. What temperature this occurs at varies between different substances.

As a result, sufficiently high temperatures lend themselves to a more accurate approximation. Because as temperature increases, the average kinetic energy (temperature) increases, and the gas phases is more favorable. With lower pressures, the liquid-vapor interphase favors more liquid interactions and the model begins to break down. This is emblematic of the Van der Waals equation itself, not the code. And due to the differences in reactivity between different gasses, the exact pressure needed for the accurate model depends on which gas.

Polar and nonpolar covalent bonds are generally weak enough to allow for gasses to form, assuming the molecules are made up of lighter elements. This means that lighter organic molecules, diatomic molecules, and the lightest elements all appear as gasses. Ionic bonds are too strong to break and form gasses under normal conditions. Even weaker bonds, such as hydrogen bonds can have an effect on the phase-change of molecules. These effects on phase-change all factor into compressibility. The stronger intermolecular forces are, the greater temperature must be to overcome, and the greater the compressibility of the substance.

**7. Cite all references you use.**

Air Liquide. (n.d.). *Gas Encyclopedia - Chlorodifluoromethane*. Retrieved from Air Liquide: https://encyclopedia.airliquide.com/chlorodifluoromethane

Gershenson, M. (2007, January 15). *Lecture 15. The van der Waals Gas (Ch. 5).* Retrieved from Physics 351 - Thermal Physics: www.physics.rutgers.edu/~gersh/351/Lecture%2015.ppt

Smith, I. E. (2012, June 18). *Order of magnitude of number.* Retrieved from MathWorks: https://www.mathworks.com/matlabcentral/fileexchange/28559-order-of-magnitude-of-number

Transtutors. (n.d.). *Van Der Waal's Equation*. Retrieved from Transtutors: http://www.transtutors.com/chemistry-homework-help/gaseous-state/van-der-waals-equation.aspx

**8. Improve the above code. Make a list of changes and explain their advantages in the report. Document the changes in the report. To get full credit, make at least 2 significant changes.**

We created an Guest User Interface using Matlab GUIDE so that the user can input values of temperature and pressure in different units and array sizes as well as a string of the gas in editable boxes. It has a Calculate button which produces the answer(s) in a scrollable box as well as a plot of values in different parameters against Z. The plot will have its limits from the lowest and greatest array values and will create intermediate points for a correct plot. If one of the inputs is a constant and the other an array, the array parameter will be used for the plot. If both are constant then two separate plots will be created one with constant T and the other with constant P and it will add a value of the same order of magnitude (using an order function) as the constant to augment the array for plotting. The order function was found on file exchange. Please refer to the files named “compressibility” and “compressibilitycalculator.”