Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #47059

Leveraging MATLAB for Non-Linear Thermodynamics Analysis in Engineering Education

Dr. Emmanuel K. Glakpe, Howard University

Dr. Emmanuel Glakpe is a professor in the Mechanical Engineering Department at Howard University in Washington DC and a registered professional engineer. He received BSc., M.S., and Ph.D. degrees from the University of Science and Technology, Ghana, Stanford University, CA, and University of Arizona, AZ., respectively. A Fellow of ASME, Dr. Glakpe teaches classes in the Fluid/Thermal Sciences thread of the curriculum in both the undergraduate and graduate programs in the Department at Howard. His research interests are in the broad areas of Energy, Computational Fluid Dynamics, and Global Engineering Education. He has published extensively in these areas and has received several awards for his contributions to enhancing teaching and learning. He continues to work collaboratively with colleagues in Africa, Europe, and China in advancing best practices in global engineering education.

Aavash Budhathoki, Howard University

Leveraging MATLAB for Non-Linear Thermodynamics Analysis in Engineering Education

Abstract

The introductory and applied thermodynamics courses in undergraduate mechanical engineering programs teach students the fundamental principles, laws of physics, and the application of these principles to solve real-world problems. Solving thermodynamic problems not only helps students understand the science of energy but also strengthens their critical thinking skills. However, as problems become more complex and realistic, analyzing them can become tedious, often requiring the derivation and solution of numerous equations. This complexity is especially evident in transient systems where mass, energy, and entropy balances can lead to a combination of linear and non-linear equations, depending on the system's parameters.

In this paper, a demonstration of the use of MATLAB for teaching and learning Thermodynamics is reported, resulting in a detailed solution process, including a flowchart and an example problem with derived governing equations. The non-linear terms in the governing equations are identified and solved first using the MATLAB Symbolic Math Toolbox (Syms). The remaining equations then are all linear, allowing them to be arranged into a matrix form for an efficient solution. The MATLAB output is tallied with solution from Engineering Equation Solver (EES) Software, comparing the accuracy of both methods while observing the difference in utilizing each approach for overall learning and apprehension. Finally, an optimization study is then carried out in MATLAB by varying a key parameter (Pressure) that influences the system's behavior. The findings suggest that the MATLAB approach enhances understanding of fundamental principles and their application in solving complex thermodynamic problems in an educational setting and can also be beneficial for industry practitioners involved in the design and optimization of large-scale energy problems.

Keywords: Thermodynamics, teaching, learning, MATLAB, optimization, EES, critical thinking, Non-Linear, Symbolic Toolkit.

1.0 Introduction

Thermodynamics, the science of energy, a cornerstone course in undergraduate mechanical engineering curriculum, has long held a reputation for being one of the most challenging subjects encountered by students in their quest to become engineers. This fundamental course introduces students to the laws of mass and energy conservation, entropy generation, exergy principles, and their practical applications to real-world engineering systems. The complexity of thermodynamics analysis stems from multiple factors: its abstract concepts, the interconnection between various physical principles, and the mathematical rigor required to analyze systems. The complexity of coursework is especially evident in the second semester course for Thermodynamics (commonly Thermodynamics II or Applied Thermodynamics), which expands analysis to a more complex level, involving transient analysis and using the laws of mass and energy conservation, entropy balance with a generation component, and other equations, often simultaneously for analysis. While these problems are expected to be solved analytically, the analysis of these linear and non-linear equations becomes a cumbersome task often deterring students from understanding and applying these concepts to get a broader understanding of their significance in engineering applications.

The use of computational tools and software has been theorized to greatly simplify numerical analysis while also being able to graphically visualize output for further analysis [1]. Furthermore, the benefits of using computer-assisted tools to improve student learning are widely tested, in laboratories [2, 3, 4, 5], and learning teaching [6, 7, 8, 9]. In order to enhance the teaching and learning of thermodynamics, the approach to teaching thermodynamics has progressed from the traditional method to a more sophisticated method such as using computer technology and multimedia [10]. Because of the complexity of the course, including further tools for study risks the potential issue of further steepening the learning curve and deter students entirely from pursuing the study of thermodynamics. However, data from literature tends to support the use of computation tools in enhancing the learning experience. According to Mulop et al. [10], "the adoption of the blended learning approach has resulted in an improved students' performance as measured by their final examination results. From 2000 – 2004, the percentage of students achieving the minimum examination pass mark improved from 49% to 77%." Additionally, for web-based learning tools focused on multi-staging in compressors and turbines, "there was a 14% improvement in the average score of a quiz administered for the group using the module over the group without the exposure to the module." Additionally, from Domínguez et al. [11], there is strong statistical evidence demonstrating the efficacy of computational tools: "After implementing the computer applications, the academic results showed an increase of up to 16% in the pass rate." Additionally, student perception data supports this with "70% (of students believing) that the utilization of these computer applications represented a significant improvement in teaching." A general survey was carried out by the students of Applied Thermodynamics and Heat Transfer at Howard University with specific questions on their apprehension of the concepts involved, improvements on problem solving and coding skills, and their preference on traditional assignments compared to computational projects.

Existing data suggest positive results from surveys and literature data regarding the use of computational tools in the analysis of Thermodynamics problems. To get sample data to confirm these results, the major step is to choose a suitable tool that enriches the learning experience of the students while also being able to handle linear and non-linear analysis of complex equations. Microsoft Excel and Python have been very popular for modelling and solving unknown characteristics of thermodynamic systems [12,13], in addition to other software and frameworks including Julia [14], ANSYS, ChemCAD, Hysys, among others. However, due to the extensive technical requirements and background required, these resources are not as popular for teaching-learning in an undergraduate setting. Given these considerations, MATLAB emerges as an ideal computational tool for enhancing the teaching and learning of thermodynamics in undergraduate mechanical engineering programs. Mechanical engineering students are commonly introduced to MATLAB in their introductory courses, making it a familiar platform for computational analysis. Its intuitive syntax and user-friendly interface reduce the learning curve for new users, allowing them to focus on understanding complex thermodynamic concepts rather than grappling with intricate programming languages. Furthermore, MATLAB offers a variety of specialized toolboxes that are particularly beneficial for thermodynamic analysis. The Symbolic Math Toolbox enables symbolic computations, allowing students to manipulate and solve complex equations analytically. The Optimization Toolbox provides algorithms for solving linear and nonlinear optimization problems, which are essential in the design and analysis of thermodynamic systems. Simulink, a graphical programming environment within MATLAB, facilitates modeling and simulation of dynamic systems, aiding in the visualization of transient thermodynamic processes [15].

Another computational tool that has been developed and is included as attached software for several undergraduate thermodynamics, heat-transfer and fluid mechanics textbooks from McGraw-Hill and other publishers is Engineering Equation Solver (EES) [16]. The commercial software package is designed to be used for solutions of simultaneous non-linear equations. EES includes desirable features such as thermodynamic property tables for a variety of fluids — which eliminates the iterative problem solving by hand, with code that retrieves/interpolates properties at the specified thermodynamic properties — parametric tables, and optimization tools. However, while EES facilitates direct problem-solving through equation input and automated parameter calculation, this "black box" approach may limit students' understanding of the underlying computational methods. In contrast, the MATLAB approach of this paper requires students to develop their own solution algorithms, promoting a deeper apprehension of both the thermodynamic principles and the numerical methods used to solve complex equations. Through the process of coding their own solutions using MATLAB's Symbolic Math Toolbox and other built-in functions, students gain valuable insights into how properties are interpolated, how non-linear equations are solved iteratively, and how optimization algorithms work. This hands-on experience with computational methods greatly enriches the learning process by promoting critical thinking and cultivates an appreciation for scientific computing in engineering. Therefore, while EES may be a preferred tool for rapid problem-solving in industry and advanced academic research, MATLAB's educational value lies in its ability to demystify the computational aspects of thermodynamic analysis, making it particularly well-suited for undergraduate education where building fundamental understanding takes precedence over solution efficiency.

1.1 Problem Definition

Thermodynamic system analysis integrates mass and energy conservation laws with component-specific equations for various equipment like pumps, turbines, and heat exchangers. The complexity increases when incorporating entropy generation and irreversibility effects from the second law of thermodynamics, leading to exergy destruction calculations. These analyses often yield large equation systems requiring computational tools, particularly since solutions depend on fluid properties that vary throughout the system. This research develops a methodology combining numerical analysis with property tables to demonstrate the solution to such complex equation systems. The approach aims to enhance both system design capabilities and serve as an educational tool for project-based learning. The objective is to solve a system of derived thermodynamic equations for transient systems using MATLAB, plot the outputs, and parametrically observe the change in results based on a change in parameter. MATLAB's SYMS toolbox will be used to solve any non-linear equations, to import property values, and to interpolate between two states. The flowchart shown in Fig. 1 describes the working logic of the MATLAB program. A transient system is used for analysis because it introduces non-linearities in the form of quality and time-dependent equations. A similar problem-solving approach can be applied for steady state analysis too, with MATLAB not requiring the use of SYMS toolbox to solve for any non-linearities. An example problem taken from Van Wylen and Sonntag [17] and modified for clarity is solved using the approach advanced in the paper. The definition of the problem along with the schematic is shown in Figure 2. Additionally, the Engineering Equation Solver (EES) is used to solve the same thermodynamic equations and compare the results with those obtained from MATLAB. This comparison helps ensure accuracy and reliability of the proposed approach.

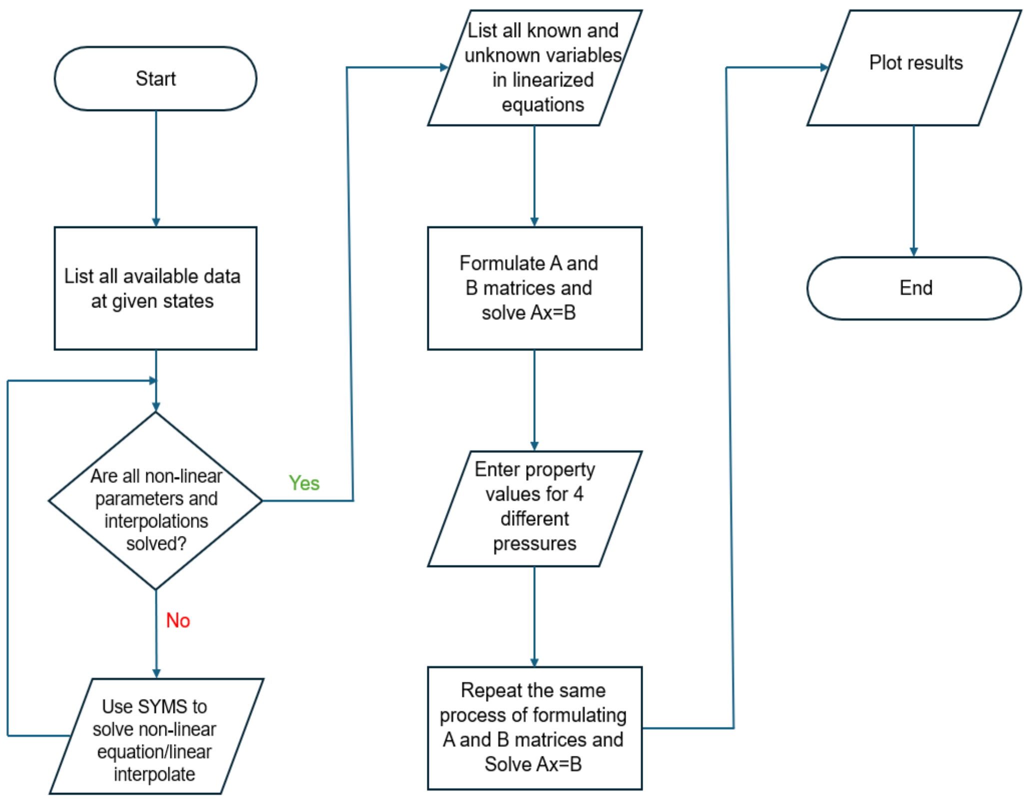


Figure 1: Flowchart showing logic for MATLAB code

1.2 Methodology

This study employed a two-phase approach to evaluate the effectiveness of computational tools in enhancing students' understanding of complex thermodynamics concepts. The research was conducted with a cohort of 20 mechanical engineering students enrolled in an advanced thermodynamics course. As part of the first phase, the students were presented with conventional assignments (one sample problem defined in section 1.3) that required manual calculations and analytical problem-solving techniques. This established a baseline for comparing traditional solution methods with computational approaches and allowed students to develop a fundamental understanding of the underlying thermodynamic principles.

Next, the students were assigned the same or similar problems to be solved computationally using MATLAB. To ensure a systematic and organized approach, students followed a structured workflow as illustrated in Figure 1. Students first listed all known/given parameters and performed state analysis with these conditions, followed by using MATLAB Syms toolbox for the evaluation of non-linear parameters and interpolations. They then identified variables in linearized equations and formulated matrices to solve for remaining unknowns. The students also utilized their MATLAB code to conduct parameter optimization studies by varying outlet pressure and studying its impact on the mass, entropy, and heat transfer within the system. This phase enhanced their understanding of how varying design parameters affect system performance and introduced them to real-world engineering optimization challenges.

The effectiveness of this computational approach was evaluated through multiple assessment methods. Solution validation was performed through cross-verification of MATLAB results with Engineering Equation Solver (EES), which helps assess solution accuracy and computational efficiency. Next, the study aimed to analyze student performance, comparing grades between traditional homework assignments and exams to MATLAB-based projects. Furthermore, student feedback was collected through a comprehensive survey assessing their comfort level with computational tools, perceived improvement in understanding core thermodynamic concepts, practical applicability of the learned computational skills, and overall learning experience and engagement. In summary, the team performed detailed comparisons of numerical results between MATLAB and EES solutions, conducted quantitative analysis of student performance metrics, and carried out qualitative assessment of student feedback and learning outcomes.

This methodological framework has been designed to be readily adaptable by other ASEE community members, allowing for implementation across various thermodynamics courses and similar engineering subjects. The structured approach ensures reproducibility while maintaining flexibility for customization based on specific course requirements and student needs. The methodology emphasizes not just the technical implementation but also the pedagogical aspects of incorporating computational tools in engineering education. This balanced approach ensures that students develop both theoretical understanding and practical computational skills necessary for modern engineering practice.

1.3 Plant Technology Example Problem

A thermodynamic analysis of the transfer of liquids by pressurized gas plant technology is to be performed. Two tanks A and B (see schematic) containing the refrigerant R-134a are connected by a valve. The fluid in tank A is initially saturated vapor at 80 ^{o}F with a volume of 3.2 ft^3, and in the insulated tank B it is at -40 ^{o}F, quality of 0.01 with a volume of 2.5 ft^3. The valve is opened slightly, allowing the fluid to flow from A to B. A pressure regulator connected to tank B allows liquid to flow out when the pressure inside B reaches 23.85 lbf/in^2 and this continues until the pressure in tank A has dropped to 23.85 lbf/in^2. The amount of heat transferred to tank A is enough to keep the temperature of the R-134a in tank A constant at 80 ^{o}F as the specific enthalpy out of the tank is 116.45 Btu/lbm. Determine key thermodynamic parameters of Tank A and B in this engineering system for varying values of exit pressure.

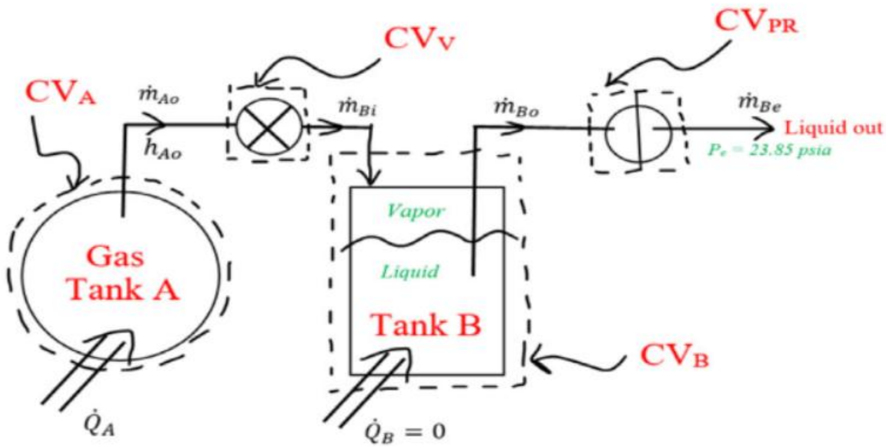


Figure 2: Plant Technology

2. Problem Nomenclature

Some key parameters given in the problem are listed in this section. The bold letters indicate different properties: ***u*** for specific internal energy, ***h*** for specific enthalpy, ***V*** for Volume, ***υ*** for specific volume, ***T*** for temperature, ***x*** for quality, ***Q*** for Heat Transfer, and ***m*** for mass. The subscripts denote the fluid in tank (**A** denoting tank A, **B** denoting tank B), and since the problem is a time-dependent (transient) problem, initial time is denoted by $\mathbf{t_1}$ while final time is denoted by $\mathbf{t_2}$. Finally, the saturated liquid and saturated vapor states are denoted using the common subscripts of **f** and **g** respectively. A detailed explanation of all variables can be found below.

$u_{B,t1}$ Specific internal energy of tank B at time t_1
$h_{A,t1}$ Specific enthalpy of fluid in tank A at time t_1
$h_{A,t2}$ Specific enthalpy of fluid in tank A at time t_2
h_{Be} Specific enthalpy of fluid leaving B tank at time t_1
V_B Volume of tank B
$\upsilon_{B,f,t2}$ Specific volume of saturated liquid of fluid in tank B at time t_2
$\upsilon_{B,fg,t2}$ Difference in specific volume of saturated vapor and liquid of fluid in tank B at time t_2
$\upsilon_{B,t1}$ Specific volume of fluid in tank B at time t_1
$\upsilon_{A,t1}$ Specific volume of fluid in tank A at time t_1
$\upsilon_{A,t2}$ Specific volume of fluid in tank A at time t_2
$u_{B,f,t2}$ Specific internal energy of saturated liquid of fluid in tank B at time t_2
$u_{B,fg,t2}$ Difference in specific internal energy of saturated vapor and liquid of fluid in tank B at time t_2
V_A Volume of tank A
$u_{A,t1}$ Specific internal energy of fluid in tank A at time t_1
$u_{A,t2}$ Specific internal energy of fluid in tank A at time t_2
T_A Temperature of tank A
T_B Temperature of tank B
$h_{Ao,avg}$ Average specific enthalpy of fluid exiting tank A
$s_{Ao,avg}$ Average specific entropy of fluid exiting tank A

$\chi_{B,t2}$	Quality of fluid in tank B at time t_2
$m_{B,t2}$	Mass of fluid in tank B at time t_2
m_{A0}	Mass leaving tank A
Q_A	Heat Transfer into Tank A
$S_{gen,A}$	Entropy Generated in Tank A
m_{Bo}	Mass of fluid leaving Tank B
$S_{gen,B}$	Entropy Generated in Tank B
$s_{B,t2}$	Specific Entropy of Fluid in Tank B at time t_2
$u_{b,t2}$	Internal Energy of Tank B at time t_2
$m_{A,t1}$	Initial Mass of Tank A
$m_{A,t2}$	Final Mass of Tank A
$m_{B,t1}$	Initial Mass of Tank B

3. MATLAB Approach using Fundamental Equations

3.1. Mass Conservation Principle

The rate of change of mass within a control volume is equal to the net rate at which mass enters or leaves the control volume through its boundaries. For Control Volume 1 (see Fig. 3), it can be mathematically expressed as:

$$\Sigma \dot{m}_{out} + \frac{\Delta m}{\Delta t}|_{CV1} - \Sigma \dot{m}_{in} = 0$$

Integrating between the time limits 't_1' and 't_2', we get,

$$\Sigma m_{out} + m_{CV1}(t2) - m_{CV1}(t1) - \Sigma m_{in} = 0 \ldots (1)$$

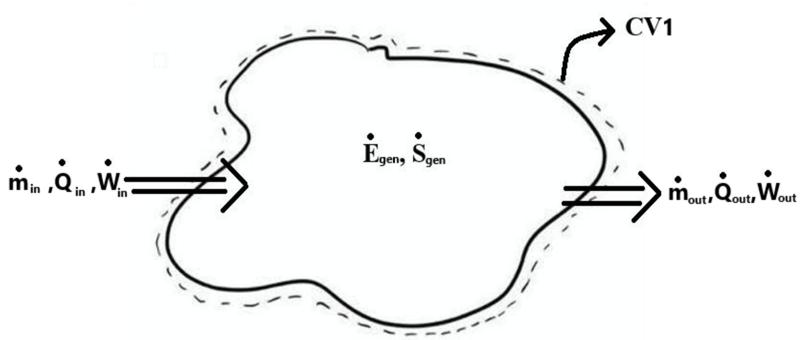


Figure 3: Control Volume 1

3.2. Law of Energy Conservation

The rate of change of energy within a control volume is equal to the net rate of energy entering or leaving the control volume through its boundaries, plus the rate of energy generation within the control volume. For Control Volume 1 (see Fig. 3), it can be mathematically expressed as:

$$\Sigma \dot{E}_{out} + \frac{\Delta E}{\Delta t}|_{CV1} - \Sigma \dot{E}_{in} - \dot{E}_{gen} = 0$$

Integrating between the time limits 't_1' and 't_2', we get,

$$\Sigma E_{out} + E_{CV1}(t_2) - E_{CV1}(t_1) - \Sigma E_{in} - E_{gen} = 0 \ldots (2)$$

3.3. Entropy Equations

The rate of change of entropy within a control volume is equal to the net rate of entropy entering or leaving the control volume through its boundaries, plus the rate of entropy generation within the control volume due to irreversibilities. For Control Volume 1 (see Fig. 2), it can be mathematically expressed as:

$$\Sigma \dot{S}_{out} + \frac{\Delta S}{\Delta t}|_{CV1} - \Sigma \dot{S}_{in} - \dot{S}_{gen} = 0$$

Integrating between the time limits 't_1' and 't_2', we get,

$$\Sigma S_{out} + S_{CV1}(t_2) - S_{CV1}(t_1) - \Sigma S_{in} - S_{gen} = 0 \ldots (3)$$

For the given problem, we assume that the changes in kinetic and potential energy values for the mass stream are negligible when compared to enthalpy and that there is no energy generation in the tanks and valves. The entropy generated in the two valves is assumed to be zero. Using these general assumptions in conjunction with Equations (1), (2) and (3), we arrive at two sets of equations, one pertaining to a set of variables relating to the non-linear quality term and the remaining parameters which can be solved with the coefficient matrix, A, and constant matrix, B.

3.4. Equations Pertaining to the Non-Linear Quality Term

After applying equations 1-3 to the given control volumes, equations 4-15 are obtained. Equations 4-7 contain four unknowns (i.e. final state of tank B is unknown and thus the values cannot be derived using property tables): m_{Bo}, $m_{B,t2}$, $u_{B,t2}$, and $\chi_{B,t2.}$ Despite having four unknowns, and four equations, trying to solve the equations simultaneously results in multiplication of these variables making them non-linear (degree of the equation is not one). This restricts the use of matrices and so, we must obtain the values of these parameters utilizing an additional step, which is to use SYMS toolbox to solve these variables. The obtained values can then be plugged into the remaining linear equations and solved using matrices.

$$m_{B,t2}u_{B,t2} - m_{B,t1}u_{B,t1} = h_{Ao,avg}m_{Ao} - h_{Be}m_{Bo} \ldots (4)$$
$$m_{B,t2} = \frac{V_B}{v_{B,f,t2}+\chi_{B,t2}v_{B,fg,t2}} \ldots (5)$$
$$m_{B,t2} - m_{B,t1} = m_{Ao} - m_{Bo} \ldots (6)$$
$$u_{B,t2} = u_{B,f,t2} + \chi_{B,t2}u_{B,fg,t2} \ldots (7)$$

The MATLAB code that utilizes the SYMS toolkit and solves the non-linear problem along with the complete implementation is available in a GitHub repository [18].

3.5. Linear Equations for Matrix Formation

The following are the resulting linear equations.

$$-m_{A,t2} + m_{A,t1} = m_{Ao} \ldots (8)$$
$$m_{A,t2}u_{A,t2} - m_{A,t1}u_{A,t1} = Q_A - h_{Ao,avg}m_{Ao} \ \ldots (9)$$
$$S_{gen,A} = \frac{Q_A}{T_A} + m_{Ao}s_{Ao,avg} - m_{A,t2}s_{A,t2} + m_{A,t1}s_{A,t1} \ldots (10)$$
$$S_{gen,B} = -m_{Ao}s_{Ao,avg} + m_{Bo}s_{Be} + m_{B,t2}s_{B,t2} - m_{B,t1}s_{B,t1} \ldots (11)$$
$$s_{B,t2} = s_{B,f,t2} + \chi_{B,t2}s_{B,fg,t2} \ldots (12)$$
$$m_{A,t1} = \frac{V_A}{v_{A,t1}} \ldots (13)$$
$$m_{A,t2} = \frac{V_A}{v_{A,t2}} \ldots (14)$$
$$m_{B,t1} = \frac{V_B}{v_{B,t1}} \ldots (15)$$

3.6. Mass Balance and Property Relations

Based on the problem definition, and the nomenclature of the solution approach, the following relations between the flow variables are deduced.

$$m_{Ao} = m_{Bi}$$
$$m_{Bo} = m_{Be}$$
$$h_{Ao} = h_{Bi}$$
$$h_{Bo} = h_{Be}$$
$$s_{Bo} = s_{Be}$$

Equations (4) to (7) are used to obtain a solution for $\chi_{B,t2}$ and $m_{B,t2}$ in the symbolic math toolbox. Equations (8) to (15) can then be solved simultaneously in matrix form, Ax=B, in which the matrices A, B and the vector of unknowns x are obtained as follows.

3.7. Coefficient Matrix, A

Matrix A =

[1	0	0	0	0	0	0	1	-1	0
- hA_o.avg	1	0	0	0	0	0	uA,t_1	-uA,t_2	0
-sA_o.avg	-1/T_a	1	0	0	0	0	-sA,t_1	sA,t_2	0
1	0	0	-1	0	0	0	0	0	1
sA_o.avg	0	0	-sBe	1	- mB,t_2	0	0	0	sB,t_1
0	0	0	0	0	1	0	0	0	0
-hA_o.avg	0	0	hBe	0	0	mB,t_2	0	0	-uB,t_1
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1]

3.8. Constant Matrix, B

$B = [0_1;\ 0;\ m_{B,t2};\ 0;\ S_{B,f,t2} + \chi_{B,t2}S_{B,fg,t2};\ 0;\ V_A/\upsilon_{A,t1};\ V_A/\upsilon_{A,t2};\ V_B/\upsilon_{B,t1}]$

3.9. MATLAB Solver

x = A\B
where x is the matrix with unknowns:

$x = [\ m_{Ao};\ Q_A;\ S_{gen,A};\ m_{Bo};\ S_{gen,B};\ s_{B,t2};\ u_{b,t2};\ m_{A,t1};\ m_{A,t2};\ m_{B,t1}]$

4. Engineering Equation Solver (EES) Solution

The solution process in EES is simple relative to the steps executed on MATLAB. Because of the existing property table for R-134a, obtaining state values is relatively simple and can be done using the 'CALL realthermoprops' function after defining the working fluid using "R$='R134a'". Also, since the number of unknowns is equal to the number of equations, equations 1-12 can be directly typed onto the EES Equations Window after retrieving property values. EES being able to solve non-linear equations solves all equations simultaneously giving us the required output. The EES code for the example problem is listed in the Appendix.

5. Results and Discussion

5.1 MATLAB and EES Results at Exit Pressure of 23.85 psia

After obtaining the quality values, the MATLAB operator '\' is utilized to obtain the solutions to the ten simultaneous equations. The results are listed in Table 1.0 below.

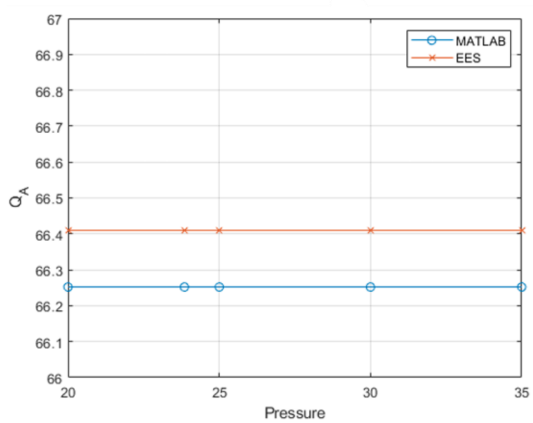
Table 1.0: Key Parameters obtained from MATLAB and EES

Parameters	MATLAB output	EES output
$\chi_{B,t2}$	0.0310	0.03102
$m_{B,t2}$ [lbm]	35.07	35.05
m_{A0} [lbm]	5.408	5.408
Q_A [Btu]	66.41	66.41
$S_{gen,A}$ [Btu/°R]	2.546	2.546
m_{Bo} [Btu]	6.603	6.603
$S_{gen,B}$ [Btu/°R]	0.1399	0.140
$s_{B,t2}$ [Btu/lbm°R]	0.03712	0.03712
$u_{b,t2}$ [Btu/lbm]	16.27	16.27

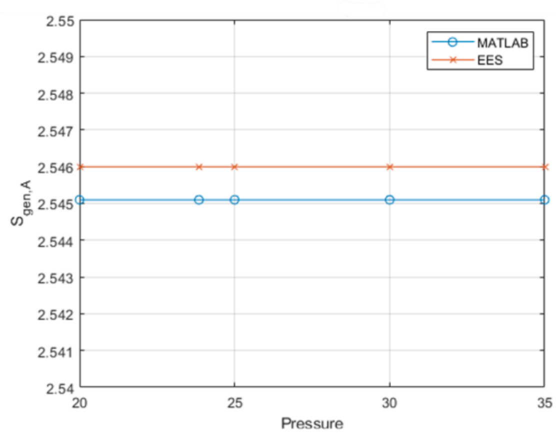
$m_{A,t1}$ [lbm]	6.798	6.798
$m_{A,t2}$ [lbm]	1.39	1.39
$m_{B,t1}$ [lbm]	35.05	35.05

5.2 Comparison of MATLAB and EES Results for Varying Exit Pressure Values

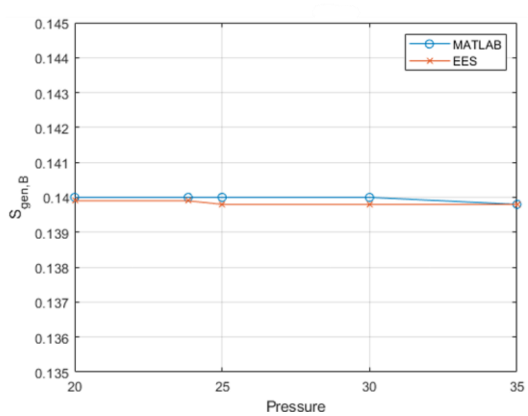
The flexibility of the proposed solution on MATLAB is demonstrated by changing one of the given parameters, P_e and comparing the results with that obtained from EES. The code is run for varying pressure values, namely 20, 25, 30, and 35 psia. By changing just a few property values at this state, all the other key parameters can be conveniently calculated to study the effects of the variation in exit pressure (see Figure 4 i-v.).



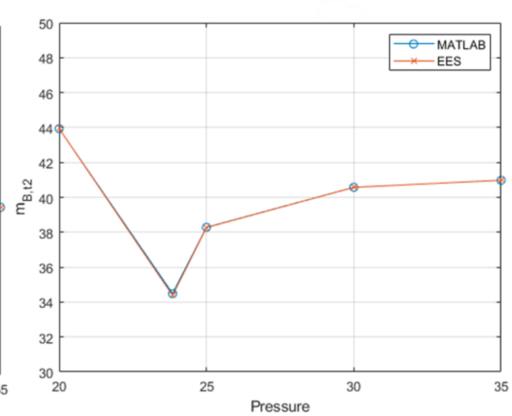
i.



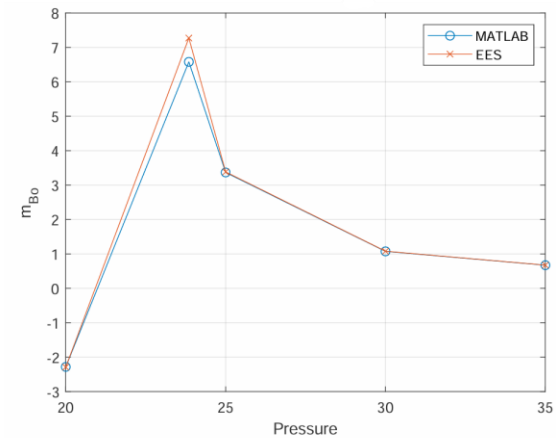
ii.



iii.



iv.



v.

Figure 4 - i: Variation of Heat Transfer to Tank A, **ii:** Variation in Entropy Generation in Tank A, **iii:** Variation in Entropy Generation in Tank B, **iv:** Variation in Mass Left in Tank B
v: Variation in Mass Leaving Tank B

The results obtained using MATLAB closely match those from the EES, as shown in the comparison tables 1.0 and 1.1. For example, the values for entropy generation in both tanks and the heat transfer to Tank A are nearly identical in both approaches. This agreement confirms the reliability of the MATLAB-based method for solving complex thermodynamic problems. The matching results provide confidence in using MATLAB for uncovering the logic used in advanced analysis software such as EES. The graphs from Figures 4-i to 4-v also offer valuable insights for optimizing the gas plant system. For instance, Figure 4-iii shows that the entropy generation in Tank B is lowest at 23.85 psia, suggesting this pressure is the most efficient for the system. Similarly, Figure 8 reveals how the mass remaining in Tank B changes with pressure, highlighting conditions where the system operates reliably. These results can help pinpoint the operational limits and optimal working conditions. Using such graphs, engineers can adjust parameters like pressure to improve system performance effectively. Optimization is further supported by observing how specific parameters interact with each other in the system. For example, the stable heat transfer in Tank A, as shown in Figure 4-i, confirms that its role as a heat reservoir remains unchanged regardless of pressure changes. Meanwhile, the variations in mass flow and entropy in Tank B highlight where inefficiencies or reversals occur. By analyzing these results, system design can be improved to prevent issues like mass flow reversal. These observations are especially useful for students learning to optimize systems with real-world constraints.

While both MATLAB and EES successfully solved the thermodynamic system analysis with comparable accuracy, MATLAB approach offers several distinct pedagogical advantages. First, MATLAB's widespread availability and integration into engineering curricula makes it a more accessible tool for students. Unlike EES, which requires paid subscriptions primarily available to instructors, MATLAB is typically provided to students through university licenses and is often introduced in foundational engineering courses, allowing students to build upon already familiar software. More importantly, the MATLAB approach fosters a deeper understanding of computational methods in thermodynamics. While EES provides a streamlined "black box" solution process where equations are simply input and solved, the MATLAB implementation requires students to develop a structured approach to problem-solving. Students using MATLAB must identify and separate linear from non-linear equations, learn to use the Symbolic Math Toolbox (SYMS) for handling non-linear terms like quality factors, understand matrix operations for solving systems of linear equations, and implement their own solution algorithms. This step-by-step process not only solves the problem but also provides valuable insights into how computational tools handle different types of equations. For instance, the process of using MATLAB's Syms toolbox to solve non-linear equations (demonstrated in Section 3.4) gives students practical experience with advanced mathematical tools, while the matrix formulation (shown in Section 3.7) reinforces linear algebra concepts in a real-world application. Therefore, while EES may be more efficient for industry applications where rapid solutions are prioritized, MATLAB's approach better serves educational objectives by developing both theoretical understanding and practical computational skills essential for future engineers.

5.3 Student Performance Analysis

A study examining the impact of computational tools on academic performance was conducted with 20 undergraduate students enrolled in a thermal science course. Students were assigned 3 ‘traditional’ homework assignments from thermodynamics textbooks [17,19] out of which two of the same problems were reassigned as projects to be solved using MATLAB. Analysis of student performance data (Fig. 5) revealed that MATLAB-based computational projects yielded significantly better outcomes compared to traditional assessments. Students scored an average of 12.55 points higher on computational projects than on traditional homework assignments, with the highest-class average of 69.70% achieved on a MATLAB-based project. Additionally, these computational assignments generated higher participation rates and submission frequencies. Performance followed a clear pattern across assessment types: computational tool-based assignments yielded the highest scores, followed by traditional homework, with closed books/notes examinations showing the lowest average grades. These findings suggest that MATLAB-based projects effectively enhanced student engagement and deepened their understanding of thermodynamic concepts, particularly when applying these principles to complex, real-world problems. The notably lower performance on exams and traditional homework assignments warrants further investigation to understand the underlying factors and potential areas for pedagogical improvement.

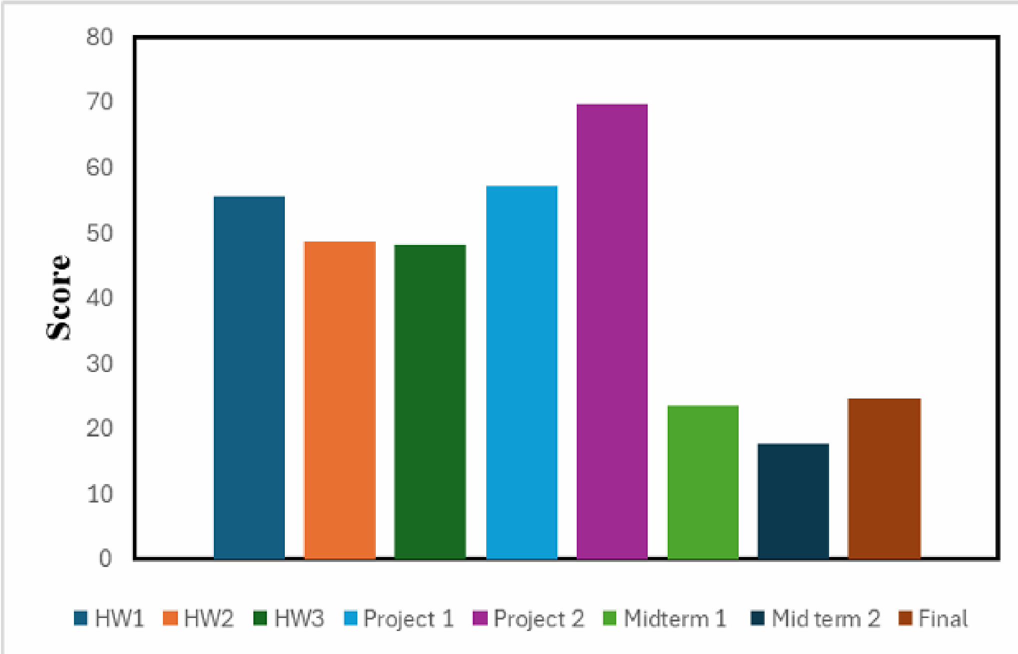


Figure 5: Average Score of Students in Assignments, Projects, and Examinations

5.4 Survey Results

Responses to nine survey questions (see Fig. 6) were collected to provide a broader understanding of how computational tools influenced the students' learning experience. These questions were based on a previous similar study carried out regarding the use of MATLAB in engineering thermodynamics teaching-learning which also aimed to study the impact of MATLAB in student learning experience [11]. These questions aim to evaluate the key impacts of the project such as improvements in understanding thermodynamics concepts, problem-solving abilities and the quality of teaching facilitated by the software. They also explored how the tools contributed to achieving better learning outcomes, fostering skills beyond the course, and encouraging students to consider similar applications in other contexts. Additionally, the survey examined the accessibility of the software and the role of academic staff in supporting its integration.

Number	Questions Included in the Survey
1	The use of the applications improved my understanding of the problems of the course.
2	The use of the problem applications improved my problem-solving skills.
3	I consider that the use of these tools represents a significant improvement in teaching.
4	The use of the applications promoted critical thinking in making engineering decisions.
5	I believe that the use of these tools helped me to improve the results obtained in the course.
6	The use of the applications has improved my competences beyond those of the course.
7	I would like to have more similar tools in the future in other similar courses.
8	I consider the use of these software applications to be important for the understanding of the classes
9	The academic staff actively engaged in creating access to the required software tools.

Figure 6: Survey Questions

The survey results (see Fig. 7) align closely with the performance trends shown in the chart, confirming the significant benefits of using computational tools like MATLAB and EES in thermodynamics education. Most students agreed that these tools enhanced their understanding of complex course problems, simplifying concepts and calculations that are often challenging to approach manually. By providing clear visualizations and enabling parameter optimization, these tools help make thermodynamics more accessible and engaging. Many students also reported that the tools improved their problem-solving skills, with many selecting "Strongly Agree." This shows that such applications foster critical thinking and technical expertise, giving students practical experience with real-world engineering scenarios.

Students overwhelmingly supported the use of these tools in future courses, reflecting a strong preference for technology-driven learning methods. This indicates that software tools are not only supplementary but essential for understanding complex topics and bridging the gap between theory and practice. However, feedback about academic staff support and access to software was mixed, highlighting a need for improvement in this area. Providing better resources, such as tutorials, software licenses, and workshops, could address these concerns and further enhance the learning experience. Overall, the survey underscores the value of computational tools in engineering education while identifying opportunities to maximize their impact through improved student support.

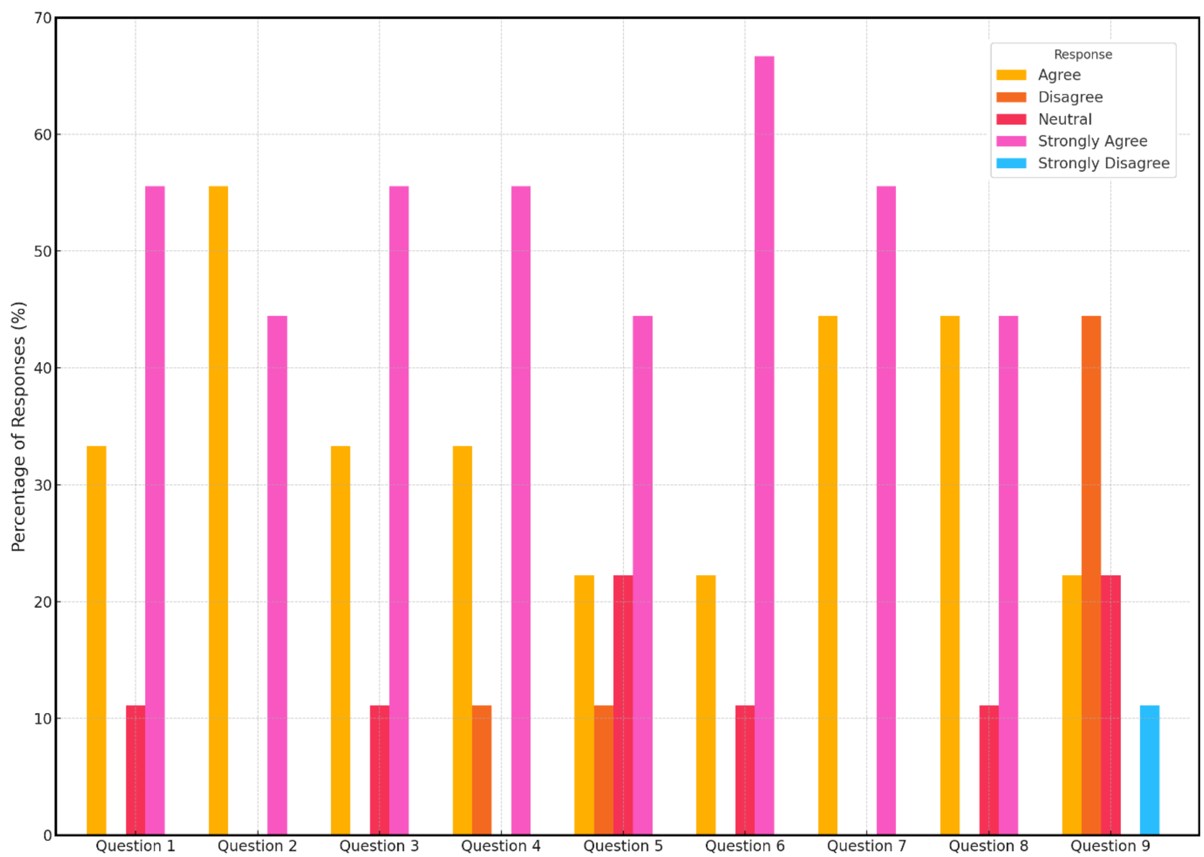


Figure 7: Survey Results

6. Conclusions

This study demonstrated the effectiveness of using MATLAB for solving complex thermodynamic problems in an educational setting. The implementation of computational tools in thermodynamics coursework showed significant benefits, with 89% of students reporting enhanced conceptual understanding. The paper presented a systematic approach to solving non-linear transient thermodynamic problems from various textbooks [17,19], using a gas plant technology example to demonstrate MATLAB's capabilities in handling complex equation systems and parameter optimization. The methodology not only simplified the analysis of complicated thermodynamic systems but also provided students with valuable computational skills applicable to real-world engineering challenges. The results suggest that integrating computational projects into thermodynamics education can significantly improve student engagement and understanding while preparing them for industry-standard analytical practices.

7. Acknowledgements

The authors gratefully acknowledge the participation of mechanical engineering students from the Class of 2025 at Howard University for their participation in the survey. Special thanks to the Department of Mechanical Engineering with their help in integrating MATLAB-based projects into the curriculum, which was instrumental in making this study possible.

7. References

[1] J. G. Acevedo, G. V. Ochoa, and L. G. Obregon, "Development of a new educational package based on e-learning to study engineering thermodynamics process: combustion, energy and entropy analysis," Heliyon, vol. 6, no. 6, e04269, Jun. 2020

[2] H. Binous and A. Bellagi, Orthogonal collocation method using Mathematica© in the graduate chemical engineering curriculum, Comput. Appl. Eng. Educ. 24 (2016), no. 1,101–113.

[3] F. J. Papa and S. Meyer, "A computer-assisted learning tool designed to improve clinical problem-solving skills," Comput. Methods Programs Biomed., vol. S0196-0644(89)80412-3, 1989.

[4] J. Ramírez, D. Soto, S. López, J. Akroyd, D. Nurkowski,M. L. Botero, N. Bianco, G. Brownbridge, M. Kraft, and A. Molina, A virtual laboratory to support chemical reaction engineering courses using real-life problems and industrial software, Educ. Chem. Eng. 33 (2020), 36–44.

[5] M. I. Szávuly, Á. Tóos, R. Barabás, and B. Szilágyi, From modeling to virtual laboratory development of a continuous binary distillation column for engineering education using MATLAB and LabVIEW, Comput. Appl. Eng. Educ. 27 (2019), no.5, 1019–1029.

[6] U. A. Al-Mubaiyedh and H. Binous, Teaching arc-length continuation in the chemical engineering graduate program using MATLAB©, Comput. Appl. Eng. Educ. 26 (2018), no. 4,1033–1049.

[7] E. Corral Abad, M. J. Gómez García, E. Diez-Jimenez, P. M. Moreno-Marcos and C. Castejón Sisamon, Improving the learning of engineering students with interactive teaching applications, Comput. Appl. Eng. Educ. 29 (2021), no. 6,1665–1674.

[8] E. Corral Abad, M. J. Gómez García, R. Ruiz Blázquez, C. Castejon and J. C. García-Prada, Effects of an android app on mechanical engineering students, Comput. Appl. Eng. Educ.26 (2018), no. 4, 1050–1057.

[9] R. Molina, G. Orcajo, and F. Martinez, Kbr (kinetics in batch reactors): A MATLAB-based application with a friendly graphical user interface for chemical kinetic model simulation and parameter estimation, Educ. Chem. Eng. 28(2019), 80–89.

[10] Mulop, N., Yusof, K. M., and Tasir, Z. "A Review on Enhancing Teaching and Learning of Thermodynamics." *Procedia - Social and Behavioral Sciences*, Vol. 56, pp. 703-712, 2012.

[11] Domínguez, J. C., Lorenzo, D., García, J., Hopson, C., Rigual, V., Alonso, M. V., and Oliet, M., 2023, "MATLAB Applications for Teaching Applied Thermodynamics: Thermodynamic Cycles," Computer Applications in Engineering Education, 31(4), pp. 900-915.

[12] P. Salvy, G. Fengos, M. Ataman, T. Pathier, K. Soh, V. Hatzimaikahs, pyTFA and matTFA: a Python package and a Matlab toolbox for Thermodynamics-based Flux Analysis, Bioinformatics 35 (2019) 167-169

[13] T. Jack, Computerised Calculations of Thermo-Physical Steam and Air Properties, Int J Pure Appl. Sci. Technol. 9 (2012) 84-93

[14]Paoli, L. T., Inguva, P. K., Haslam, A. J., and Walker, P. J., 2024, "Confronting the Thermodynamics Knowledge Gap: A Short Course on Computational Thermodynamics in Julia," Education for Chemical Engineers, 48, pp. 1-14.

[15]MathWorks, Inc., 2020, "MATLAB Products," Natick, MA. Available: https://www.mathworks.com/products.html

[16] F-Chart Software, 2021, "Engineering Equation Solver," F-Chart Software. Retrieved September 2021. Available: https://www.fchart.com/ees/

[17] Van Wylen, Gordon J., and Sonntag, Richard E., "Fundamentals of Classical Thermodynamics",3rd Edition, John Wiley & Sons, 1986

[18] Sharma, G., "MATLAB-Optimization-Thermodynamics," Version 1.0.1, GitHub Repository, 2024, https://github.com/GauravSharma123/MATLAB-Optimization-Thermodynamics, DOI: 10.5281/zenodo.14643116 (accessed January 14, 2024)

[19] Y. A. Çengel, M. A. Boles, and M. Kanoğlu, Thermodynamics:An engineering approach, McGraw-Hill Education, 2019.

Appendix: EES Code for Sample Problem

```
//Tank A
T_A1=80[F]
x_A1 = 1 //saturated vapor
Vol_A=3.2[ft^3]
R$='R134a'
CALL realthermoprops(R$, T=T_A1, x=x_A1: T_a1,P_a1,v_a1,h_a1,s_a1,u_a1)
P_A2=23.85[lbf/in^2]
T_A2=80[F]
CALL realthermoprops(R$, T=TA2, P=P_A2: T_a2,P_a2,v_a2,h_a2,s_a2,u_a2)

//average for tank A
h_ao=(h_a1+h_a2)/2
s_ao=(s_a1+s_a2)/2


//Tank B
T_B=-40[F]
x_B = 0.01
Vol_B=2.5[ft^3]
CALL realthermoprops(R$, T=T_B, x=x_B: T_b1,P_b1,v_b1,h_b1,s_b1,u_b1)

P_B2=23.85[lbf/in^2]
x_g=1
x_f=0
CALL realthermoprops(R$, P=P_B2, x=x_g: T_gb2,P_gb2,v_gb2,h_gb2,s_gb2,u_gb2)
CALL realthermoprops(R$, P=P_B2, x=x_f: T_fb2,P_fb2,v_fb2,h_fb2,s_fb2,u_fb2)

//Fluid Leaving Tank B
P_Leave=20[lbf/in^2]
x_Leave=0
CALL realthermoprops(R$, P=P_Leave, x=x_Leave: T_Leave,P_Leave,v_Leave,h_Leave,s_Leave,u_Leave)

//Definitions
v_bfg2=v_gb2 - v_fb2
s_bfg2=s_gb2 - s_fb2
u_bfg2=u_gb2 - u_fb2

//Mass Equations
m_a1=Vol_A/v_a1
m_a2=Vol_A/v_a2

m_b1=Vol_B/v_b1

m_ao=m_a1-m_a2
-m_bo+m_ao=m_b2-m_b1

//Energy Equations
Q_a=m_a2*u_a2 - m_a1*u_a1 + m_ao*h_ao

//Entropy Equations
S_gena = Q_a/(Converttemp('F','R',T_A1)) + m_ao*s_ao - m_a2*s_a2 + m_a1*s_a1
S_genb = -m_ao*s_ao+m_bo*s_Leave+m_b2*s_b2-m_b1*s_b1
s_b2 = s_fb2+x_b2*s_bfg2

//Non Linear Equations
u_b2=u_fb2+x_b2*u_bfg2
m_b2=Vol_B/(v_fb2+x_b2*v_bfg2)
m_b2*u_b2-m_b1*u_b1=m_ao*h_ao-h_Leave*m_bo
```

Figure A.1: EES solution for the same problem.