Extension of Generalized Fluid System Simulation Program's Fluid Property Database

Kishan Patel

NASA – Marshall Space Flight Center
ER43: Thermal & Combustion Analysis Branch
Student Intern from Mississippi State University

Abstract

This internship focused on the development of additional capabilities for the General Fluid Systems Simulation Program (GFSSP). GFSSP is a thermo-fluid code used to evaluate system performance by a finite volume-based network analysis method. The program was developed primarily to analyze the complex internal flow of propulsion systems and is capable of solving many problems related to thermodynamics and fluid mechanics. GFSSP is integrated with thermodynamic programs that provide fluid properties for sub-cooled, superheated, and saturation states. For fluids that are not included in the thermodynamic property program, look-up property tables can be provided. The look-up property tables of the current release version can only handle sub-cooled and superheated states. The primary purpose of the internship was to extend the look-up tables to handle saturated states. This involves a) generation of a property table using REFPROP, a thermodynamic property program that is widely used, and b) modifications of the Fortran source code to read in an additional property table containing saturation data for both saturated liquid and saturated vapor states. Also, a method was implemented to calculate the thermodynamic properties of user-fluids within the saturation region, given values of pressure and enthalpy. These additions required new code to be written, and older code had to be adjusted to accommodate the new capabilities. Ultimately, the changes will lead to the incorporation of this new capability in future versions of GFSSP. This paper describes the development and validation of the new capability.

Nomenclature

c_p	=	specific heat at constant pressure (Btu/lbm-R)
c_{pL}	=	saturated liquid specific heat at constant pressure (Btu/lbm-R)
c_{pV}	=	saturated vapor specific heat at constant pressure (Btu/lbm-R)
h	=	enthalpy (Btu/lbm)
h_L	=	saturated liquid enthalpy (Btu/lbm)
h_V	=	saturated vapor enthalpy (Btu/lbm)
k	=	thermal conductivity (Btu/ft-s-R)
k_L	=	saturated liquid thermal conductivity (Btu/ft-s-R)
k_V	=	saturated vapor thermal conductivity (Btu/ft-s-R)
$K_{rotation}$	=	rotational K-factor
L/d	=	length-to-diameter ratio
$\dot{m}$	=	mass flow rate (lbm/s)
P	=	pressure (psia)
P_r	=	reduced pressure
P_{sat}	=	saturated pressure (psia)
P_t	=	throat pressure (psia)
s	=	entropy (Btu/lbm-R)
s_L	=	saturated liquid entropy (Btu/lbm-R)
s_V	=	saturated vapor entropy (Btu/lbm-R)
T	=	temperature (°F)
T_r	=	reduced temperature
T_{sat}	=	saturated temperature (°F)
T_t	=	throat temperature (°F)
V_t	=	throat velocity (ft/s)
x	=	vapor quality

<u>Greek</u>

γ	=	specific heat ratio
γ_L	=	saturated liquid specific heat ratio
γ_V	=	saturated vapor specific heat ratio
μ	=	viscosity (lbm/ft-s)
μ_L	=	saturated liquid viscosity (lbm/ft-s)
μ_V	=	saturated vapor viscosity (lbm/ft-s)
ρ	=	density (lbm/ft^3)
ρ_L	=	saturated liquid density (lbm/ft^3)
ρ_V	=	saturated vapor density (lbm/ft^3)

I. Introduction

A NALYSIS of virtual thermo-fluid systems plays a large role in the modern aerospace industry. As engineering systems have become more complex, demand for reliable and accurate virtual models has risen. Computer programs that create and analyze these models must be effective enough to examine a variety of fluids under a wide range of scenarios. Programs must be capable of calculating a number of fluid properties at different times and locations in the model. The General Fluid System Simulation Program (GFSSP) was created to meet this demand. At a time when several specific codes capable of analyzing a unique system existed, GFSSP was introduced as a modular program capable of analyzing general fluid flows. A number of fluids can be used in GFSSP at different states. Users of the program are also able to create their own fluids through the use of property tables. However, GFSSP is unable to analyze user-fluids undergoing phase-change or existing as two-phase mixtures. This deficiency limits the number of fluids that may be used and the types of scenarios that may be analyzed.

II. Background

A. What is GFSSP?

The Generalized Fluid System Simulation Program, or GFSSP, is a general-purpose computer program to compute pressure, temperature, flow rate, and flow distribution in a flow network based on time. Originally, the program was developed to analyze the internal flow of a turbopump and the transient flow of a propulsion system. However, the program was designed to be general so that it could be used to analyze a wide array of scenarios. GFSSP can model many types of fluids under a variety of conditions, such as phase-change, compressibility, conjugate heat transfer, and external body forces. The system makes use of a combination of nodes, branches, and conductors to model a thermo-fluid scenario. Scalar properties, such as pressure, temperature, and specie concentrations, are calculated at nodes while mass flow rates and heat transfer rates are calculated at branches and conductors, respectively. GFSSP makes use of a graphical user interface called Visual Thermofluid Analyzer of Systems and Components, or VTASC, to build the network of nodes, branches, and conductors in a user-friendly environment. Users are also able to use VTASC to run completed models and view results. Thermodynamic data for the fluids are derived from two programs that are integrated into GFSSP: GASP/WASP and GASPAK. These two databases allow users to pick from 35 fluids to use in GFSSP. Also, twenty-one resistance options are available to define branches as sources or sinks of momentum, such as pumps, pipes, orifices, and parallel plates. GFSSP utilizes a combination of equations to solve models. These include a finite volume formulation of mass, momentum, and energy conservation equations with thermodynamic equations of state for real fluids and energy conservation equations for solids. This system of equations is solved by the combination of the Newton-Raphson method and the successive substitution methods. [3]

B. Description of GFSSP Thermodynamic Databases and Fluid Properties

Performing calculations on various fluids requires basic thermodynamic property values that must be readily available. GFSSP is integrated with a collection of thermodynamic property databases, including GASP, WASP, and GASPAK. These databases are computer programs used to calculate the properties of various fluids at different states. In GFSSP, the user has two options when selecting a fluid. The user may either choose a fluid from a combined database of GASP and WASP or from GASPAK alone. With the GASP/WASP option, the user has access to 12 fluids, which can be seen in Table 1. These fluids are common to many thermodynamic and fluid mechanic systems.

Table 1. GASP/WASP Fluids

Index	Fluid
1	helium
2	methane
3	Neon
4	nitrogen
5	carbon monoxide
6	oxygen
7	argon
8	carbon dioxide
9	fluoride
10	hydrogen
11	water
12	RP-1

The GASPAK option provides the user with 35 fluids, which can be seen in Table 2. GASPAK contains more fluids that are common to everyday appliances, such as refrigerants and air. The first twelve fluids in the database are common to both GASP/WASP and GASPAK.

Table 2. GASPAK Fluids

Index	Fluid	Index	Fluid
1	helium	19	krypton
2	methane	20	propane
3	neon	21	xenon
4	nitrogen	22	R-11
5	carbon monoxide	23	R12
6	oxygen	24	R122
7	argon	25	R32
8	carbon dioxide	26	R123
9	parahydrogen	27	R124
10	hydrogen	28	R125
11	water	29	R134A
12	RP-1	30	R152A
13	isobutane	31	nitrogen trifluoride
14	butane	32	ammonia
15	deuterium	33	ideal gas
16	ethane	34	air
17	ethylene	35	hydrogen peroxide
18	hydrogen sulfide		

Both GASP/WASP and GASPAK provide thermodynamic data for sub-cooled liquid, superheated vapor, and liquid-vapor saturation states. The use of fluids that are common between the two databases should provide similar, if not identical, results.

For fluids that are not found in GASP/WASP or GASPAK, the user may create a customized fluid for use in a GFSSP model. Creation of these user-fluids requires seven property tables in the form of data files along with the

molecular weight. The seven property tables are unique to the fluid and contain thermodynamic data for a specific property based on varying values of temperature (T) and pressure (P). The seven properties are listed below. To achieve accurate results, each value in the property tables must have the values listed in parenthesis. Also, the temperature and pressure must be in units of R and psia, respectively.

- Thermal conductivity, k (Btu/ft-s-R)
- Density, ρ (lbm/ft^3)
- Viscosity, μ (lbm/ft-sec)
- Specific heat ratio, γ (dimensionless)
- Enthalpy, h (Btu/lbm)
- Entropy, s (Btu/lbm-R)
- Specific heat, c_p (Btu/lbm-R)

Currently, the user is able to create and customize three user-fluids. Figure 1 below is a screenshot of the fluid options panel in VTASC. The customizable user-fluids are labeled as "Fluid 1," "Fluid 2," and "Fluid 3" in the fluid library. The red oval in Figure 1 shows where the user can enter the file names of the seven property tables and the molecular weight. Every property file must be located in the working directory.
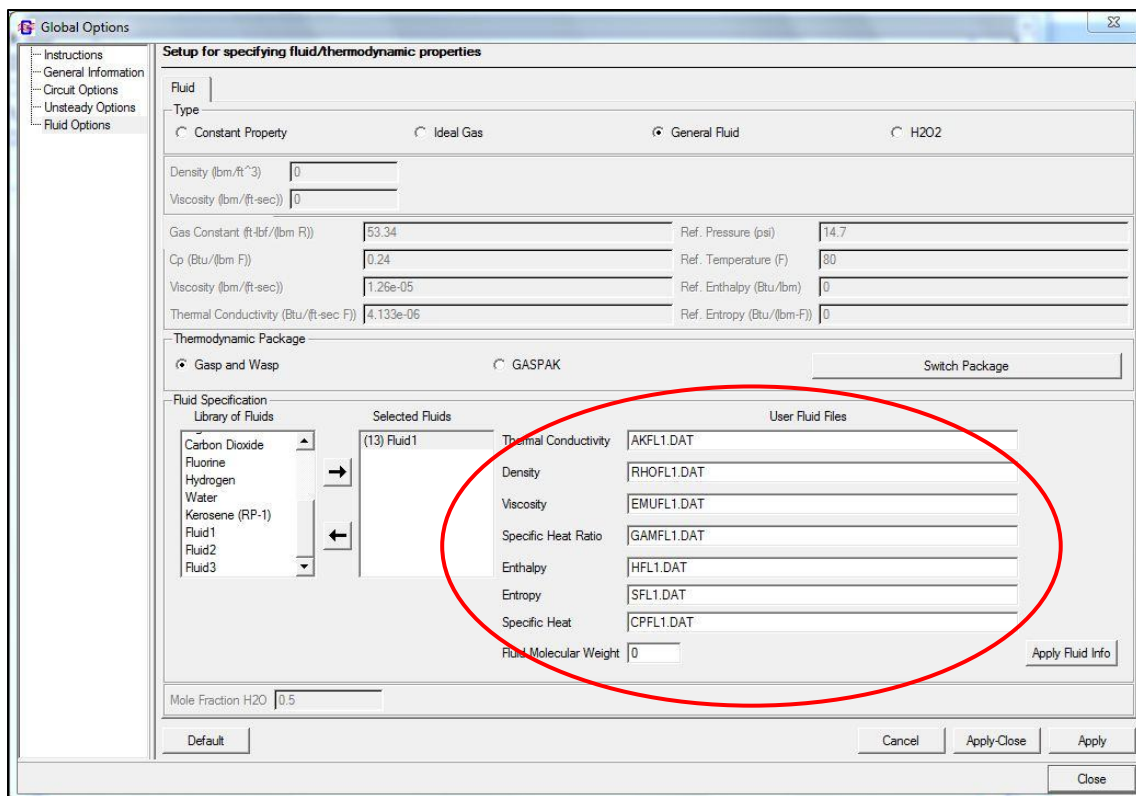
Figure 1. Fluid options panel in VTASC

III. Problem

Depending on the conditions of its environment, fluids can exist at different states. Figure 2 below shows a temperature-volume graph for a general fluid. The dome on the graph represents the separation between the various states. Sub-cooled liquids exist to the left of the dome while superheated vapors exist to the right of the dome. The region underneath the dome represents the saturation region where two-phase mixtures exist.
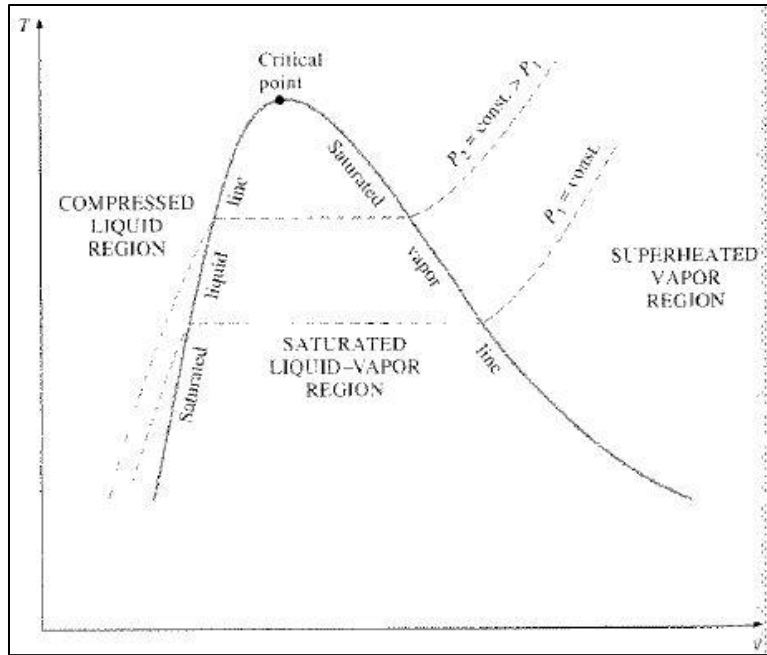
Figure 2. Saturation dome on a T-v diagram [1]

For fluids that are defined by GASP/WASP or GASPAK, all three regions can be analyzed: sub-cooled, superheated, and saturated. For user-defined fluids, however, only the sub-cooled and superheated regions can be analyzed. In the current version of GFSSP, there is no provision to analyze the saturated region of user-fluids. This flaw limits the applicability of the program for fluids not built into GFSSP and restricts the types of scenarios that may be modeled.

IV. Solution

In order to solve the problem, two approaches were taken. First, an attempt was made to use generalized charts for compressibility, enthalpy, and entropy in order to calculate the thermodynamic properties of a saturated fluid. This method proved to be inadequate for liquid-vapor mixtures, and thus, was discontinued. The second attempt made use of saturation tables that were unique to a given fluid. The development and results of both methods can be seen below.

A. Generalized Charts

The first solution made use of three different generalized charts: compressibility, enthalpy and entropy. Each of the charts presents data based on two independent variables: reduced pressure (P_r) and reduced temperature (T_r). Generalized compressibility charts are designed to calculate the compressibility while the generalized enthalpy and entropy charts represent a substance's departure from its ideal values of enthalpy and entropy, respectively. An example of an enthalpy departure chart can be seen in Figure 3 below.
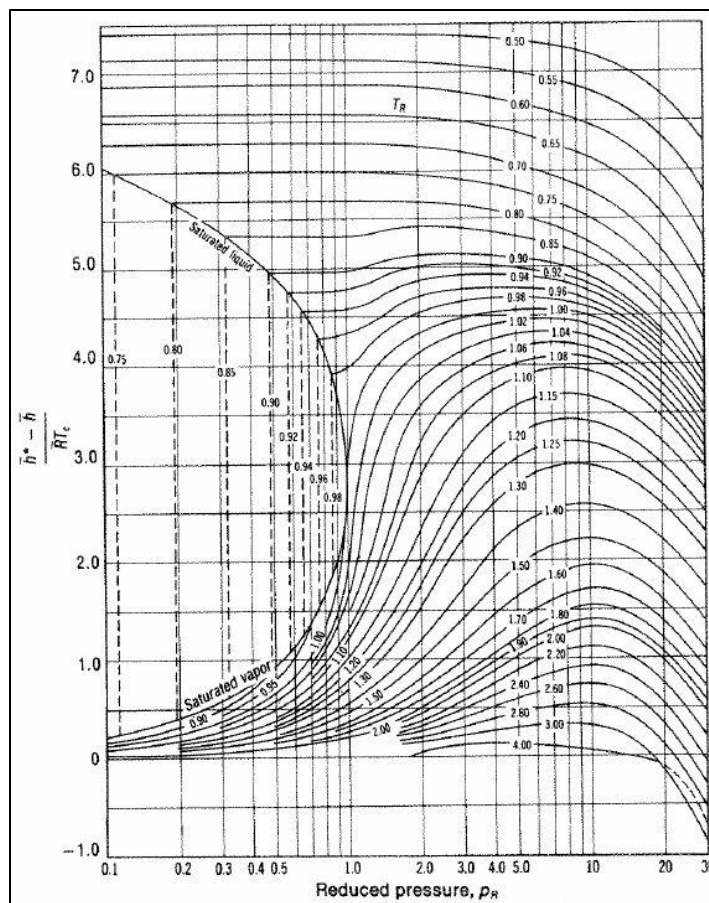
Figure 3. Generalized enthalpy correction chart [2]

For user-defined fluids in GFSSP, the goal was to use these tables to calculate thermodynamic properties, such as density and entropy. A tabulated form of the generalized charts could have been integrated into the source code, and the user would only have to provide the critical pressure and critical temperature of the fluid. Once the enthalpy had been calculated from GFSSP's enthalpy subroutine, the generalized enthalpy chart could have been used to measure the vapor quality (x), provided that the substance is a two-phase mixture. This quality combined with the generalized entropy chart could have been used to the measure the entropy departure, which in turn could have been used to calculate the actual entropy of the substance. From the generalized compressibility chart, the compressibility can be identified, which can then used with the ideal gas law to calculate the density. To calculate the remaining thermodynamic properties, which include specific heat (cp), specific heat ratio (γ), thermal conductivity (k), and viscosity (μ), additional correlations would have to have been added to the code.

Example problems were worked by hand to test the accuracy of this method. Unfortunately, the generalized charts proved ineffective within the saturation zone. Further research revealed that much of the charts' data is accurate for the vapor region, but not enough data is given for the liquid and saturated regions. The generalized charts are used primarily when working with highly compressed gas where the ideal case fails to be accurate, but the charts do not apply to liquids or liquid-vapor mixtures. This inadequacy resulted in the cancellation of the generalized chart approach.

B. Saturation Property Tables

After the failure of the generalized charts, a second approach was developed that made use of saturation tables. These saturation tables were intended to serve as the eighth property table that the user must provide when defining a fluid. During the development and testing process, property tables and saturation tables were created for water and tested in GFSSP against the properties for water from WASP. Four basic objectives were created to complete this method:

- Generate seven property tables for water
- Create an eighth property table containing saturation data for water
- Modify Fortran source code to read in additional table
- Implement method of calculating thermodynamic properties of user-fluids within saturation region, given pressure and enthalpy

1. Generation of Seven Property Tables

A thermodynamic property program called REFPROP was used to develop the values for the seven property tables. REFPROP is a popular program in the thermo-fluid industry and is published by the National Institute of Standards and Technology (NIST). A single table was created in REFPROP Version 8.0 that contained all the data needed for the seven property tables. This large table was saved as a data file in a *.dat* format. A simple Fortran script, found in Appendix A, was used to read the REFPROP data table and create seven property tables in a format that could be read by GFSSP. A series of inputs had to be made in order to create the tables properly. In order for the reader to be able to recreate the tables, the process will be explained in detail. If one wishes to use the Fortran script in Appendix A, the variables must be redimensioned to accommodate the proper number of temperatures and pressures. Also, the file input name must be adjusted to the read the proper REFPROP data table.

Before the data table was created in REFPROP, the program's options were adjusted. The following panels can be found under the "Options" tab on the menu bar.

- Units: The units of the values appearing in the REFPROP table can be selected in this panel. The units of all nine dimensions were set to British Engineering units by simply clicking on the "English" button in the "Reset Units" list. Six of these units match with the required units for GFSSP, as discussed in Part B above. Thermal conductivity, however, is in units of Btu/hr-ft-R when it should be in units of Btu/sec-ft-R. This minor issue is fixed in the Fortran script found in Appendix A.
- Properties: The properties that are to appear in the REFPROP table are selected in this panel. Nine properties were selected: temperature, pressure, density, enthalpy, entropy, specific heat, specific heat ratio, thermal conductivity, and viscosity. The selected properties can be seen in Figure 4 below. Seven of the properties are found in the "Thermodynamic" tab, as seen in the top panel. The remaining properties, k and μ, are found in the "Transport, Misc." tab, as seen in the bottom panel of Figure 4.
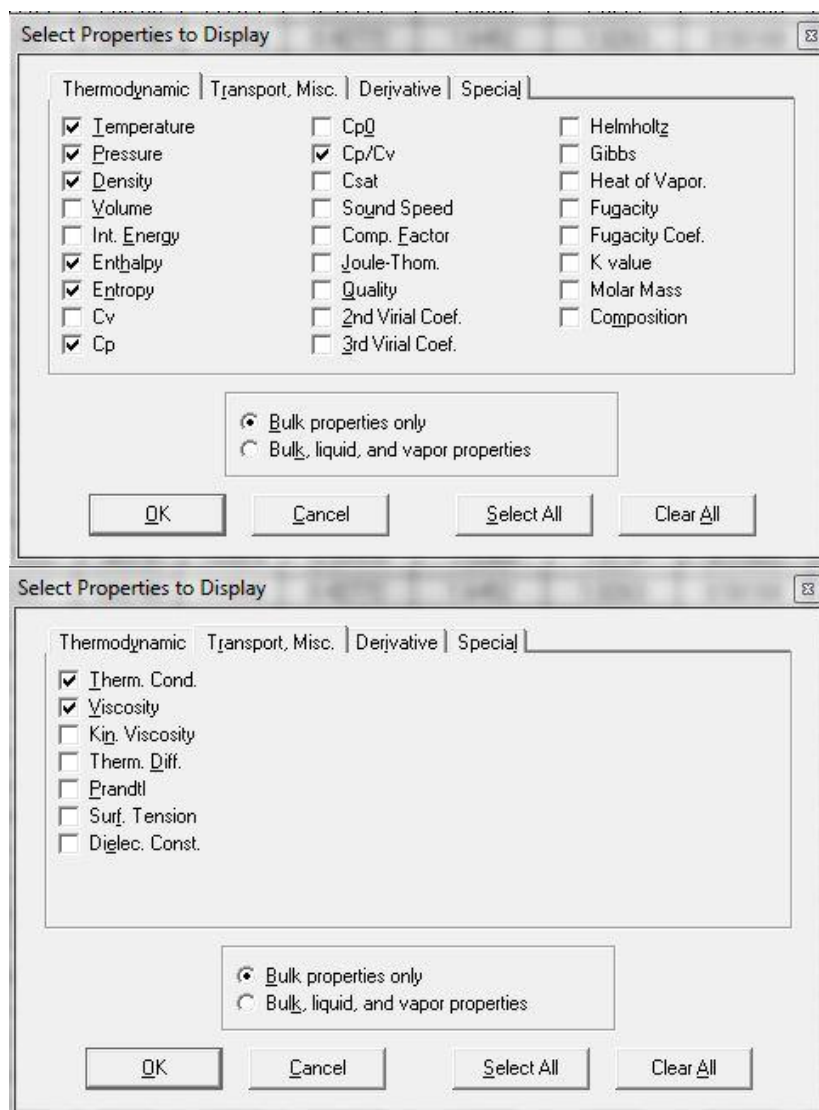
Figure 4. REFPROP property dialogue panels

- Property Order: The order in which the properties are listed in the REFPROP table is determined in the Property Order panel. The properties were rearranged to appear in the following order: T, P, ρ, h, s, c_p, γ, k, and μ.

After the basic options were edited, the substance was chosen by opening the "Pure Fluid" panel under the "Substance" tab on the menu bar. Here, water was chosen as the fluid. To generate the values for the REFPROP table, the "Isoproperty Tables" option was selected under the "Calculate" tab on the menu bar. The temperature was varied while the pressure remained constant. On the following panel, the constant pressure was specified and the parameters for the temperature variation were inputted. The Isoproperty Table option was repeated for steadily increasing values of pressure while keeping the same temperature variation. However, each subsequent run was added to the first table by selecting "Add to current table," as highlighted in Figure 5 below.
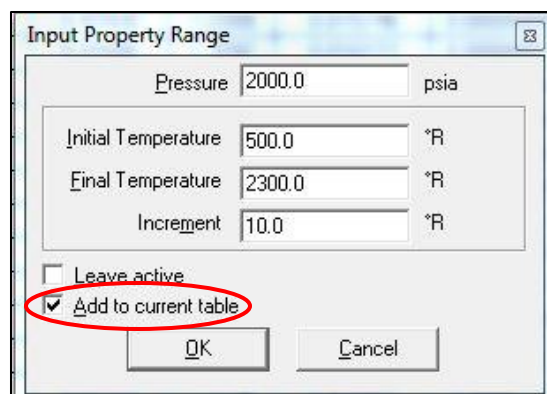
Figure 5. Addition of isoproperty table values to first table

By default, an empty row is created between each run of the Isoproperty Table option to delineate different constant pressures. In order for the data to be read correctly, all of the empty rows were deleted so that there were no spaces between adjacent blocks of data. When the Isoproperty Table option had been run for the desired temperature range, the REFPROP table was saved as a data file by selecting "Save Table" under the "File" tab on the menu bar. Care was taken to minimize the extra information, such as heading labels and row numbers by deselecting the options, as seen in Figure 6. Also, spaces were set as the delimiter for the data file and a file name was chosen.
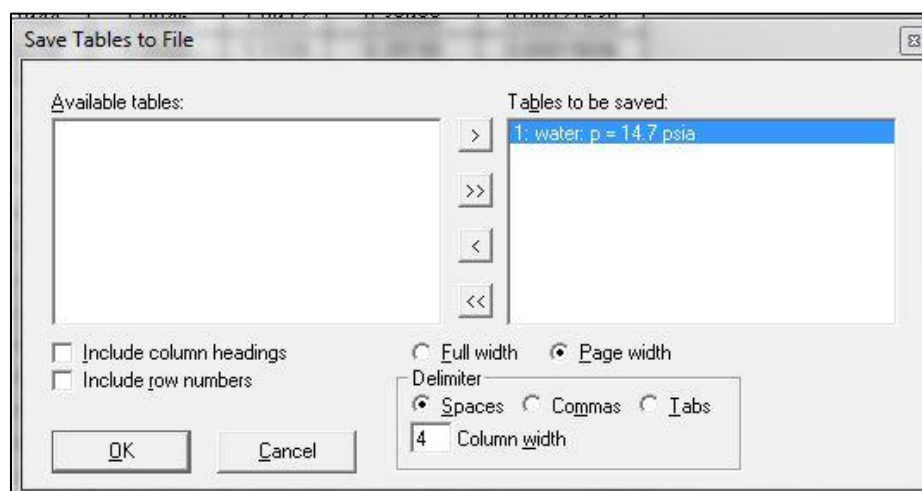
Figure 6. Save panel with options

Upon creating saving the REFPROP table, the data file was modified so that the number of rows of data appeared in the first row of the file. Immediately following the first row were the nine columns of data for temperature, pressure, and the main seven properties. The Fortran script found in Appendix A was run to take the REFPROP data file, called REFPROP.dat, and separate its data into seven different property files used to define user-fluids in GFSSP. The format of the property tables is very important when GFSSP is reading in the data. Figure 7 below is an excerpt from a c_p property table with annotations to explain the significance of each number. As seen in the figure, the number of pressure points and the number of temperatures appear in the first row. In the next line, each of the temperature values is listed in ascending order; in this case 30 temperatures are given. Afterwards, 25 blocks of data appear. The first number in the block is the pressure value. Immediately following are 30 c_p values that correspond to the 30 temperatures at the specified pressure.

Figure 7. Format for property tables of user-fluids

While using this approach to solve the saturated user-fluid problem, two sets of property tables were created for water. The first set was coarser and contained fewer data channels while the second set was finer and contained more data channels. Two sets were created in order to determine if using greater amounts of data channels would provide more accurate answers. The coarse property tables contained temperatures from 500 R to 1625 R with an increment of 15 R while the pressures ran from 14.7 psia to 1264.7 psia at 50 psia increments. This resulted in 76 temperatures and 26 pressures in the data files. On the other hand, the fine property tables contained temperatures running from 500 R to 1700 R at 10 R increments and pressures from 10 psia to 1310 psia at increments of 20 psia, which gave 121 temperatures and 66 pressures in the data files. These water property tables were validated by being used in simple models and comparing the results with water from WASP.

2. Creation of Saturation Table

After the basic seven property tables were created for water, a saturation table was created with REFPROP. This saturation table served as an eighth property table for user-fluids and represented data from the saturation region of water. The table contained values for saturation pressure (P_{sat}) and saturation temperature (T_{sat}) as well as values for the seven thermodynamic properties at both the saturated liquid and saturated vapor states:

- Density (ρ_L & ρ_V)
- Enthalpy (h_L & h_V)
- Entropy (s_L & s_V)
- Specific heat (c_{pL} & c_{pV})
- Specific heat ratio (γ_L & γ_V)
- Thermal conductivity (k_L & k_V)
- Viscosity (μ_L & μ_V)

Similar to the property tables, a simple Fortran script, found in Appendix B, was used to read the REFPROP saturation table and create a saturation property file that could be read by GFSSP. The process of creating this saturation table will be described in detail so one can recreate the file.

The options settings in REFPROP were kept the same as the options for the seven property files. Details for these options can be seen in Part 1 above. In REFPROP, the saturation table was created by selecting "Saturation Tables" under the "Calculate" tab. This opens a panel, as seen in Figure 8, where the characteristics of the saturation table can be specified, such the type of saturation and the varying property.
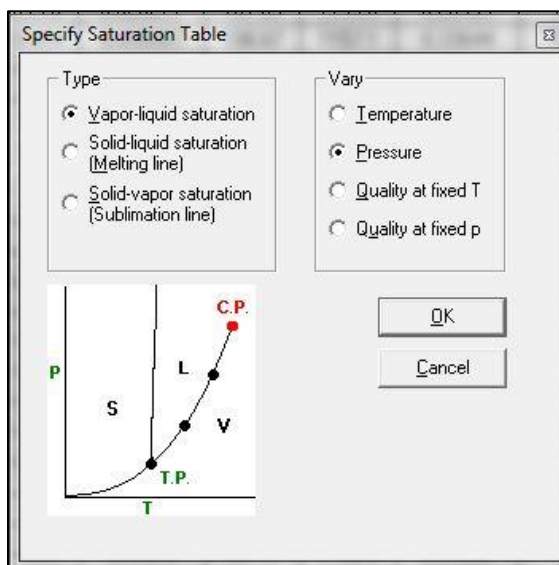

Figure 8. Saturation table specifications panel

Vapor-liquid saturation was selected in the specifications panel and pressure was set as the property to vary. In the following panel, the range and increment of temperature was inputted, and REFPROP generated the corresponding saturation table, which can partially be seen in Figure 9. The REFPROP saturation table gave 16 columns data in the following order: T_{sat}, P_{sat}, ρ_L, ρ_V, h_L, h_V, s_L, s_V, C_{pL}, C_{pV}, γ_L, γ_V, k_L, k_V, μ_L, and μ_V.

	Temperature (°R)	Pressure (psia)	Liquid Density (lbm/ft³)	Vapor Density (lbm/ft³)	Liquid Enthalpy (Btu/lbm)	Vapor Enthalpy (Btu/lbm)	Liquid Entropy (Btu/lbm-°R)	Vapor Entropy (Btu/lbm-°R)	Liquid Cp (Btu/lbm-°R)	Vapor Cp (Btu/lbm-°R)	Liquid Cp/Cv	Vapor Cp/Cv	Liquid Therm. Cond. (Btu/h-ft-°F)	Vapor Therm. Cond. (Btu/h-ft-°F)	Liquid Viscosity (lbm/ft-s)	Vapor Viscosity (lbm/ft-s)
1	494.67	0.10000	62.421	0.00033957	3.0109	1077.2	0.0061045	2.1777	1.0073	0.45077	1.0002	1.3284	0.32620	0.0099206	0.0011372	0.0000062193
2	653.30	10.100	60.270	0.026271	161.84	1144.0	0.28453	1.7879	1.0050	0.48811	1.1004	1.3338	0.39036	0.013875	0.00021176	0.0000080113
3	687.85	20.100	59.414	0.050007	196.67	1157.1	0.33644	1.7327	1.0103	0.50658	1.1357	1.3404	0.39401	0.015105	0.00017284	0.0000084515
4	710.16	30.100	58.806	0.072968	219.27	1165.0	0.36874	1.7004	1.0148	0.52221	1.1603	1.3465	0.39508	0.015982	0.00015402	0.0000087382
5	727.04	40.100	58.319	0.095462	236.45	1170.7	0.39261	1.6775	1.0189	0.53632	1.1798	1.3523	0.39530	0.016690	0.00014216	0.0000089557
6	740.78	50.100	57.906	0.11764	250.50	1175.1	0.41171	1.6598	1.0227	0.54939	1.1963	1.3579	0.39514	0.017296	0.00013372	0.0000091329
7	752.46	60.100	57.542	0.13957	262.49	1178.6	0.42772	1.6452	1.0263	0.56168	1.2108	1.3634	0.39477	0.017832	0.00012728	0.0000092835
8	762.67	70.100	57.214	0.16132	273.01	1181.6	0.44155	1.6329	1.0298	0.57334	1.2238	1.3687	0.39429	0.018316	0.00012212	0.0000094150
9	771.78	80.100	56.915	0.18293	282.41	1184.2	0.45377	1.6222	1.0331	0.58447	1.2357	1.3740	0.39372	0.018760	0.00011786	0.0000095323
10	780.01	90.100	56.639	0.20442	290.95	1186.4	0.46474	1.6127	1.0363	0.59516	1.2467	1.3792	0.39311	0.019173	0.00011425	0.0000096383
11	787.55	100.10	56.380	0.22582	298.79	1188.3	0.47469	1.6042	1.0394	0.60546	1.2570	1.3843	0.39246	0.019559	0.00011114	0.0000097352
12	794.51	110.10	56.138	0.24714	306.05	1190.0	0.48383	1.5964	1.0425	0.61542	1.2667	1.3894	0.39178	0.019923	0.00010841	0.0000098247
13	800.98	120.10	55.908	0.26839	312.83	1191.6	0.49229	1.5893	1.0455	0.62507	1.2759	1.3945	0.39109	0.020268	0.00010599	0.0000099080
14	807.05	130.10	55.690	0.28960	319.19	1192.9	0.50016	1.5828	1.0484	0.63446	1.2847	1.3995	0.39039	0.020597	0.00010382	0.0000099859
15	812.75	140.10	55.481	0.31076	325.19	1194.2	0.50753	1.5767	1.0513	0.64360	1.2931	1.4045	0.38968	0.020911	0.00010186	0.000010059
16	818.14	150.10	55.282	0.33189	330.89	1195.3	0.51447	1.5710	1.0541	0.65253	1.3012	1.4095	0.38896	0.021213	0.00010010	0.000010128
17	823.26	160.10	55.090	0.35299	336.30	1196.3	0.52103	1.5657	1.0569	0.66125	1.3091	1.4144	0.38824	0.021503	0.000098404	0.000010194
18	828.13	170.10	54.905	0.37407	341.47	1197.2	0.52725	1.5606	1.0597	0.66979	1.3166	1.4194	0.38752	0.021784	0.000096931	0.000010257
19	832.78	180.10	54.726	0.39513	346.42	1198.1	0.53317	1.5559	1.0624	0.67817	1.3240	1.4243	0.38680	0.022055	0.000095535	0.000010317
20	837.24	190.10	54.553	0.41619	351.17	1198.9	0.53882	1.5513	1.0651	0.68640	1.3311	1.4292	0.38607	0.022318	0.000094235	0.000010374
21	841.51	200.10	54.385	0.43724	355.74	1199.6	0.54422	1.5470	1.0678	0.69449	1.3381	1.4341	0.38535	0.022573	0.000093021	0.000010429
22	845.62	210.10	54.222	0.45828	360.15	1200.2	0.54941	1.5429	1.0705	0.70246	1.3448	1.4390	0.38463	0.022822	0.000091882	0.000010482
23	849.58	220.10	54.063	0.47933	364.41	1200.8	0.55439	1.5389	1.0731	0.71031	1.3515	1.4440	0.38390	0.023064	0.000090812	0.000010533
24	853.40	230.10	53.909	0.50038	368.52	1201.4	0.55918	1.5351	1.0757	0.71805	1.3580	1.4489	0.38318	0.023300	0.000089802	0.000010582
25	857.10	240.10	53.758	0.52144	372.52	1201.9	0.56381	1.5315	1.0783	0.72569	1.3643	1.4538	0.38246	0.023531	0.000088848	0.000010630
26	860.67	250.10	53.610	0.54251	376.39	1202.4	0.56828	1.5280	1.0809	0.73325	1.3706	1.4587	0.38174	0.023756	0.000087942	0.000010676
27	864.14	260.10	53.466	0.56359	380.15	1202.8	0.57260	1.5246	1.0835	0.74072	1.3767	1.4636	0.38103	0.023977	0.000087082	0.000010721
28	867.51	270.10	53.325	0.58469	383.81	1203.2	0.57679	1.5213	1.0861	0.74811	1.3828	1.4685	0.38031	0.024194	0.000086263	0.000010765
29	870.78	280.10	53.186	0.60580	387.38	1203.5	0.58085	1.5181	1.0886	0.75544	1.3887	1.4734	0.37960	0.024406	0.000085482	0.000010807
30	873.96	290.10	53.051	0.62693	390.85	1203.9	0.58480	1.5151	1.0912	0.76269	1.3946	1.4784	0.37889	0.024615	0.000084735	0.000010848
31	877.05	300.10	52.918	0.64809	394.25	1204.1	0.58863	1.5121	1.0937	0.76989	1.4004	1.4833	0.37818	0.024820	0.000084020	0.000010889
32	880.07	310.10	52.787	0.66926	397.56	1204.4	0.59236	1.5092	1.0963	0.77704	1.4061	1.4883	0.37747	0.025022	0.000083335	0.000010928

Figure 9. Sample saturation table for water in REFPROP

The REFPROP saturation table was saved as a data file in a manner similar to the basic property files. No headings or row numbers were included and spaces were set as the delimiter. The data file was then modified to include the number of data channels in the topmost row while the 16 columns of property values appeared immediately below the first row. In order to make the data easier to handle in GFSSP, a Fortran script was used to rearrange the columns in the data file. This script, as seen in Appendix B, read the saturation table created by REFPROP, called satwaterref.dat, and created a new data file in which the properties appeared in the following order: P_{sat}, T_{sat}, h_L, ρ_L, cp_L, μ_L, γ_L, k_L, s_L, h_V, ρ_V, cp_V, μ_V, γ_V, k_V, and s_V. The first two columns gave the saturated pressure and saturated temperature. The next seven columns gave the seven properties at the saturated liquid state while the final seven columns gave the seven properties at the saturated vapor state.

Like the basic seven property files, two saturation tables were created. One was a coarse table with fewer data channels while the other was a fine table with more data channels. These tables were also used to determine how much the accuracy of the results improved when the detail of the property tables increased. The coarse table featured pressure running from 0.1 psia to 3200 psia at increments of 10 psia, which presented 321 data channels. The fine table, on the other hand, had the pressure running from 0.1 psia to 3202.6 psia at 7.5 psia increments, which created 428 data channels.

3. Modification Fortran Source Code to Read in Additional Table

The addition of a saturation table required new lines of code in the GFSSP source code. Specifically, revisions were made to the FLTABLE subroutine. This subroutine was developed to read in the property tables for user-fluids and determine the property values for fluids outside the saturation region. Additions were made to FLTABLE to read in the eighth property table and to determine whether or not the user-fluid is saturated at a given node. Once the enthalpy is calculated from the ENTHALPY subroutine, the pressure and enthalpy are used to bracket the saturation pressure and determine whether the fluid is saturated. If the fluid is saturated, the quality is calculated and linear interpolation is used to calculate ρ, c_p, μ, γ, k, and s at both the saturated liquid and saturated vapor states. These additions required changes to be made to the rest of the source code. For example, new index variables were added to the common block to accommodate the eighth property table. Also, every call statement to FLTABLE was adjusted to include the new variables added to the subroutine.

4. Implementation of Method to Calculate Thermodynamic Properties in Saturation Region

Once revisions were made to read the saturation table, the actual calculation of saturated properties required more code to be added to the DENSITY subroutine, which was developed to calculate the density at each node. For user-fluids in the saturated region, code was added to assign the thermodynamic property values for saturated liquid and saturated vapor states to the proper variables in the common block. Also quality was used to linearly interpolate the values of ρ, c_p, μ, γ, k, and s at each node.

V. Analysis and Results

Upon implementing the new capability, the fidelity and accuracy of the adjusted program were tested through the use of example problems found within GFSSP's user manual. Dr. Majumdar and other developers of GFSSP created these examples to test and demonstrate the various capabilities of the program. These examples come standard with the installation of GFSSP601 and were modified to show that the property tables and modified code gave comparable results to the standard examples.

Each example was run three times. The first trial was run with the released version of GFSSP601 with water from GASP/WASP; this trial served as the benchmark to which the modified code and property tables were compared. The second and the third trials were run with the modified version of GFSSP601 and water from the REFPROP property tables. In the second trial, the simpler property tables with fewer data points were used while the more detailed property tables were utilized in the third trial. The purpose of using both sets of property tables was to determine whether a greater amount of data points led to a more accurate answer when compared with the benchmark. Specifically, only the examples that contained water as the fluid were examined: Examples 1, 3, 5, and 6. These examples only test the steady-state condition of GFSSP. In order to test the transient condition, Example 8 was modified to use water instead of air as the fluid. In addition, a simple model was created to demonstrate the new capability of analyzing a vapor-liquid mixture of a user-defined fluid. The results of each validation test are discussed below.

A. Example 1: Simulation of a Flow System Consisting of a Pump, Valve, and Pipe Line

The system in Example 1 consists of two water reservoirs connected by a 1500-foot pipe that has a six-inch diameter and a relative roughness of 0.005. The receiving reservoir is located 150 feet higher than the supply reservoir. A pump is responsible for transporting water from the supply reservoir through a valve and up the pipe to the receiving reservoir. A diagram schematic of this system can be seen below in Figure 10.
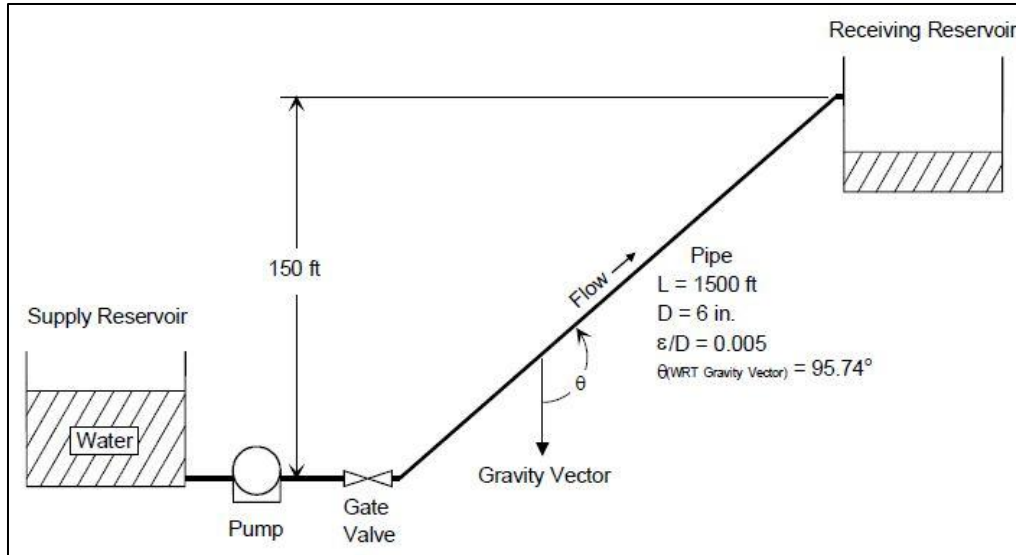
Figure 10. Schematic of Pumping System and Reservoirs [3]

The GFSSP model consists of four nodes and three branches. Nodes 1 and 4 represent the supply and receiving reservoir, respectively. Node 2 is an internal node between the pump outlet and the valve inlet. Node 3 is an internal node representing the outlet of the valve and the inlet of the pipe connected to the receiving reservoir. Branches 12, 23, and 34 represent the pump, gate valve, and pipe, respectively. A diagram of the VTASC model can be seen below in Figure 11.
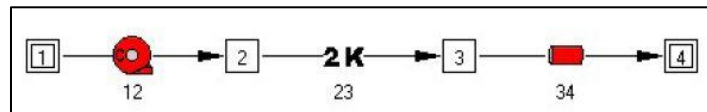

Figure 11. VTASC Model of Example 1 [3]

After the model for Example 1 was modified, GFSSP was run three times. The results of this model can be seen in Table 3 below. Data from the released GFSSP601 are listed as the benchmark. Trial 3 represents the data from the simpler property tables and trial 2 gives data from the detailed property tables.

Table 3. Validation with Example 1

Variable	Benchmark	Trial 1		Trial 2	
		Value	Percent Error	Value	Percent Error
$\dot{m}$ (lbm/sec)	191	191	0.000	191	0.000
P_2 (psi)	229	229	0.000	229	0.000
T_2 (°F)	60.03	60.03	0.000	60.03	0.000

Significant data from Example 1 was examined, including the mass flow rate through the branches and the pressure and temperature at the pump outlet, or node 2. The table shows that the results from the modified code and property tables provide values that match perfectly with the benchmark results.

B. Example 3: Simulation of Compressible Flow in a Converging-Diverging Nozzle

GFSSP's ability to analyze compressible flow is tested in Example 3. This example models a converging-diverging nozzle with choked flow. The nozzle has a length of 6.3 inches and a 0.492-inch throat diameter. The inlet diameter of the nozzle is 0.6758 inches and the throat is located 0.158 inches aft of the inlet. At the inlet, the fluid is considered to be at 150 psia at the inlet while the back pressure set to 60 psia. At both the inlet and outlet, the temperature is set at 1000°F. A schematic for this system can be seen in Figure 12.
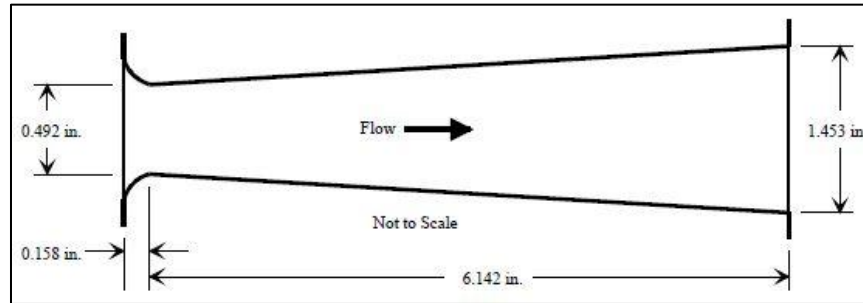

Figure 12. Schematic of Example 3 [3]

The fluid system is modeled in VTASC with 17 nodes and 16 branches. Nodes 1 and 17 are boundary nodes representing the inlet and outlet, respectively. Each of the remaining nodes is an internal node and represented by a closure. Each branch has a constant flow area that represents the area at the midpoint of the branch location. Detailed schematics for the VTASC can be seen in Figure 13.
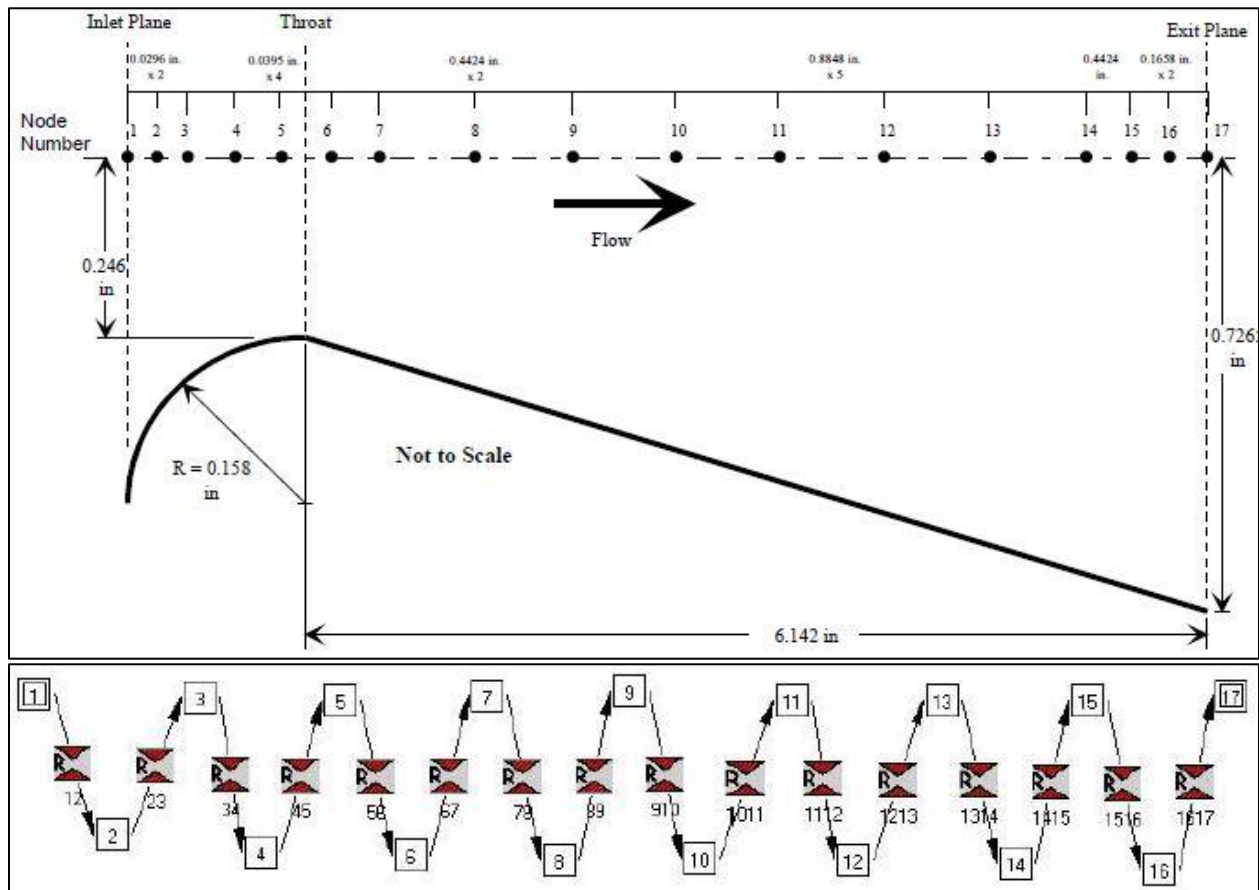

Figure 13. Detailed schematic of converging-diverging nozzle (top) and corresponding VTASC model (bottom) [3]

The model that comes standard with the released version utilizes the second law option to solve the system. Currently, GFSSP601 is not able to use the second-law option for fluids defined by user-generated property tables. Therefore, the standard example had to be modified to use the first-law option, which uses pressure and enthalpy as the state variables. This method gave slightly differently values than the second-law option. For consistency, both the benchmark and trial cases used the first-law method.

Similar to Example 1, the nozzle system was run three times in GFSSP. When comparing the results between the benchmark and the property tables, significant data for a typical nozzle was examined, including the flow rate and the pressure, temperature and velocity at the throat (P_t, T_t, and V_t). P_t and T_t are taken from node 5 while V_t is taken from branch 56. The results can be seen in Table 4 below.

Table 4. Validation with Example 3

Variable	Benchmark	Trial 1		Trial 2	
		Value	Percent Error	Value	Percent Error
$\dot{m}$ (lbm/sec)	0.309	0.31	0.324	0.309	0.000
P_t (psi)	110.7	110.7	0.000	110.8	0.090
T_t (°F)	997.9	991.9	-0.601	994.3	-0.361
V_t (ft/sec)	1830	1820	-0.546	1820	-0.546

Trial 1 shows that the results of the simple property tables are comparable with the benchmark results from GFSSP601. Each of the percent errors are less than 1%. The use of Trial 2 proves that a greater amount of data points helps reduce the error. Both flow rate and throat temperature approached the benchmark as the number of data channels increased. Surprisingly, the percent error for throat pressure increased between Trials 1 and 2. This small increase may simply be the result of rounding errors in the program.

C. Example 5: Simulation of a Flow System Involving a Heat Exchanger

Heat exchangers are common in complex fluid systems. Example 5 tests GFSSP's capability of properly analyze the effects of a heat exchanger. In this simple configuration, a counter flow heat exchanger is developed by using a hot branch and cold branch with flow propagating in opposite directions. As seen in Figure 14, the configuration has hot water at 50 psia and 100°F flowing through a pipe with a length of 10 in and an inner diameter of 0.25 in. The hot water than enters a heat exchanger that is 10 in long and flows through another pipe with a length of 10 in and an inner diameter of 0.25 in. The cold water starts at 50 psia and 60°F, flows through a pipe with a length of 10 in and an inner diameter of 0.5 in. Similar to the hot water, the cold water then enters the 10-inch heat exchanger and then flows through a pipe with a length of 10 in and an inner diameter of 0.5 in. The calculated specific heat values for the hot and cold water are 0.9978 Btu/lbm-R and 1.0014 Btu/lbm-R, respectively.
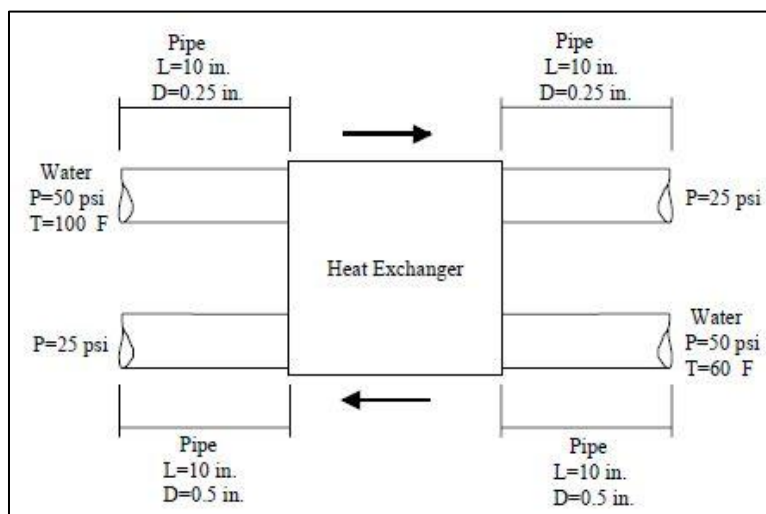

Figure 14. Flow schematic of heat exchanger [3]

In VTASC, the heat exchanger was modeled with eight nodes and six branches, as seen in Figure 15. The hot water flows from left to right at the top of the figure while the cold water flows from right to left at the bottom of the figure. Nodes 1, 4, 5, and 8 are boundary nodes. Nodes 1 and 5 represent the inlets for the hot and cold flows, respectively. Nodes 4 and 8 represent the outlet or downstream flows. Nodes 2 and 6 are internal nodes that represent the inlets of the heat exchanger for hot and cold water, respectively. Similarly, Nodes 3 and 7 are internal nodes that mark the outlets of the heat exchanger for the hot and cold water, respectively. Branches 12, 34, 56, and 78 are the pipes that direct flow into and out of the heat exchanger. Branches 23 and 67 represent the hot and cold sides of the heat exchanger, respectively.
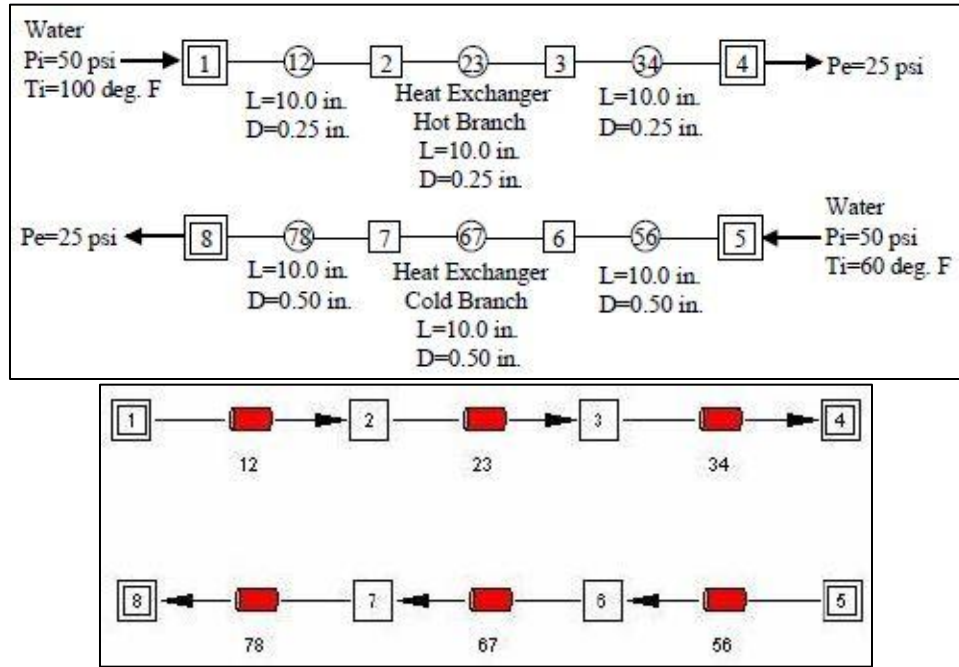

Figure 15. Detailed GFSSP model of heat exchanger (top) and corresponding VTASC model (bottom) [3]

Again, the model was run three times and the results were compared. The values of flow rate for both the hot and cold branches were examined. Also, the temperature and pressure at the outlets of both branches were also compared. The outlet pressure and temperature for the hot branch was taken from Node 3 while the outlet pressure and temperature for the cold branch was taken from Node 7. The results of the models can be seen in Table 5.

Table 5. Validation with Example 5

Variables	Benchmark	Trial 1		Trial 2	
		Value	Percent Error	Value	Percent Error
$\dot{m}_{hot}$ (lbm/sec)	0.885	0.885	0.000	0.885	0.000
$\dot{m}_{cold}$ (lbm/sec)	5.41	5.41	0.000	5.41	0.000
T_{hot} (°F)	72.53	72.55	0.028	72.55	0.028
T_{cold} (°F)	64.53	64.54	0.015	64.54	0.015
P_{hot} (psi)	33.69	33.69	0.000	33.69	0.000
P_{cold} (psi)	33.27	33.27	0.000	33.27	0.000

Trials 1 and 2 both show that the modified code with property tables provides comparable results to the benchmark. The error of the temperatures is less than a tenth of a percent, and the remaining values match perfectly with the values from GFSSP601 with GASP/WASP. The detailed property tables of Trial 2 gave the same results

with the simpler property tables of Trial 1. However, since the errors are so small and near perfection, a noticeable increase in accuracy is not to be expected.

D. Example 6: Radial Flow on a Rotating Radial Disk

GFSSP is also able to analyze the rotational effect on a flow. Example 6 models the flow through a closed impeller. The impeller has water as the fluid and is 11 inches in diameter. An important characteristic of the water in the impeller is the rotational K-factor ($K_{rotation}$) which describes the water's "slip." A higher $K_{rotation}$ results in a greater pressure rises for radial outward flow. For this impeller, the proposed correlations are $K_{front} = 0.8455 - 0.1403 \left(\frac{r-r_i}{r_o-r_i} \right)$ for the front face and $K_{back} = 0.8857 - 0.1762 \left(\frac{r-r_i}{r_o-r_i} \right)$ for the back face. The effects of friction are neglected for this example. Figure 16 shows a schematic of this system.
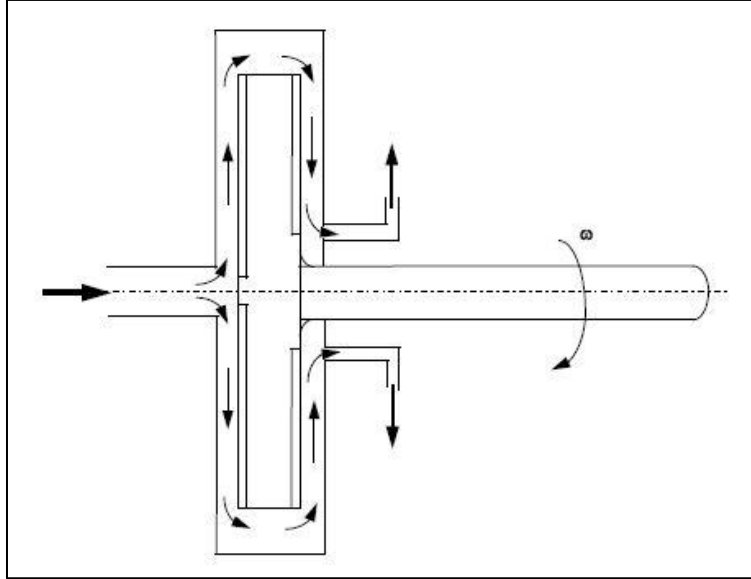

Figure 16. Flow schematic of rotating radial disk [3]

The corresponding GFSSP model contains 13 nodes and 12 branches, as seen in Figure 17. Branches 23, 34, 45, 56, 67, 89, 910, 1011, and 1112 are rotating at 5000 rpm. To find the area of each branch, the average cross sectional area was taken between each node using the equation $A_{ab} = \frac{1}{r_b-r_a} t \int_{r_a}^{r_b} 2\pi r dr$, where A_{ab} is the area of the branch ab, t is the thickness of the impeller in the direction transverse to the flow, and r is the radius.
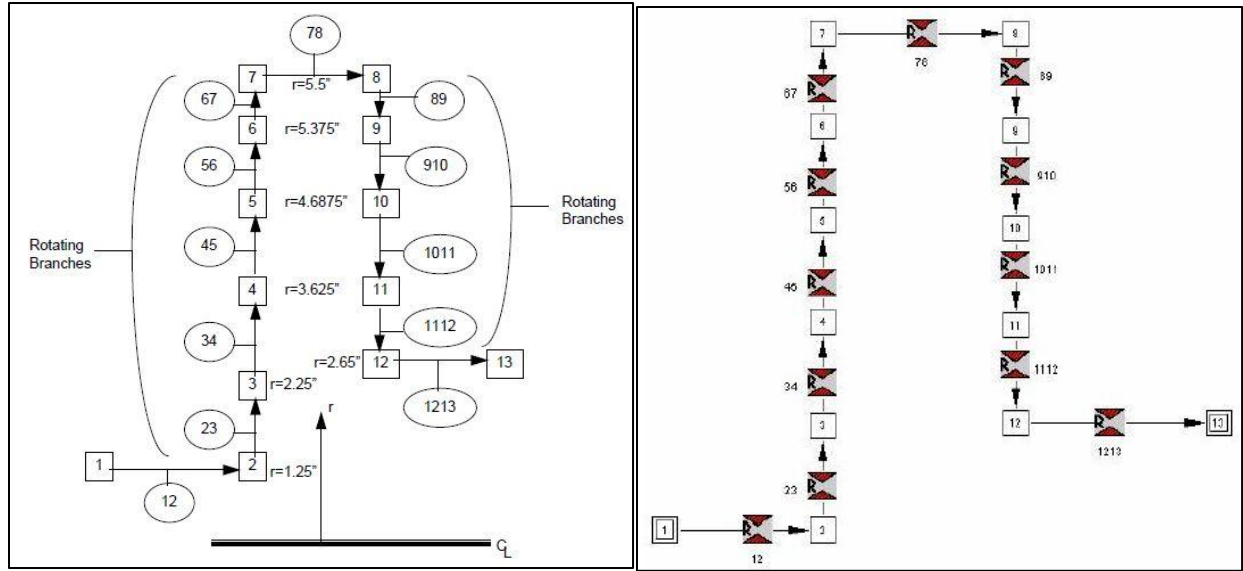
Figure 17. Detailed schematic of Example 6 (left) and corresponding VTASC model (right) [3]

After reviewing and adjusting the model, Example 6 was run three times. The results can be seen in Table 6. Key values that were compared include the flow rate as well as the maximum and minimum values of pressure and temperature. The maximum pressure and temperature occur at node 7, which is the topmost node on the backside of the impeller, and the minimum pressure and temperature are present at node 12, which is the outlet of the rotating disk.

Table 6. Validation with Example 6

Variables	Benchmark	Trial 1		Trial 2	
		Value	Percent Error	Value	Percent Error
$\dot{m}$ (lbm/sec)	7.29	7.29	0.000	7.29	0.000
P_{max} (psi)	313.5	313.4	-0.032	313.4	-0.032
T_{max} (°F)	80.05	80.03	-0.025	80.04	-0.012
P_{min} (psi)	132.8	132.8	0.000	132.8	0.000
T_{min} (°F)	80.01	80	-0.012	80	-0.012

The results of the trials are very promising. Each of the values is very close to its corresponding benchmark, and the greatest error is 0.032%. In Trial 2, the error of T_{max} decreases while the others variables remain the same; this further proves that greater data points increase the accuracy of the property tables.

E. Example 8: Simulation of the Blow Down of a Pressurized Tank

Unsteady flow is very common in real-life applications of fluid mechanics. Example 8 was the first test of the modified code's fidelity in a transient state. The example models a pressurized tank experiencing blow down effects. By default, this example features a tank with an internal volume of 10 ft^3 that is filled with air at 100 psia and 80°F. The pressure within the tank forces air through an orifice that is 0.1 inches in diameter for 200 seconds. To make use of the property tables, the model was modified to feature water as the fluid. The pressure was increased to 1000 psia and the temperature was increased to 1040°F. No changes were made to the tank dimensions or the ambient conditions. A simple schematic of the modified example can be seen in Figure 18.
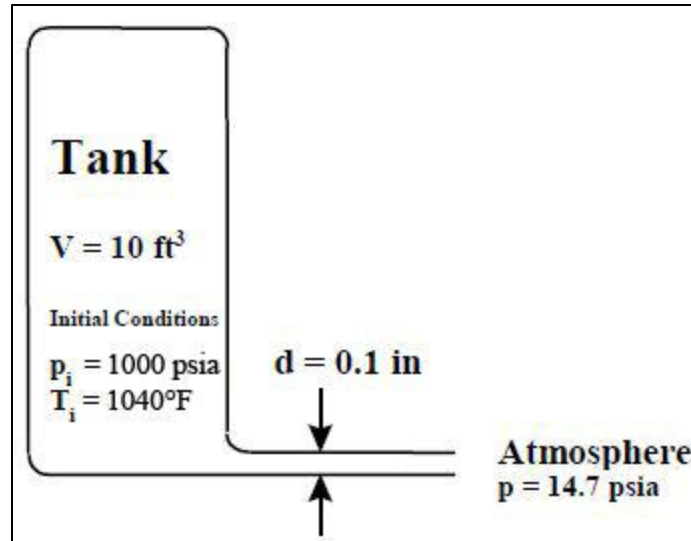
Figure 18. Schematic of modified Example 8 [3]

The venting process is modeled in VTASC with two nodes and a connecting branch, which can be seen in Figure 19 below. Node 1 is an internal node that represents the interior tank. The ambient conditions are set by Node 2, which is a boundary node. For the unsteady formulation, a history file was provided for Node 2 to establish the pressure, temperature, and specie concentration at discrete time. In particular, Node 2 had its properties set to the constant ambient conditions of 14.1 psia and 80°F. For Node 1, the initial conditions were set to 1000 psia and 1040°F. Branch 12 was specified as a compressible orifice with a 1.0 flow coefficient.
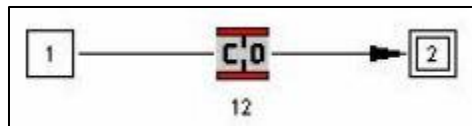

Figure 19. VTASC model of Example 8 [3]

After Example 8 was modified, the same three cases were run. The flow rate as well as the tank's pressure and temperature were examined with respect to time. Winplot graphs featuring the plots of all three cases can be seen in Figures 20 through 22 below.
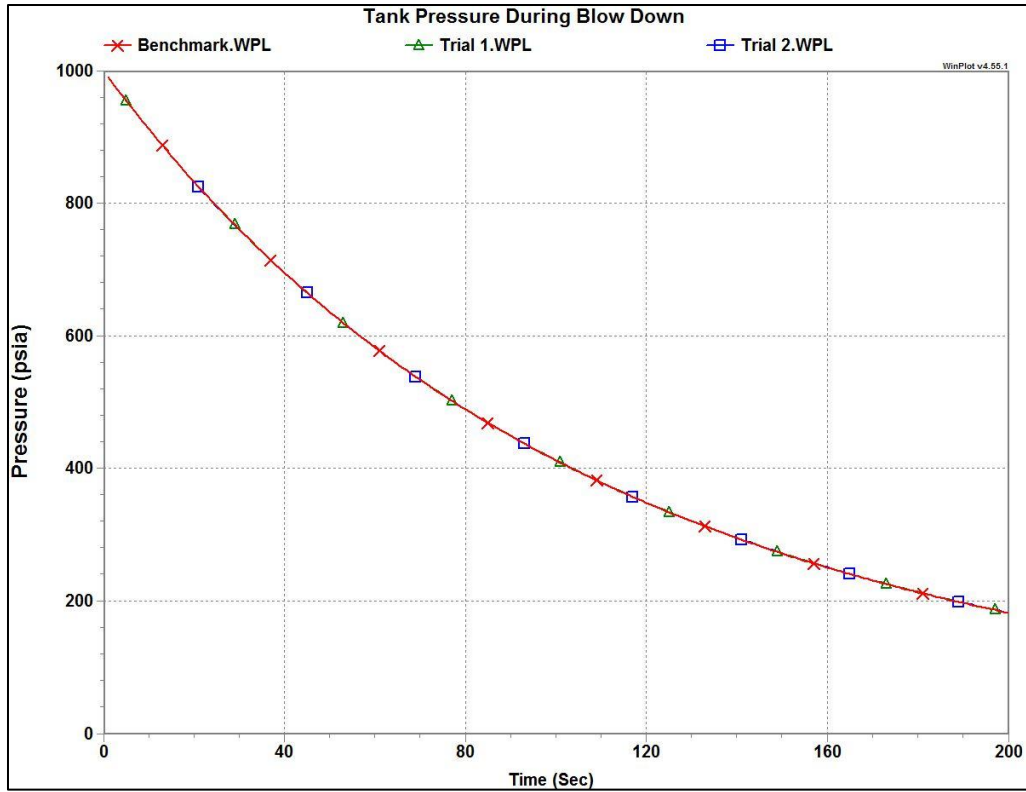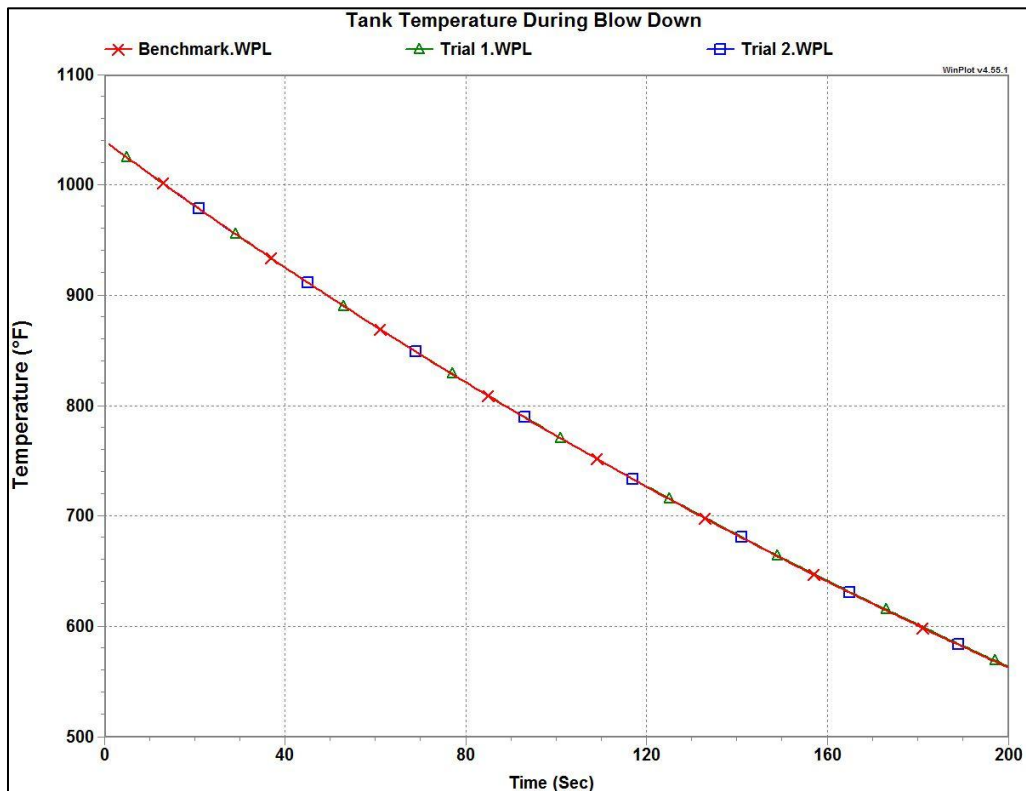
Figure 20. Comparative plot of tank's pressure change

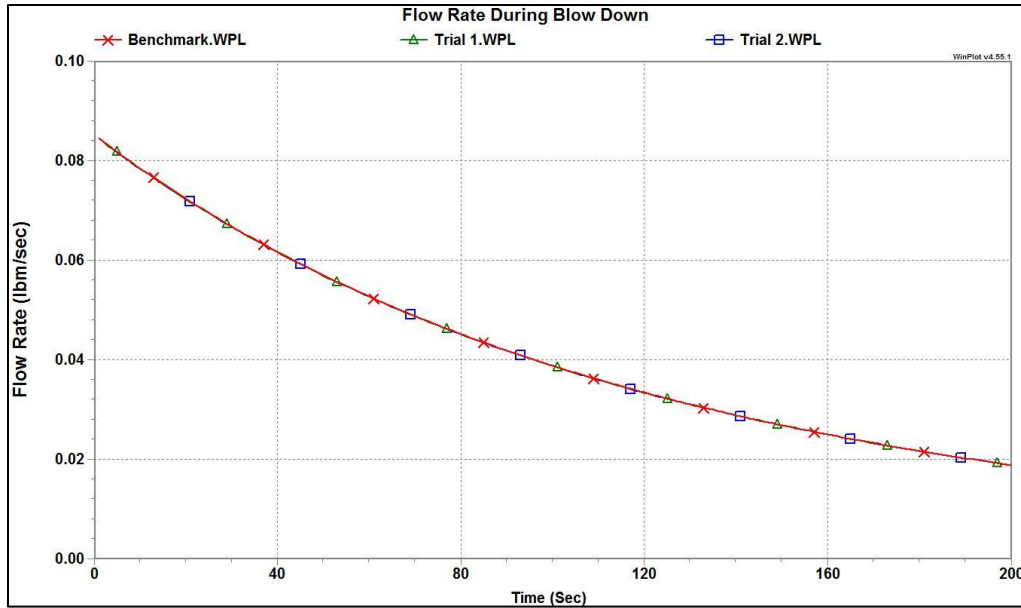**Figure 21. Comparative plot of tank's temperature change**

Figure 22. Comparative plot of flow rate

The results of these plots is very promising. Each graph has plots of all three cases, and each of the trial cases line up almost perfectly with the benchmark. Examination into the tabulated data showed little to no error. This verifies that the modified GFSSP601 is accurate for transient cases.

F. Simulation of Water Undergoing Phase Change in Simple Pipe

A simple steady-state model was created to test the new two-phase mixture capability of GFSSP. The model consisted of a single pipe that was designed to mimic the throttling process of the refrigeration cycle. A VTASC representation of the model can be seen in Figure 23. The model has three nodes and two branches. Nodes 1 and 3 are boundary nodes with pressure and temperature values of liquid water while Node 2 is an internal node. At Node 1, the pressure and temperature are 60 psia and 275°F respectively while at Node 3, the pressure and temperature are 15 psia and 60°F respectively. Branches 12 and 23 are pipe options with different dimensions. In order to encourage phase-change through the first pipe, Branch 12 is designed with a very large length-to-diameter ratio. The pipe's length is 1000 ft and its diameter is 0.1 in, which creates a *L/d* ratio of 120,000. As the water flows through the pipe, the large *L/d* produces a greater resistance, which produces a pressure drop to a point where the liquid water enters the saturation region. The effects of this pressure drop can be seen at Node 2 of the output file. The second pipe, Branch 23, is designed to allow the water to flow into ambient conditions, where it exists as a sub-cooled liquid. Specifically, the pressure and temperature Node 3 are 15 psia and 60°F respectively while the dimensions of the second pipe are 1000 in. in length and 1 in. in diameter.
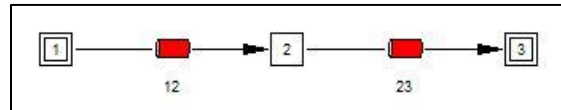

Figure 23. VTASC model of simple pipe with phase change

After the model was created, GFSSP was run for the same three cases. The results can be seen below in Table 7. Pressure, temperature, density, and quality were examined at Node 3 while the flow rate was taken from Branches 12 and 23.

Table 7. Results of simple pipe with induced phase change

Variables	Benchmark	Trial 1		Trial 2	
		Value	Percent Error	Value	Percent Error
P (psi)	15.01	15.01	0.000	15.01	0.000
T (°F)	213	210.6	-1.127	212.9	-0.047
ρ (lbm/ft3)	0.5814	0.2393	-58.841	0.5805	-0.155
ṁ (lbm/sec)	0.00393	0.00377	-4.071	0.00393	0.000
Quality	0.06485	0.158	143.639	0.06495	0.154

The results from this model are remarkable. As seen in Table 5, the quality for the benchmark was 0.06485, which indicates that the water experienced a phase-change from a liquid to a liquid-vapor mixture. Trial 1 produced a mixture of accuracies. For example, the pressure, temperature, and flow rate values were close to the benchmark. However, the density and quality were very far from the standard; the quality had a large error of about 144%. When the model was run with the more complex property tables, the results improved drastically. The pressure and flow rate matched perfectly with the benchmark while the temperature came to less than a tenth of a percent in error. The error in density and quality largely improved to about 0.15%.

This model is very important because it proves that the added phase-change capability for user-fluids is viable. The results also show that the property tables only prove accurate results provided that the property tables are detailed to a certain extent. Table 1 has a pressure increment of 10 psia while Table 2 has a pressure increment of 7.5 psia. This small change in increment made a huge difference as seen by the drop in quality error from 144% to 0.15%.

G. Run Time Comparison

The time it takes for a computer to analyze data, calculate equations, and produce results is an important variable. In order to run models in an efficient manner, the run time must be as low as possible. While each of the example problems and models were being run, the run time for each case was also observed. The goal was to determine whether using property tables were more or less efficient than using the built-in thermodynamic property databases. Table 8 lists the run times from each case for every model.

Table 8. Comparison of run times

Model	Benchmark (sec)	Trial 1 (sec)	Trial 2 (sec)
Example 1	0.0000	0.0156	0.0156
Example 3	0.1404	0.0624	0.1716
Example 5	0.0000	0.0000	0.0000
Example 6	0.0156	0.0156	0.0156
Example 8	0.2496	0.0624	0.3120
Simple Pipe	0.0000	0.0000	0.0000

For the most part, there is little correlation between the run time and the method of analyzing the models. Example 5 and the liquid-vapor simple pipe, for example, have an almost instantaneous run time, so a change is difficult to notice. Example 6 has a run time of 0.156 sec that is consistent across all cases. Example 1 had a slight increase in run time when using the property tables. Examples 3 and 8 had very interesting results. In these two models, Trial 1, which made use of the coarse property tables, had a shorter run time than the benchmark while Trial 2, which used the fine property tables, had a longer run time. The fact that Trial 2 results were generally more accurate than Trial 1 results shows that, for some examples, the accuracy of the model must be sacrificed to get a shorter run time.

Of course, all of these run times are very small. This is because the models were either steady-state or very simple transient cases. Run time becomes more of an issue for large models in the transient case. The results above prove that for some models, the use of a simple property table instead of the built-in databases can cut down on the run time. However, using a property table that is too detailed may also increase the run time. In other models, using

property tables provides no advantage or disadvantage over using GASP/WASP or GASPAK. Ultimately, the time of operation depends on what features are within the model itself.

VI. Conclusion

Effective thermo-fluid analysis programs are able to analyze a broad scope of fluids over a wide range of states in a minimal time. Through the use of user-fluids, one can essentially use any type of fluid in a GFSSP model. Also, with the addition of the saturation table in the form of an eighth property table, one can also analyze two-phase mixtures of user-fluids. The addition of saturation tables to GFSSP opens the door to using any fluid in any of the three states: sub-cooled liquid, superheated vapor, and saturated mixture. The revisions and results from the internship can be included in future versions of GFSSP. Currently, the modified GFSSP601 source code is being integrated with the GFSSP602. Also, changes are being made to VTASC602 so that users can provide the saturation table in the fluid options panel along with the basic seven property files. Further research can be done to determine the effect of property tables on the run time of model analysis.

Appendices

Appendix A: REFPROPGEN

Fortran code to convert REFPROP property tables into a GFSSP-readable format

```fortran
      PROGRAM REFPROPGEN

      INTEGER NP, NT
      DIMENSION P(26), T(76), rho(26,76), h(26,76), s(26,76), cp(26,76)
      DIMENSION gamma(26,76), ak(26,76), aak(26,76), emu(26,76)

C     Open File to Read
      OPEN(10,FILE='REFPROP.dat',STATUS='unknown')

C     OPEN Files to Write
      OPEN(11,FILE='rhowater.dat',STATUS='unknown')
      OPEN(12,FILE='hwater.dat',STATUS='unknown')
      OPEN(13,FILE='swater.dat',STATUS='unknown')
      OPEN(14,FILE='cpwater.dat',STATUS='unknown')
      OPEN(15,FILE='gammawater.dat',STATUS='unknown')
      OPEN(16,FILE='aakwater.dat',STATUS='unknown')
      OPEN(17,FILE='emuwater.dat',STATUS='unknown')

C     Read in number of pressures and temperatures
      READ(10,*) NP,NT

C     Read in property values
      DO I=1,NP
      DO J=1,NT
        READ(10,*) T(J),P(I),rho(I,J),h(I,J),s(I,J),cp(I,J),
     &            gamma(I,J),ak(I,J),emu(I,J)
      ENDDO
      ENDDO

C     Write property file for density
      WRITE(11,*) NP,NT
      WRITE(11,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(11,*) P(I),(rho(I,J),J=1,NT)
      ENDDO

C     Write property file for enthalpy
      WRITE(12,*) NP,NT
      WRITE(12,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(12,*) P(I),(h(I,J),J=1,NT)
      ENDDO

C     Write property file for entropy
      WRITE(13,*) NP,NT
      WRITE(13,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(13,*) P(I),(s(I,J),J=1,NT)
      ENDDO
```

```
C     Write property file for specific heat
      WRITE(14,*) NP,NT
      WRITE(14,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(14,*) P(I),(cp(I,J),J=1,NT)
      ENDDO

C     Write property file for specific heat ratio
      WRITE(15,*) NP,NT
      WRITE(15,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(15,*) P(I),(gamma(I,J),J=1,NT)
      ENDDO

C     Convert units of thermal conductivity and write property file
      aak=ak/3600
      WRITE(16,*) NP,NT
      WRITE(16,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(16,*) P(I),(aak(I,J),J=1,NT)
      ENDDO

C     Write property file for viscosity
      WRITE(17,*) NP,NT
      WRITE(17,*) (T(J),J=1,NT)
      DO I=1,NP
        WRITE(17,*) P(I),(emu(I,J),J=1,NT)
      ENDDO


      END
```

Appendix B: REFPROPSATGEN
Fortran code to convert REFPROP saturation table into a GFSSP-readable format

```
      PROGRAM REFPROPSATGEN

      INTEGER ND
      DIMENSION Tsat(321),Psat(321),rhol(321),rhov(321),hl(321),hv(321),
     &        sl(321),sv(321),cpl(321),cpv(321),gaml(321),gamv(321),
     &        akl(321),akv(321),emul(321),emuv(321)

C     Open REFPROP file to read
      OPEN(1,FILE='satwaterref.dat',STATUS='unknown')

C     Open write file
      OPEN(2,FILE='satwater.dat',STATUS='unknown')

C     Read in number of data points
      READ(1,*) ND

C     Read in propery values
      DO I=1,ND
```

```
      READ(1,*) Tsat(I),Psat(I),rhol(I),rhov(I),hl(I),hv(I),sl(I),
   &            sv(I),cpl(I),cpv(I),gaml(I),gamv(I),akl(I),
   &            akv(I),emul(I),emuv(I)
   ENDDO

C     Convert thermal conductivity from btu/h-ft-R to btu/h-s-R
   akl=akl/3600
   akv=akv/3600

C     Write saturation file
   WRITE(2,*) ND
   DO I=1,ND
      WRITE(2,*) Psat(I),Tsat(I),hl(I),rhol(I),cpl(I),emul(I),
   &            gaml(I),akl(I),sl(I),hv(I),rhov(I),cpv(I),emuv(I),
   &            gamv(I),akv(I),sv(I)
   ENDDO

   END
```

Acknowledgements

References

[1]Çengel, Y., Introduction to Thermodynamics and Heat Transfer, 2nd ed., McGraw Hill, Boston, 2008, p. 118.
[2]Moran, M. and Shapiro, H., *Fundamentals of Engineering Thermodynamics*, 3rd ed., John Wiley & Sons, Inc., New York, 1996, p. 835.
[3]Majumdar, A., Steadman, T., and Moore, R., "Generalized Fluid System Simulation Program (GFSSP) Version 5.0 (Draft Report)," URL: http://gfssp.msfc.nasa.gov/pdf/GFSSPUsersManual_V5.pdf.