Grand Valley State University
ScholarWorks@GVSU

Masters Theses   Graduate Research and Creative Practice

8-2018

Modeling of Cold Compressor Pump Down Process

Kyle A. Dinger
Grand Valley State University

Follow this and additional works at: https://scholarworks.gvsu.edu/theses

Part of the Engineering Commons

Recommended Citation

Modeling of Cold Compressor Pump Down Process

Kyle A. Dinger

A Thesis Submitted to the Graduate Faculty of

GRAND VALLEY STATE UNIVERSITY

In

Partial Fulfillment of the Requirements

For the Degree of

Master of Science in Engineering

School of Engineering

August 2018

Acknowledgments

I would like to thank Dr. Wael Mokhtar for his unwavering support and encouragement during my undergraduate and graduate studies and especially over the course of this project. Dr. Mokhtar's enthusiasm for Grand Valley State University engineers achieving their potential is a catalyst driving the success of Grand Valley Engineering.

I would like to thank Dr. Peter Knudsen for his time and efforts instructing me and shaping the direction of this research. Without his guidance on the technical aspects of this project as well as providing context for the processes therein, there would have been no project.

I would like to thank Bruce Dinger for his continuous support and ability to reduce a problem down to its lowest level. It is easy to become overwhelmed in the face of a complex problem. It takes a concerted effort to work through the convolution and organize the issues in the simplest form and clarify how to deal with them. It has been a great pleasure to have a wonderful father and fantastic engineer to provide a second set of eyes and to challenge my assumptions in order to better understand the problem at hand.

I would like to thank Dr. Venkatarao Ganni for pushing the study of cryogenics engineering at the collegiate level and for allowing me the opportunity to pursue a more complete understanding of the application of engineering fundamentals at the Facility for Rare Isotope Beams. The challenges presented and the opportunities given are world class.

I would like to thank Dr. Mehmet Sozen for his time and attention while serving on my thesis committee. I have learned an enormous amount from Dr. Sozen while at Grand Valley.

I would like to thank my family and friends both in Michigan and Tennessee. Spending time with them allowed for a break from the stress of the work. The love and encouragement from them have been invaluable throughout the entirety of this endeavor. I will always be having fun while I am with you.

Abstract

The Facility for Rare Isotope Beams (FRIB) will use a sub-atmospheric helium refrigeration process operating at 2 K (31 mbar) to support the superconducting radio frequency (SRF) Niobium structures (known as cavities), which are housed within 'cryo-modules'. The cryo-modules are large containers whose exterior forms a vacuum chamber that serves as a thermal shield. The cryo-modules, and the superconducting devices contained within, are used to accelerate charged particles. The accelerator at FRIB is comprised of three separate linear segments, separately or collectively, called a linear accelerator or 'LINAC'. The helium used as the working fluid to cool the SRF Niobium cavities is supplied from a 4.5 K refrigerator, but the sub-atmospheric condition will be produced by 'pumping-down' the LINAC using cryogenic (cold) centrifugal compressors to remove mass, thus reducing the pressure within the SRF Niobium cavities. The initial condition of liquid helium before starting a 'pump-down' can range from a 2 K sub-cooled liquid to a saturated liquid at around 1 bar. These initial condition extremes will result in pump-down processes that are different. This variability of initial conditions increase the complexity of the overall process. As such, a process model can provide considerable insight into the best approach to use for a particular pump-down.

This research has developed a simplified model of sub-atmospheric components downstream of the 4.5 K cold box. The initial condition of the helium within the SRF Niobium cavity is assumed to be a saturated mixture at near atmospheric pressure and remain a saturated mixture as the pump-down proceeds. The prime mover in this study is a single radial centrifugal cold compressor removing mass from the Niobium SRF cavities. A model for the return transfer line is incorporated to simulate pressure drop, heat in-leak, and mass accumulation of the sub-atmospheric helium returning from the LINAC back to the cold compressor. A counter flow heat exchanger is also a part of the model. This heat exchanger uses the sub-atmospheric helium stream leaving the SRF cavity to the cool the supply

stream from the 4.5 K cold box. The model accounts for the non-constant thermal capacity rates

present in this heat exchanger. The sum of the SRF cavities are modeled as a single dewar process,

with a non-flowing two-phase mixture. The dewar process involves heat transfer to the liquid, and

mass and energy depletion. The model is used to study the time to achieve a desired final within the

dewar for a given set of system parameters. The component models are individually validated. The

overall process can be extended and validated and compared to the FRIB process after such

commissioning is complete. This model serves as the foundation for further process studies.

Table of Contents

List of Figures

Key to Symbols

Symbol	Description
A	Cross sectional area
a	Speed of sound in fluid
C	Absolute fluid velocity
C_{max}	Larger stream heat capacity for a given heat exchanger division
C_{min}	Lower stream heat capacity for a given heat exchanger division
C_f, f	Fanning friction factor
$C_{h,i}, C_{l,i}$	Heat capacity of high or low pressure stream at given division 'i'
C_p	Specific heat of fluid at constant pressure
$C_{R,i}^*$	Specific heat ratio across heat exchanger division
C_v	Specific heat of fluid at constant volume
d	Diameter
D	Diffusion parameter
G_{in}	Mass flux at the return line volume inlet
GC	Geometry constant for return line volume
h	Specific enthalpy
h_b	Height of given passage
K_f	Torque coefficient
L	Length of inlet duct
M	Mach number
$\dot{m}$	Mass flow rate

$\dot{m}_g$	Mass flow rate of saturated vapor out of cryo-module
Ntu	Number of transfer units
P	Pressure
P_0	Stagnation pressure
Pr	Pressure ratio across cold compressor
r	Radius measured from impeller wheel centerline
R	Gas constant for fluid
Rn	Richardson number
T	Temperature
T_0	Stagnation temperature
Tr	Temperature ratio across cold compressor
U	Impeller blade speed, internal energy
UA	Overall conductance
$\tilde{U}, \tilde{V}, \tilde{V}_l, \tilde{V}_v, \tilde{v}, \tilde{m}$	Intermediary variables in dewar derivation
u	Specific internal energy
v	Specific volume
$v_{h \to s}$	Inducer hub to shroud radius ratio
V	Volume
$\dot{\mathcal{V}}$	Volumetric Flow Rate
W	Relative fluid velocity
x	Vapor quality
Z_b	Number of blades on impeller

α	Absolute fluid angle, relaxation constant
β	Relative fluid angle, saturated volume expansivity
β_b	Impeller blade angle
γ	Ratio of specific heats
ε	Stationary clearance of impeller blade
θ_i^*	Non-dimensional temperature differential across heat exchanger division
κ_T	Saturated isothermal compressibility
λ	Latent heat of vaporization
μ	Dynamic viscosity
μ_{slip}	Slip factor
ρ	Fluid density
σ	Entropy gain
ω	Angular velocity

Subscripts

0	Stagnation condition
1	Node one: outlet of 4.5 K cold box inlet to (h) stream of heat exchanger
2	Node two: outlet of (h) stream heat exchanger before JT valve
3	Node three: cryo-module conditions and inlet to (ℓ) stream
4	Node four: outlet of (ℓ) stream inlet to return line volume
5	Node five: outlet to return line volume inlet to cold compressor
6	Node six: outlet of cold compressor

b	blade
est	Value used to estimate heat exchanger stream temperature profile
h	High pressure stream (heat exchanger), impeller hub
i	Nodal position in heat exchanger matrix
in	Inlet
l	Liquid, low pressure stream (heat exchanger)
m	Meridional direction
out	Outlet
s	Isentropic, compressor shroud
T	Tip
v	Vapor
x	Inlet to generic duct for derivation
y	Outlet to generic duct for derivation
θ	Tangential direction

1. Introduction

Cooling required for super-conducting radio-frequency (SRF) structures used in modern particle accelerators is needed at temperatures at or below 4.5 K. The only refrigerant that will not solidify at this temperature is helium. Typically these structures, which are typically referred to as 'cavities' and are constructed of Niobium, are cooled below 4.5 K to achieve optimum performance and cost. Since the normal boil point for helium is around 4.5 K, this requires the helium to be sub-atmospheric at some point in the refrigeration process. A good example of this kind of refrigeration process can be found at Michigan State University's Facility for Rare Isotope Beams (FRIB), which uses the Ganni Cycle Floating Pressure Process [1] for optimum operational efficiency and availability.

1.1. Overview

The warm compressor system provides the availability (exergy) for the entire process. The process being inclusive of refrigeration system, distribution system, and the end useful use; i.e. the 'load'. Most modern helium systems using recuperative heat exchange use twin rotary screw compressors. Ideally this compression process is isothermal, with input power rejected as heat to the environment. The 4.5 K uses the availability of the high pressure (~ 20 bar) and near atmospheric temperature helium and cools the helium using components such as heat exchangers and adiabatic expanders. The nomenclature of 'cold box' means that components are housed within a vacuum vessel in conjunction with insulation to minimize heat in-leak to the helium.
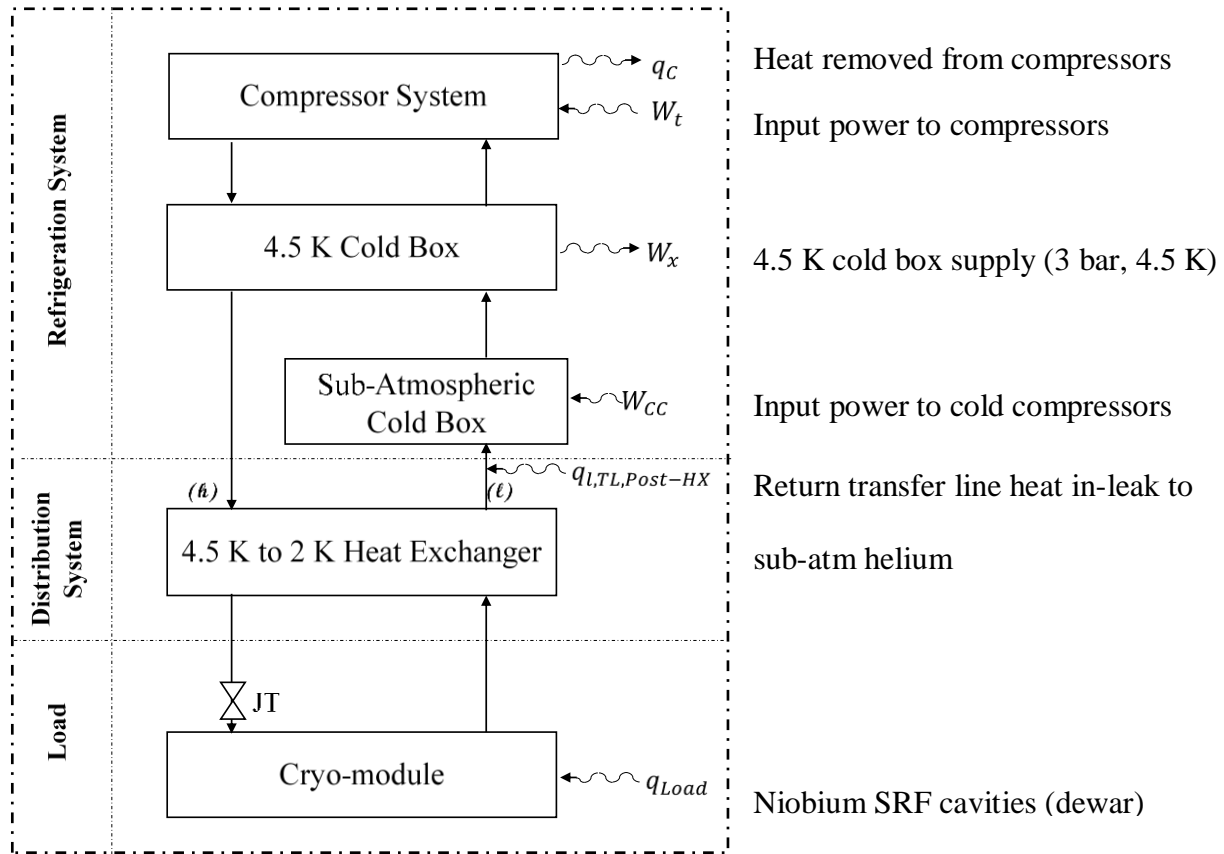
Figure 1. FRIB simplified sub-atmospheric helium process

Figure 1 shows a simplified diagram of the overall FRIB helium process. Note, not all

components in this diagram are modeled in this study namely, the warm compressor system and

the 4.5 K cold box.

1.1. Cryo-module

Cryo-modules are vacuum vessels that serve to provide thermal insulation, via the vacuum and

insulation. These containers are used to house the superconducting devices in the LINAC,

including the SRF Niobium cavities. Super-conducting magnets are also within cryo-modules.

However, these will not be considered for this study. Although these components are insulated,

there is still an unneglectable heat in-leak that must be considered. Further, during the beam

operations, when the SRF cavities are in (roughly) steady sub-atmospheric operation, the RF is

pulsed on and off with a cycle time for RF operation in milliseconds. There is a dynamic

component of heat that goes into the sub-atmospheric helium. Heaters are needed to compensate for these transients.

Within the cryo-module, the SRF cavities are supplied by a large pipe header. These collectively are modeled as a dewar (process). This 'dewar' process is a saturated unsteady process, as the system depressurizes with time.

1.2.4.5 K to 2 K heat exchanger

The sub atmospheric heat exchanger, in the cryo-module, is used to recover refrigeration from gas which has recently been removed from the cryo-module. The supply fluid is assumed to be supercritical (4.5 K and 3 bar). It is assumed that the helium gas that is being removed from the cryo-module is a saturated vapor. This heat exchanger is an essential component to affect good overall process efficiency. It is imperative that the heat exchanger be located as physically close to the vapor leaving the SRF cavity as practical to avoid heat in-leak at 2 K. However, this also elevates the suction temperature of the first cold compressor, the effect of which increases the required volume flow for a given mass flow rate.

1.3. Transfer line system

The helium transfer lines handle the supply and return helium flow to and from the LINAC tunnel. The transfer line is a collection of pipes within an overall vacuum jacket and includes a thermal radiation shield. A cross section of the FRIB LINAC transfer line can be seen in Figure 2.
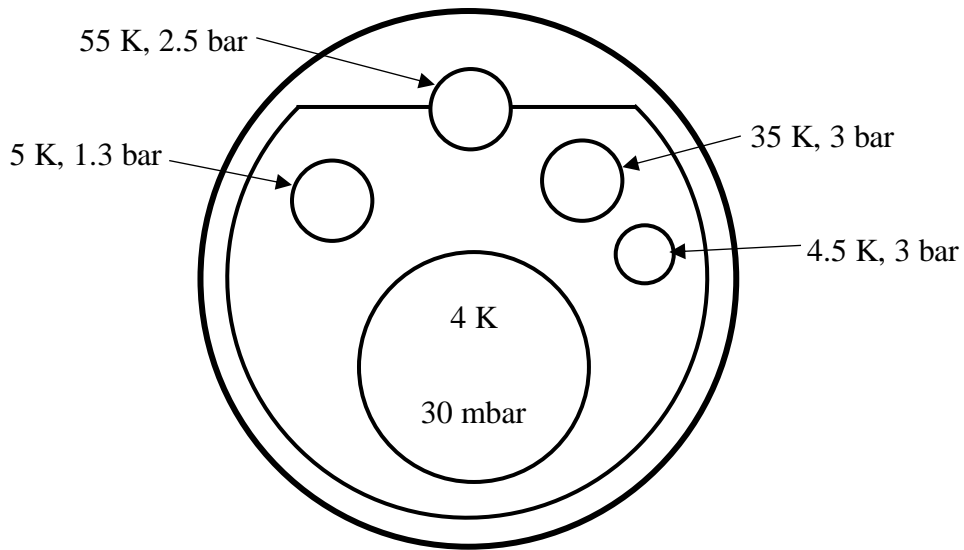
Figure 2. Cross section of FRIB transfer lines [2]

Figure 2 identifies a *4 K 30 mbar* line and a *4.5 K 3 bar* line, these are the return from the SRF

cavities and supply from the 4.5 K cold box, respectively. These process conditions shown in

Figure 2 are nominal, and can vary. The pressure drop through the sub-atmospheric line is

primarily a function of the mass flow across the cold compressors and the effect of friction in the

transfer line system. The effect of heat in-leak is assumed to only effect the gas temperature.

1.4. Sub-Atmospheric Cold Box

The cold compressors are the prime movers in this system model. They are used to achieve the

sub-atmospheric pressure conditions in the cryo-modules to support the superconducting devices.

These cold compressors are housed in the sub-atmospheric cold box above ground and are

mounted on the return side of the helium distribution system, seeking to remove mass from the

cryo-module and reinject helium back into the 4.5 K cold box.

The cold compressors are radial centrifugal devices which are assembled in a "train" to achieve

an overall compression pressure ratio, above the desired sub-atmospheric conditions in the

LINAC. In order to achieve the sub-atmospheric conditions in the LINAC, the cold compressors

need to remove mass from the cryo-module containers and depressurize the system. This mass

removal along with the desired pressure ratios dictate the speed of the cold compressors as the system depressurizes and approaches the desired operating condition. This mass flow rate out of the cryo-modules during the depressurization is dictated by the cryogenics staff and is chosen to minimize time as well as attempt to maximize efficiency of the system pump-down.

Historically these pump-down "paths" have been determined empirically. This study is intended build the initial models which will be incorporated in future studies to give insight into possible process paths for the system.

2. Literature Review

It is understood that there are a variety of process paths and a significant need to better model

and understand the transient behavior of these systems.

This study will model the components downstream of the 4.5 K cold box to observe the effects

of the depressurization process. Figure 3 shows how the system level model will be broken up

into components and nodes.
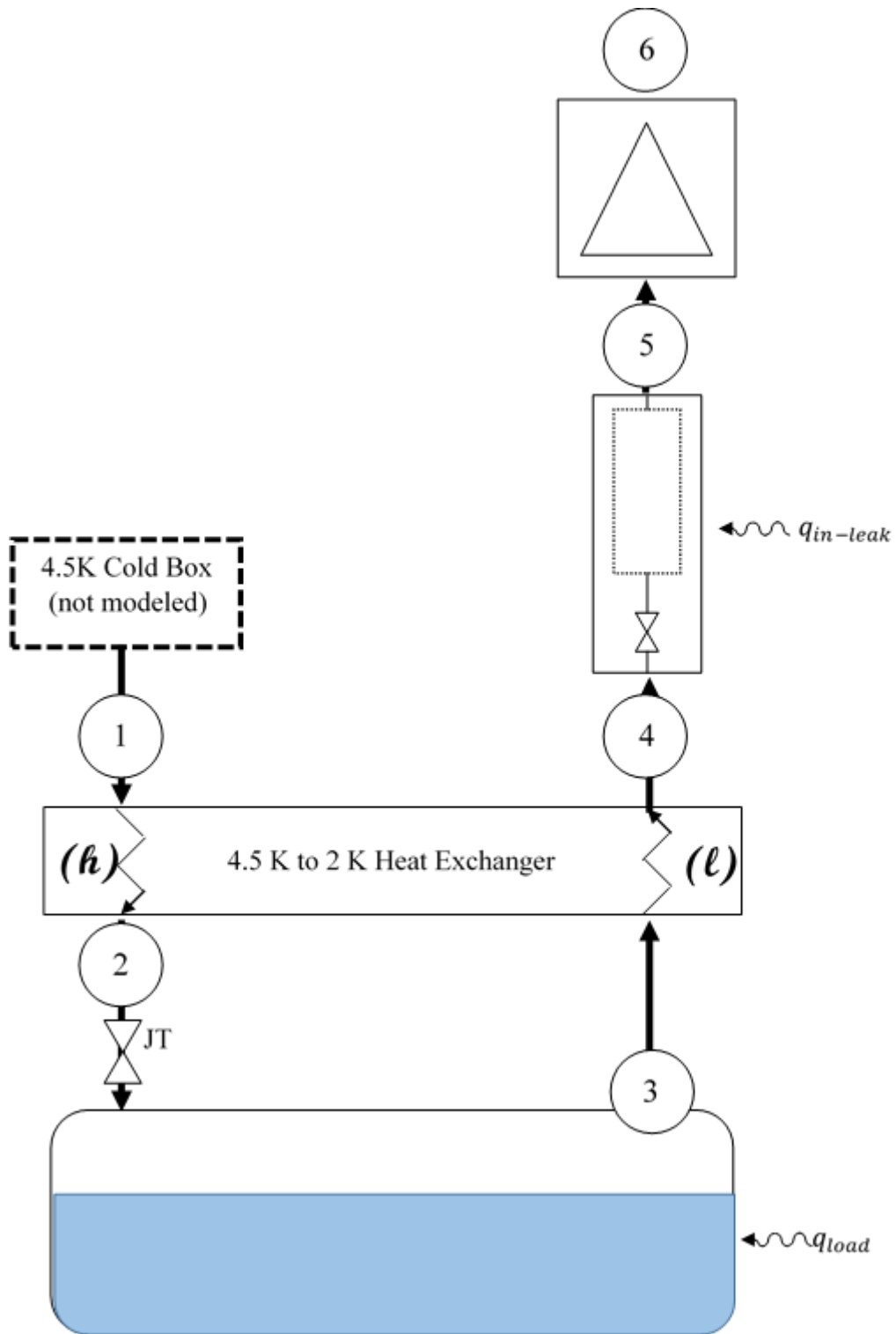
Figure 3. Present study model map showing components and nodes in the system

The nodes and components can be seen in Table 1.

Table 1. Model nodes and descriptions corresponding to the model map for the system

Node	Description
1	Outlet of 4.5 K cold box, inlet to high pressure stream (warm end) of heat exchanger
2	Outlet of high-pressure stream (cold end) of sub-atmospheric heat exchanger and inlet to JT valve prior to cryo-module
3	Outlet of cryo-module, inlet to low-pressure (cold end) of heat exchanger
4	Outlet of low-pressure (warm end) of heat exchanger and inlet to return line volume
5	Outlet of return line volume and inlet to cold compressor
6	Outlet of cold compressor

The cold compressor will be modeled using a single cold compressor model with a prescribed mass flow rate and constant pressure ratio. The model is adapted from Design of Radial Turbomachinery by Whitfield and Baines [3]. It is a mean streamline one dimensional model that simplifies the components of the device into ducts. In this case, the inlet, impeller, and vaneless diffuser are being modeled as stationary duct, rotating duct, and duct housing unguided swirling flow respectively [4]. The non-idealities of the system are modeled using loss coefficients to solve for the dimensionless entropy gain for each of the ducts. The performance of these models are validated against pedigreed test data by Eckardt [5] and loss model studies from Oh *et al* [6].

The cryo-module dewar process was modeled as a variable pressure, constant volume (rigid) vessel containing saturated liquid and saturated vapor, subject to a given heat in-leak, supply two-phase mass flow, and exiting saturated vapor flow. Further, for this study, the liquid volume fraction within the container was assumed to be held constant by the JT valve. In kind, there is a constant liquid level in the cryo-module dewar. There is a constant heat inleak on the dewar to model the load heat as well as thermal contamination. The incoming supply flow is expanded

isenthalpically into the two component phases in the dewar. The supply flow rate is assumed to support the saturated liquid being boiled off. For a given time step a mass flow rate is prescribed and the rate of change of pressure in the system is calculated and then integrated to find the next time step pressure. Following the calculation of the next time step pressure, the next time step mass flow rates (outlet and supply) that correspond to the prescribed process path (mass flow rate as a function of load pressure) are calculated.

The heat exchanger is modeled as an insulated metallic mass in thermal equilibrium with the component high and low-pressure streams, with no pressure drop across both streams, and no mass or energy accumulation in either the fluid or the construction material. In order to account for the varying specific heat of the helium, the heat exchanger was divided into 10 divisions [7]. Within each division the heat capacity of each of the streams was evaluated at the inlet temperatures to produces a model that allows for varying heat capacity along the length of the heat exchanger. Finally the heat exchanger was assumed to have an overall number of transfer units equal to 3.

The return transfer line is modeled in two segments. The first is a pressure drop across the line. This pressure drop produces the pressure "potential" to flow back to the sub-atmospheric cold box and cold compressor inlet. The second is a control volume of constant volume which is depressurizing as mass is pulled by the cold compressor and throttled at the inlet of the cryo-module dewar. There is a constant heat in-leak assumed over the length of the transfer line and as such the pressure and temperature of the return volume is solved for by integrating the rate of change of density and internal energy. The differential equations for each the rate of change of density and the rate of change of internal energy are solved for by evaluating the equations for conservation of mass and energy.

These models will be combined and solved beginning at the cold compressor and evaluating backwards to the 4.5 K cold box. From there the differential equations of the return transfer line and the cryo-module dewar will be integrated to evaluate for the next cold compressor inlet conditions and the next cryo-module dewar load pressure, the time incremented, and the system evaluated again until the desired load pressure is attained.

The outputs of this study will be an evaluation of time to pump-down and compressor efficiency during the process. These will be studied for different prescribed process paths for the system. Furthermore, an evaluation of how these models can be modified and elaborated upon to create a more accurate modeling of the FRIB refrigeration system and evaluate the system over a wider range of initial conditions.

3. Methodology

The methodology for modeling the components and integrating those models will be to create subroutines and evaluate each model in series and exchange information between the models as necessary.

3.1. Cold Compressor Model

The purpose of this study is to model helium property changes through each stage of the cryogenic (cold) compressor train, and evaluate the train exit conditions for a given time step during the pump down process. This model will take in superheated vapor that has left the sub-atmospheric heat exchanger and travelled through the sub-atmospheric return lines to the 2K cold box. The ducts are represented by the same governing relationship from Whitfield and Baines *Design of Radial Turbomachines*.

3.1.1. Efficiency Metrics

The observed performance metrics of the cold compressor model in this study are the isentropic efficiency (η_{isen}) and the polytropic efficiency (η_{poly}), otherwise known as the small-stage efficiency. This is different from the isentropic efficiency in that the compressor pressure ratio is not an application defining parameter in η_{poly} as it is in η_{isen}.

Where the isentropic efficiency is defined as the ratio of actual enthalpy difference to isentropic enthalpy difference of the compression process (see (1)).

$$\eta_{isen} = \frac{h_{02s} - h_{01}}{h_{02} - h_{01}} \tag{1}$$

(2) Shows how the isentropic efficiency can also be defined in terms of the pressure ratio and temperature ratio [3].

$$\eta_{isen} = \frac{Pr^{\frac{\gamma-1}{\gamma}} - 1}{Tr - 1} \tag{2}$$

Due to the pressure ratio dependency of isentropic efficiency for a given compression process makes it difficult to compare compressor efficiency of machines over different operating conditions (pressure ratio settings).

To overcome the difficulty in the comparison of turbomachines the polytropic efficiency is introduced. The polytropic efficiency in equation (3).

$$\eta_{poly} = \frac{\partial h_s}{\partial h} \tag{3}$$

Polytropic efficiency is the isentropic efficiency over an infinitesimally small enthalpic difference.

$$\eta_{isen} = \frac{Pr^{\frac{\gamma-1}{\gamma}} - 1}{Pr^{\frac{\gamma-1}{\gamma \eta_{poly}}} - 1} \tag{4}$$

Whitfield and Baines derived a relationship between the isentropic and polytropic efficiencies in (4) which leads to (5) to solve for the polytropic efficiency of a compression process.

$$\eta_{poly} = \frac{f * \ln Pr}{\ln\left(\frac{Pr^f - 1}{\eta_{isen}}\right)} \tag{5}$$

where $f = \frac{\gamma-1}{\gamma}$.

3.1.2. Derivation of Whitfield and Baines duct equation

Whitfield and Baines are modeling the components of a compressor as ducts. The nature of the duct and the fluid flowing through it determine what component of the compressor is being modeled. That is, the inlet, impeller, and vaneless diffuser can be modeled as stationary duct

handling straight axial flow, rotating curved duct, and stationary duct handling unguided swirling flow, respectively [4]. In order to accomplish this a generalized duct equation is derived. With the addition of 'dimensionless entropy gain' correlations, real (non-ideal) fluid conditions can be approximated.

To derive the relationship begin with continuity for the system.

$$\dot{m} = \rho C A \tag{6}$$

From there introduce the Mach number to remove the absolute fluid velocity C.

$$M = \frac{C}{a} = \frac{C}{\sqrt{\gamma R T}} \tag{7}$$

Replace P and T with their total pressure P_0 and total temperature T_0 analogues.

$$\frac{P_0}{P} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \tag{8}$$

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2 \tag{9}$$

The above equations are used in the derivation of the generalized duct equation that is used in this analysis [3]. The primary differences being the addition of entropy gain and the introduction of 'relative' terms. Where entropy gain can be defined as seen in (10).

$$\sigma = \exp\left(-\frac{\Delta s}{R}\right) \tag{10}$$

However, it is necessary to use correlations to solve for the dimensionless entropy gain in each duct.

The generalized duct equation can be found in equation (11) as found in [3].

$$\frac{\dot{m}\sqrt{\frac{RT'_{02}}{\gamma}}}{A_2 P'_{02}} = \frac{A_3}{A_2} \cos\beta_3\, M'_3 \left(1 + \frac{\gamma-1}{2} M'^2_3\right)^{-\left(\frac{\gamma+1}{\gamma-1}\right)}$$

$$\times\, \sigma \left(1 + \frac{\gamma-1}{2\gamma RT'_{02}} (U_3^2 - U_2^2)\right)^{\frac{\gamma+1}{2(\gamma-1)}}$$

(11)

The cold compressor model is a combination of three duct models (inlet, impeller, vaneless diffuser). The fluid properties of each duct model outlet serve as fluid properties to the inlet of the next.

3.1.3. Cold compressor model map

As the compressor is split into three ducts, with nodes in between, it is important to enumerate those positions to be clear what point within the compressor is being referred to. Figure 4 shows the compressor model map. The nodal positions (nodes $1 - 4$) shown in the compressor model map are used to specify fluid conditions between and isolate the three duct types being modeled in the compressor subroutine.
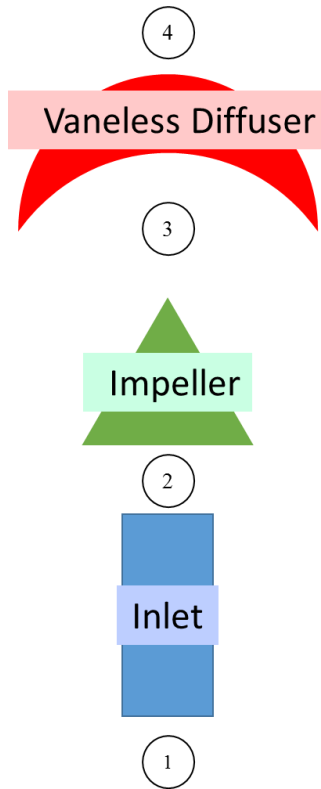
Figure 4. Compressor model map to show compressor nodes and components

3.1.4. Cold compressor inlet duct

The first duct model represents the compressor inlet. It is modeling simple stationary duct, accounting for friction losses. The fluid properties: static temperature, static pressure, and mass flow rate are provided to the model. The blade velocities are equal to zero as the inlet model is modeled as a stationary duct.

Table 2. Geometry and operating inputs to inlet duct model

Inlet duct	Value	Unit	Description
L_inlet	0.075	[m]	length of inlet duct
d_inlet	0.152	[m]	diameter of inlet duct
k_inlet	0.0015	[m]	surface roughness of inlet duct

Table 2 shows the user provided geometry data for the inlet model. The inlet is modeling pipe flow there is no angular velocity or tip speed data necessary and therefore the right hand side of (11) simplifies to:

$$\frac{\dot{m}\sqrt{\frac{RT_{01}}{\gamma}}}{A_x P_{01}} = \frac{A_2}{A_1} M_2 \left(1 + \frac{\gamma - 1}{2} M_2^2\right)^{-\left(\frac{\gamma+1}{2(\gamma-1)}\right)} \sigma \tag{12}$$

Notice (12) does not contain any relative terms. The areas A_1 and A_2 are equal in this case and

the stagnation temperature and pressure for the inlet are produced from the boundary conditions

of the inlet and stagnation pressure and temperature relationships.

The model then solves for both the right hand side (RHS) and the left hand side (LHS) of (12) by

iterating the outlet Mach number M_2 until the RHS and LHS are equal within a threshold value

of $\pm 10^{-6}$. Notice that $\sigma = 1$ in this first solution. That is, the initial solution is the isentropic

solution. This provides the model with a starting point from which to calculate outlet fluid

properties.

With the fluid properties known an accurate friction coefficient can be solved, entropy gain

updated, and the process re-iterated to find non-ideal system characteristics. The entropy gain

due to friction in the inlet is solved using (13).

$$\sigma = \exp\left(-\frac{\Delta s}{R}\right) = 1 - \frac{4 C_f L \bar{W}^2 \rho}{2 D P_{01}} \tag{13}$$

In equation (13), the term $\bar{W}$ refers to the average relative velocity at the inlet and outlet of the

duct. Once the Mach number and the entropy gain have converged for the inlet duct the solver is

terminated and the information passed to the impeller duct. The flow chart for operation can be

seen in Figure 5.
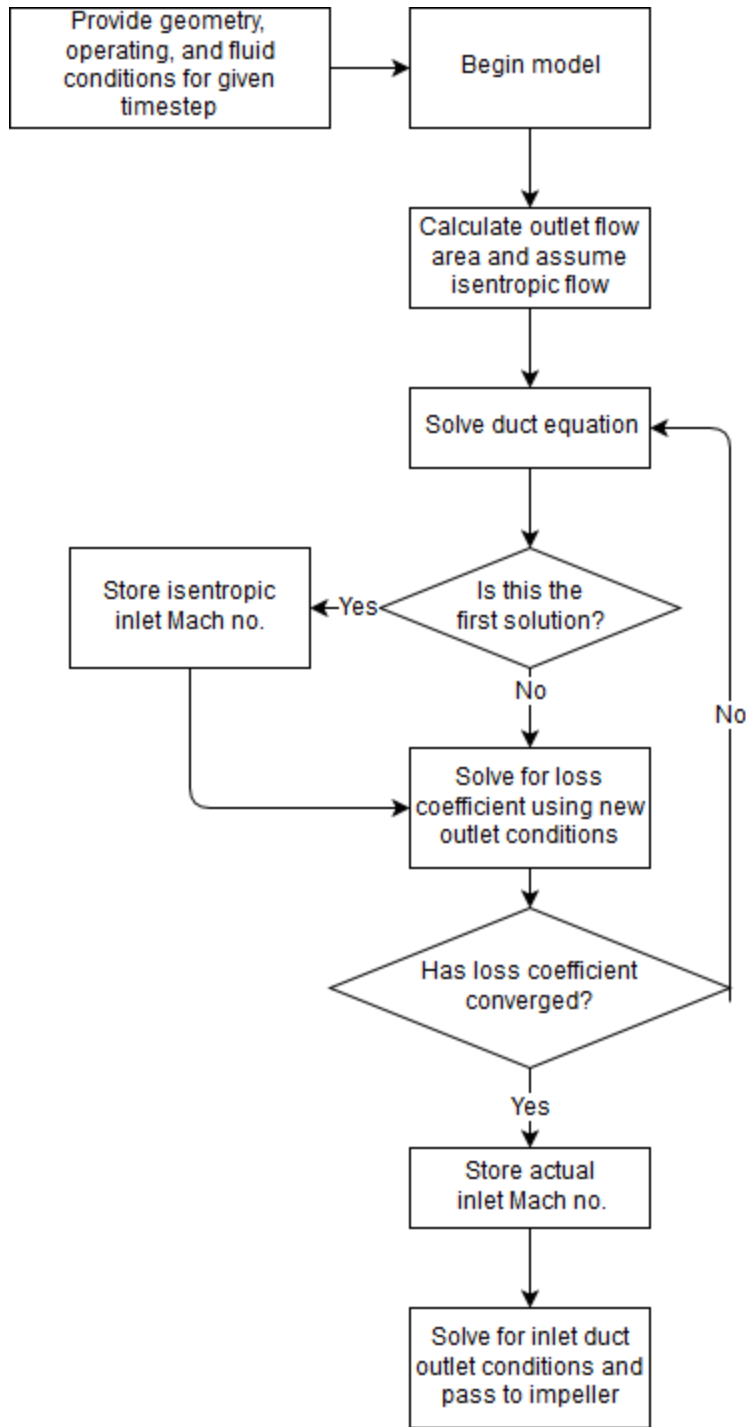
Figure 5. Inlet duct subroutine flow chart [3]

3.1.5. Cold compressor impeller duct

The impeller duct utilizes the same base continuity equation; however, the impeller is modeled

as a rotating duct. That is, ω is nonzero in (11). Once the system has been defined geometrically,

the system can be solved similarly to the inlet duct. However, in this instance, there are more

losses than just skin friction.

Table 3. Geometric constants to be prescribed

Impeller	Value	Units	Description
d_outlet3	0.005662	[m]	outlet diameter of impeller stage
Z_b	20	[-]	number of blades on impeller
by	0.1	[m]	impeller blade height
beta_b3	0	[rad]	impeller blade angle
k_impeller	0.001	[m]	surface roughness of impeller duct
eta	0.001	[m]	stationary clearance of impeller blade
rwheel_o	0.098	[m]	radius of impeller tip at outlet
omega	varies	[rad/s]	angular velocity of impeller wheel
rxs	0.044	[m]	radius measured from centerline of impeller wheel to shroud at inlet
rxh	0.018	[m]	radius measure from centerline of impeller wheel to hub exterior at inlet

The inlet area to the impeller is simply the area of the inlet blade edge minus the hub diameter,

and the outlet area is the circumference of the outer blade edge times the height of the blade at

the discharge.

The first solution is solved isentropically, assuming $\sigma = 1$. An assumption for β_3 has to be

prescribed in order to begin the procedure. The chosen assumption is $\beta_3 = 1°$. Note the

difference between β_{blade3} and β_3. β_{blade3} refers to the blade angle at the outlet of the impeller

duct and β_3 refers to the relative flow angle at the outlet of the impeller duct. From this point a

possible M_3' is solved and then a new β_3 is solved for a compared to the previous.

$$\sin \beta_3 = \frac{-\dfrac{U_3\left(1 - \mu_{slip}\right)}{W_3} + \tan \beta_{blade3}\left[1 + \tan^2 \beta_{blade3} - \dfrac{U_3^2\left(1 - \mu_{slip}\right)^2}{W_3^2}\right]}{1 + \tan^2 \beta_{blade3}} \quad (14)$$

where μ_{slip} is the slip factor. A correlation from Wiesner (1967) solved for slip factor using

$$\mu_{slip} = 1 - \frac{\sqrt{\cos \beta_{blade3}}}{Z_B^{0.7}} \tag{15}$$

where Z_B is the number of blades on the impeller. Once the angle has converged between Mach no solutions the Mach no is accepted and the solution moves on to loss correlations.

After the first (isentropic) solution is produced, loss coefficients are introduced to solve for stagnation enthalpy loss.

3.1.6. Impeller Loss Coefficients

Oh *et al* presented a series of loss coefficients for use with one dimensional models.

The first loss coefficient models losses associated with blade incidence, and was adopted from Conrad *et al* (1980).

$$\Delta h_{inc} = f_{inc} \frac{W_{ui}^2}{2} \tag{16}$$

The incidence friction factor is shown in (17).

$$f_{inc} = 0.5 - 0.7 \tag{17}$$

The loss coefficient is due to skin friction loss. This was modeled by both Jansen (1967) and Coppage *et al.* (1956).

$$\Delta h_{sf} = 2C_f \frac{L_b}{D_{hyd}} \overline{W}^2$$

$$\overline{W} = \frac{C_{3t} + C_3 + W_{2t} + 2W_{2h} + 3W_3}{8} \tag{18}$$

where C_f is the fanning friction factor, D_{hyd} is the hydraulic diameter of the impeller duct, and L_b is the axial height of the impeller blade.

The next loss considered was diffusion and blade loading loss.

$$\Delta h_{bld} = 0.05 \, D_f^2 U_2^2 \tag{19}$$

Where the diffusion factor (D_f) is calculated in (20).

$$Df = 1 - \frac{W_y}{W_x} + \frac{\dfrac{0.75\Delta h_{Euler}}{U_2^2}}{\left(\dfrac{W_{1T}}{W_2}\right)\left[\left(\dfrac{Z_b}{\pi}\right)\left(1 - \dfrac{D_{1T}}{D_2}\right) + \dfrac{2D_{1T}}{D_2}\right]} \tag{20}$$

And

$$\Delta h_{Euler} = \frac{C_{\theta y}U_y - C_{\theta x}U_x}{U_y^2} \tag{21}$$

Clearance losses were calculated from equation (22).

$$\Delta h_{cl} = 0.6 \frac{\varepsilon}{b_3} \, C_{\theta 3} \left\{ \frac{4\pi}{b_3 Z} \left[\frac{r_{2t}^2 - r_{2h}^2}{(r_3 - r_{2t})\left(1 + \dfrac{\rho_3}{\rho_2}\right)} \right] C_{\theta 3} C_{m,in} \right\}^{\frac{1}{2}} \tag{22}$$

Mixing losses were adopted from Johnston and Dean (1966).

$$\Delta h_{mix} = \frac{1}{1 + tan^2 \, \alpha_3} \left(\frac{1 - \varepsilon_{wake} - b^*}{1 - \varepsilon_{wake}} \right)^2 \frac{C_3^2}{2} \tag{23}$$

Daily and Nece (1960) published a procedure based on the power required to rotate discs in an enclosed space. A primary factor in their analysis was the use of a torque coefficient seen in (26). The stagnation enthalpy rise due to disc friction can be calculated via:

$$\Delta h_{df} = f_{df} \frac{\bar{\rho} r_3^2 U_3^3}{4\dot{m}} \tag{24}$$

Where

$$\bar{\rho} = \frac{\rho_2 + \rho_3}{2} \tag{25}$$

$$f_{df} = \frac{2.67}{Re_{dt}^{0.5}} \text{ for } Re_{df} < 3 \times 10^5$$

or

$$\tag{26}$$

$$f_{df} = \frac{0.0622}{Re_{dt}^{0.2}} \text{ for } Re_{df} > 3 \times 10^5$$

$$Re_{df} = \frac{U_3 r_3}{\nu_3} \tag{27}$$

The losses due to fluid recirculation can be calculated using equation (35) [6].

$$\Delta h_{rc} = 8 \times 10^{-5} \, sinh(3.5\alpha_3^3) \, D_f^2 U_2^2$$

$$\tag{28}$$

The next loss to be considered is leakage losses. This loss correlation was developed by Aungier (1995).

$$\Delta h_{lk} = \frac{\dot{m}_{cl} U_{cl} U_3}{2\dot{m}} \tag{29}$$

Where:

$$U_{cl} = 0.816 \sqrt{\frac{2\Delta P_{cl}}{\rho_3}} \tag{30}$$

$$\Delta P_{cl} = \frac{\dot{m}\{r_3 C_\theta - (r_2 C_{\theta 2})_m\}}{Z \bar{r} \bar{b} L_\theta} \tag{31}$$

$$\bar{r} = \frac{r_2 + r_3}{2} \tag{32}$$

$$\bar{b} = \frac{b_2 + b_3}{2} \tag{33}$$

$$\dot{m}_{cl} = \rho_3 Z \varepsilon L_\theta U_{cl} \tag{34}$$

The subroutine iteratively solves for M_3', β_3, and σ in that order. By assuming an initial value of $1°$ for β_3 and an initial value of 0 for $\frac{\Delta s}{R}$, the M_3' can be solved and then β_3 solved for the correct value. The second iteration of M_3', using the new β_3 value is the isentropic condition. After this Mach number has been solved, the isentropic fluid characteristics and properties will then be used to solve for entropy gain in the system. The relative stagnation enthalpy losses, described above, are calculated, summed, and then used to solve for the entropy gain as a function of stagnation enthalpy loss in the impeller. Seen in (35) [3].

$$\sigma = \exp\left(-\frac{\Delta s}{R}\right) = \left(1 - \frac{\gamma - 1}{\gamma R T_{03}'} U_T^2 \Delta h\right)^{\frac{\gamma}{\gamma - 1}} \tag{35}$$

Once sigma is calculated, the solution of M_3' and β_3 is re-iterated, a new set of fluid properties and flow characteristics calculated. The program tracks Mach number, relative flow angle, and loss correlation convergence during the solution.

Once the three values (outlet relative mach number, relative outlet flow angle, and entropy gain) have appropriately converged the values are saved and passed to the final duct. The flow chart for cold compressor impeller solution can be found in Figure 6.
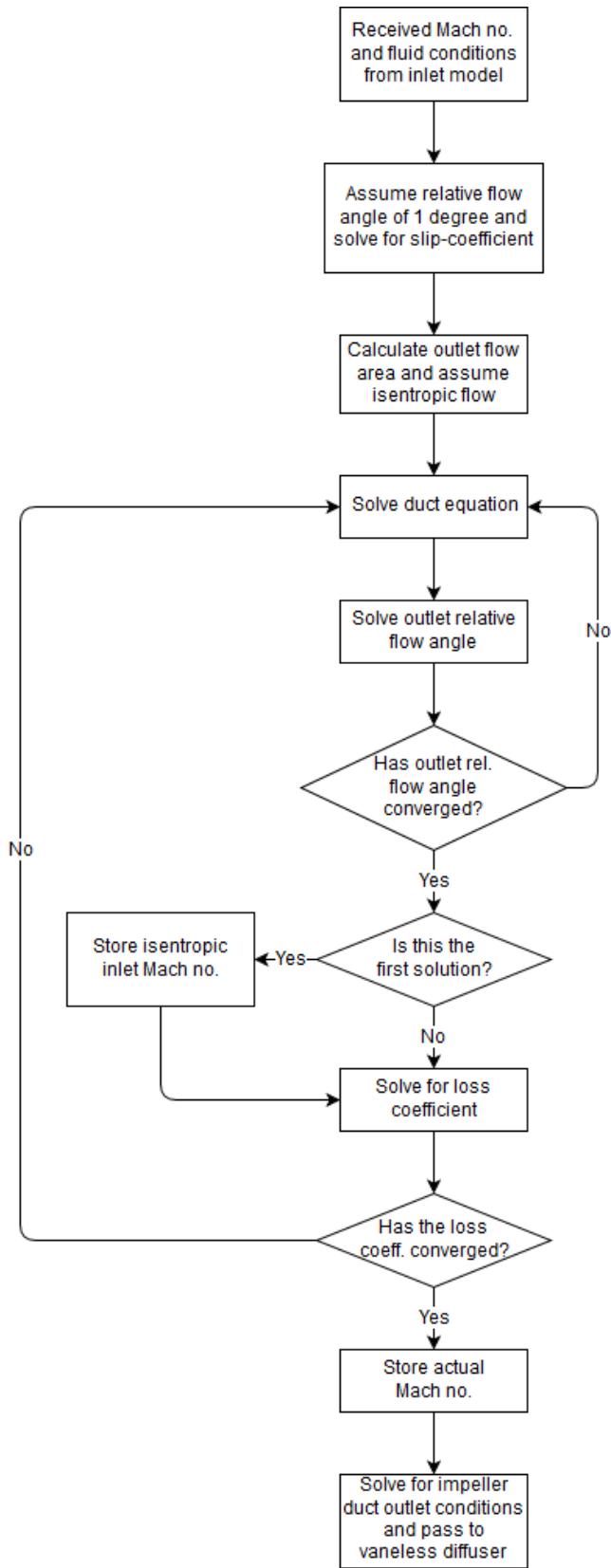
Figure 6. Impeller subroutine flow chart [3]

3.1.7. Cold compressor vaneless diffuser duct

The final duct is used to model unguided swirling flow in a stationary passage. This type of flow occurs after the compressor impeller in the vaneless diffuser. The vaneless diffuser converts the kinetic energy of the high velocity fluid into pressure energy by increasing the expansion ratio across the rotor [3].

Similarly to the impeller duct, the outlet flow angle must be calculated in order to solve for the outlet Mach number. From the angular momentum equation the relationship can be derived for the tangential fluid velocity at the outlet of the vaneless diffuser.

$$\frac{C_{\theta 3}}{C_{\theta 4}} = \frac{r_4}{r_3} + \frac{2\pi C_f \rho_3 C_{\theta 3}(r_4^2 - r_4 r_3)}{\dot{m}} \tag{36}$$

$$C_{\theta 4} = \frac{C_{\theta 3}\dot{m}r_3}{r_4\left(\dot{m} + 2C_f C_{\theta 3}\rho_3 \pi r_3(r_4 - r_3)\right)} \tag{37}$$

After solving for the tangential fluid velocity at the outlet of the vaneless diffuser the absolute flow angle α_4 can be solved using (38) from velocity triangle analysis.

$$\sin \alpha_4 = \frac{C_{\theta 4}\left(1 + \left(\frac{\gamma - 1}{2}\right)M_4^2\right)^{\frac{1}{2}}}{M_4\sqrt{\gamma R T_{04}}} \tag{38}$$

After α_y is solved equation 5.5 can be solved for isentropic Mach number is solved for and isentropic fluid conditions calculated. From there the relative stagnation enthalpy loss due to friction can be solved for using (39).

$$\Delta h = \frac{\left(C_f r_4 \left(1 - \left(\frac{r_3}{r_4}\right)^{1.5}\right)\left(\frac{C_3}{U_T}\right)^2\right)}{1.5 b_3 \cos \alpha_3} \tag{39}$$

By substituting the solution of (39) into (35) the entropy gain for the vaneless diffuser can be

solved and used in solving for M_4. This procedure is iterated until M_4, α_4, and σ, for the vaneless

diffuser, have converged.

The flow chart for cold compressor vaneless diffuser operation can be found in Figure 7.
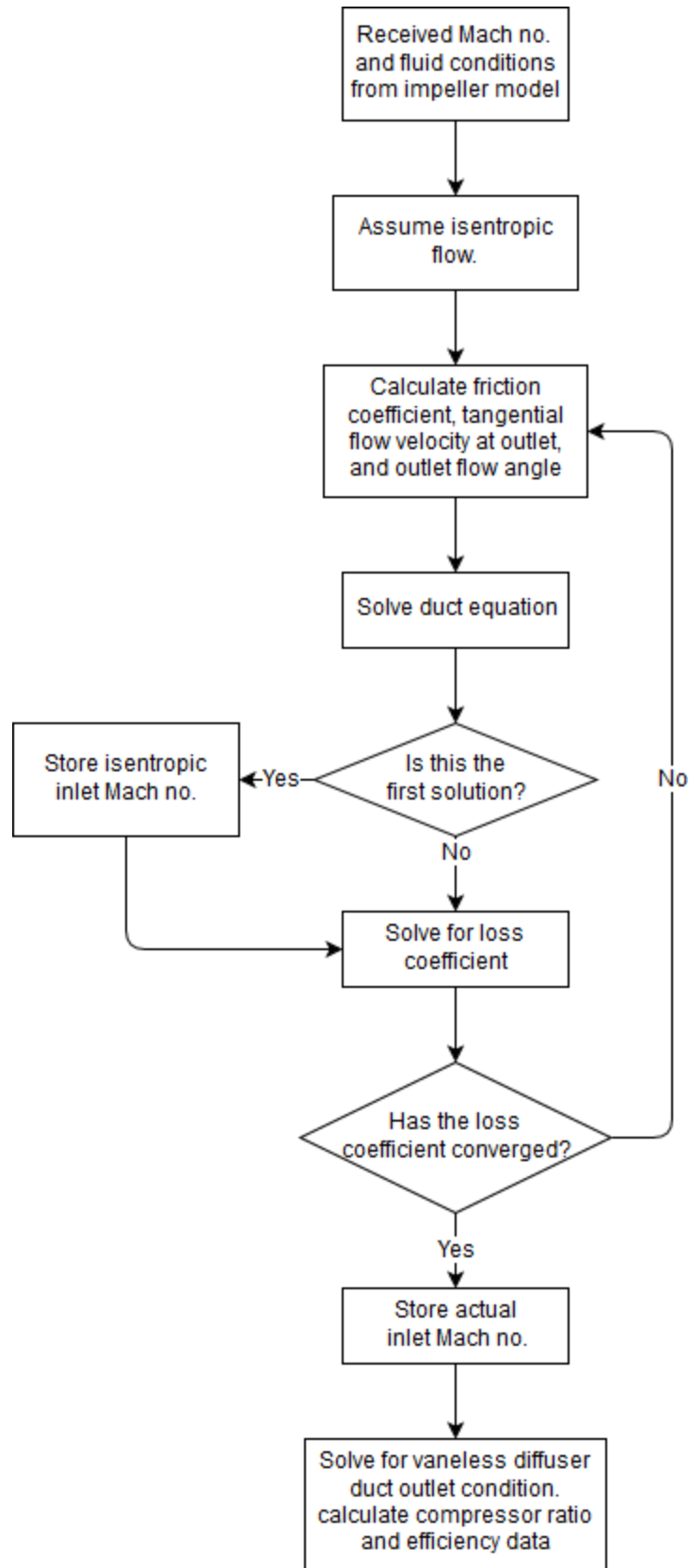
Figure 7. Vaneless diffuser subroutine flow chart [3]

3.1.8. Example Solution of Inlet Duct

An example of the solution of the inlet duct of the cold compressor can be found below.

The inlet to the duct is referred to as node one and the outlet of the duct is referred to as node

two. The geometric constants of the duct can be found be found below in Table 4.

Table 4. Inlet duct geometric constants

Symbol	Value	Unit	Description
$\dot{m}$	5.31	[kg/s]	Mass flow rate
L_{inlet}	0.100	[m]	Length of inlet duct
D_{inlet}	0.280	[m]	Diameter of inlet duct
k_{inlet}	0.005	[m]	Surface roughness of inlet duct

These constants were chosen from [5]. The model output was compared to the original pedigreed

data in order to validate the results.

The fluid constants provided to node one were again used from the Eckardt data set.

Table 5. Inlet duct flow characteristics

Symbol	Value	Unit	Description
P_1	1.013	[bar]	Static pressure at node one
T_1	288	[K]	Static temperature at node one
$\gamma_{1\to2}$	1.667	[-]	Ratio of specific heats through inlet duct
ρ_1	0.169	[kg/m3]	Density at node one

The next step in the analysis is to calculate the inlet stagnation conditions in order to utilize the

duct continuity relationships.

Table 6. Calculated node one Mach number & stagnation conditions

Symbol	Value	Unit	Description
M_1	0.510	[-]	Mach number at node one
P_{01}	1.248	[bar]	Stagnation pressure at node one
T_{01}	313.0	[K]	Stagnation temperature at node one

After the stagnation conditions have been calculated (12) can be iteratively solved for M_2.

$$\frac{\left(5.31\left[\frac{kg}{s}\right]\right)\sqrt{\frac{\left(2077.265\left[\frac{J}{kg\,K}\right]\right)(313[K])}{(1.667[-])}}}{\frac{\pi(0.280[m])^2}{4}(124.753\times10^3[Pa])}$$
$$=\frac{\frac{\pi}{4}(0.280^2-0.090^2)}{\frac{\pi(0.280[m])^2}{4}}\left(1+\frac{(1.667[-])-1}{2}M_2^2\right)^{-\left(\frac{(1.667[-])+1}{2((1.667[-])-1)}\right)}$$

In this case $M_2 = 0.607$. To check the validity of this against continuity, calculated mass flow based on the solved Mach number should be close to the stated inlet mass flow.

$$\dot{m}_{node\,two} = M_2\sqrt{\gamma R T_2}A_2\rho_2 = (0.602[-])\left(979.504\left[\frac{m}{s}\right]\right)\left(0.158\left[\frac{kg}{m^3}\right]\right)\left(\frac{\pi 0.280^2}{4}[m^2]\right)$$
$$= 5.3077\left[\frac{kg}{s}\right]$$

This calculated mass flow rate corresponds to a -0.044% discrepancy with the prescribed mass flow rate and for the purposes of the model is acceptable. Below the node two values can be found.

Symbol	Value	Unit	Description
M_{2s}	0.510	[-]	Mach number at node two
P_{02s}	1.248	[bar]	Stagnation pressure at node two
T_{02s}	312.978	[K]	Stagnation temperature at node two

Symbol	Value	Unit	Description
P_{2s}	0.934	[bar]	*Static pressure at node two*
T_{2s}	278.746	[K]	*Static temperature at node two*

The next step was to calculate the loss coefficient and then resolve the system. The final loss coefficient solved to be:

$$\sigma_{inlet} = 1 - \left(\frac{4C_f L \overline{W}^2 \rho_1}{2DP_{01}}\right)$$

$$= 1 - \frac{4(0.047[-])(0.1[m])\left(\frac{509.374\left[\frac{m}{s}\right] + 596.316\left[\frac{m}{s}\right]}{2}\right)^2 \left(0.169\left[\frac{kg}{m^3}\right]\right)}{2(0.280[m])(124.753 \times 10^3 [Pa])}$$

$$= 0.986[-]$$

The actual values at node two (non-isentropic) can be found in the table below.

Symbol	Value	Unit	Description
M_2	0.623	[-]	*Mach number at node two*
P_{02}	1.230	[bar]	*Stagnation pressure at node two*
T_{02}	312.978	[K]	*Stagnation temperature at node two*
P_2	0.907	[bar]	*Static pressure at node two*
T_2	277.142	[K]	*Static temperature at node two*

And again the mass flow balance on the inlet duct produces a discrepancy of -0.043%.

3.1.9. Validation of Cold Compressor Model

The cold compressor model was validated using the pedigreed empirical data from Eckardt [5]. The isentropic efficiency and pressure ratio results from the model described above were compared to Eckardt's results for a given mass flow rate and wheel speed.

In order to reproduce the results of the Eckardt data, the loss coefficients for the impeller needed to be selected. These loss coefficients are used in the Whitfield and Baines model to calculate the

entropy gain σ for use in the generalized duct equation, for the impeller model. Correlations to loss coefficients are readily available in published literature, and must be selected properly for a given type of compressor and modeling application. Oh *et al* [6] used a separate modeling approach to reproduce Eckardt's results. That study used a series of different loss coefficients and compared the accuracy of the predictive models against the Eckardt data. The optimal loss coefficient set in Oh *et al* was used in this study. Figure 8 shows the results of applying the model proposed in this document to the Eckardt data using loss coefficients from [6].
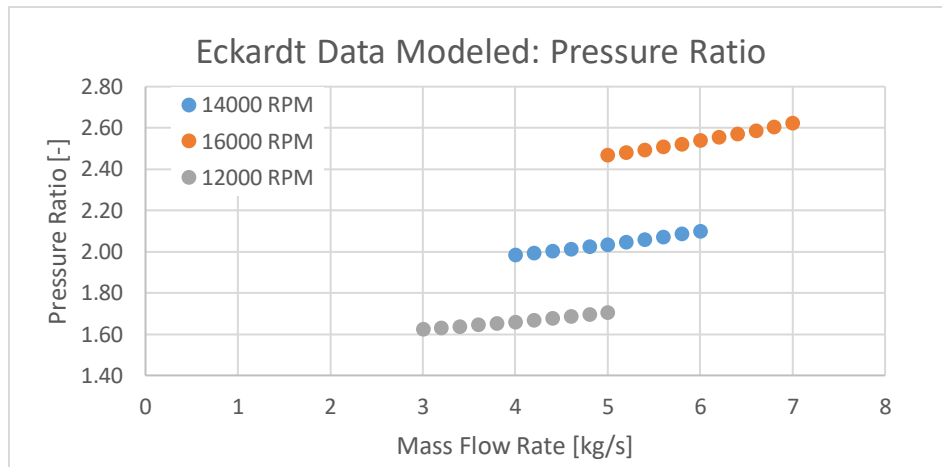
Figure 8. Whitfield and Baines' model pressure ratio results applied to Eckardt impeller

For each wheel speed the results correlate well with both studies. When compared to the design case of 5.31 kg/s at 14000 rpm the isentropic efficiency of the model matched [5] to 2.9% and the pressure ratio matched within 2.2%.
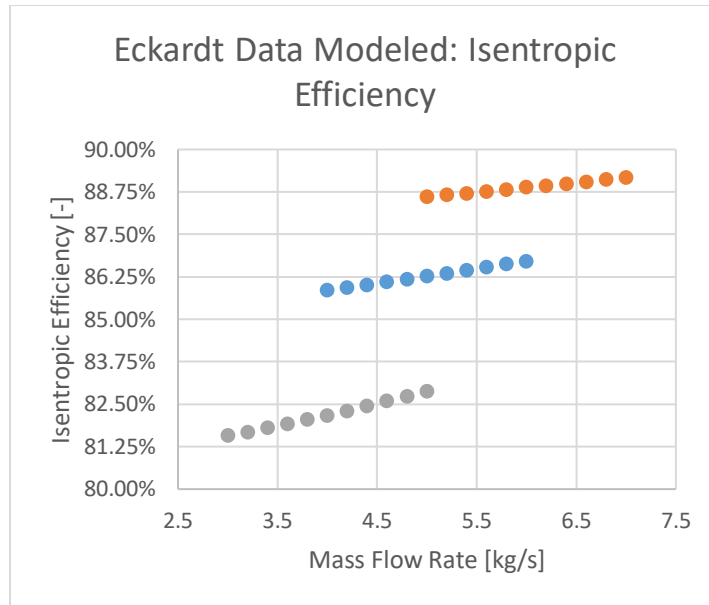
Figure 9. Whitfield and Baines' model isentropic efficiency results applied to Eckardt impeller

The only discrepancy to note is the 12000 rpm case in Figure 9. [5] and [6] have estimates between 85% and 88% where the proposed model in this paper predict between 81% and 83%. The trend follows the shape of the selected loss coefficients from [6] however shows a roughly 4% discrepancy. With largely consistent agreement across the compressor map, this was deemed acceptable for the application of this model.

3.1.10. Impeller Wheel Design and Characterization

Due to the restrictive nature of the FRIB cryogenic compressor manufacturer, the actual impeller geometry for the FRIB compressors was not available. As a result, it was necessary to design a representative impeller geometry for this model. This impeller was designed with helium used as the working fluid.

The procedure used for estimating the wheel geometry was adopted from [3]. The procedure requires the specification of a series of design parameters for the wheel geometry. These can be seen in Table 7.

Table 7. Prescribed geometric values for the impeller design process

Symbol	Value	Unit	Description
μ_{slip}	0.85	[-]	*slip factor*
$\eta_{s,impeller}$	0.85	[-]	*impeller total to total efficiency*
$\eta_{s,stage}$	0.80	[-]	*stage total to total efficiency*
$\beta_{b,2}$	-60.0	[deg]	*inlet shroud relative flow angle*
$\nu_{h\to s}$	0.40	[-]	*inducer hub to shroud radius ratio*
α_3	65.0	[deg]	*impeller discharge absolute flow angle*
$\dfrac{r_{2s}}{r_3}$	0.35	[-]	*impeller inducer shroud to discharge tip radius ratio*
$\beta_{b,3}$	0.0	[deg]	*discharge blade angle*

Along with the prescribed data geometric data it is necessary to specify the stagnation pressure and stagnation temperature at the cold compressor inlet. This allows the procedures to develop the estimate geometry non-dimensionally. The non-dimensional output of the procedure are the head coefficient, flow coefficient, non-dimensional mass flow, as well as static and stagnation specific speed of the design conditions. The dimensional output is the discharge area, the wheel diameter, the discharge blade height, the inducer shroud radius, and the inducer hub radius. With these data points the impeller geometry can be approximated.

The prescribed geometric values, specifically $\frac{r_{2s}}{r_3}$ can be modified to scale the size of the impeller. The FRIB installed impeller wheel was expected to have a diameter of roughly 7.5 *in* and a desired polytropic efficiency of *at least* 78% at the design condition. Using this as reference values for sizing the impeller the following impeller geometries were produced.

Table 8. Estimated FRIB cold compressor impeller dimensions

Symbol	Value	Unit	Description
r_{2s}	0.044 (1.731)	$[m]$ ($[in]$)	*inducer shroud radius*
r_{2h}	0.018 (0.692)	$[m]$ ($[in]$)	*inducer hub radius*
r_3	0.098 (3.847)	$[m]$ ($[in]$)	*discharge radius*
h_b	0.005662 (0.223)	$[m]$ ($[in]$)	*discharge blade height*
A_3	3.00E-02 (46.490)	$[m^2]$ ($[in^2]$)	*discharge area*

The estimated FRIB impeller dimensions satisfy the $\sim 7.5\ [in]$ wheel diameter by producing a wheel with a diameter of $7.694\ [in]$. After these geometries were inserted into the compressor model the following compressor map was produced.
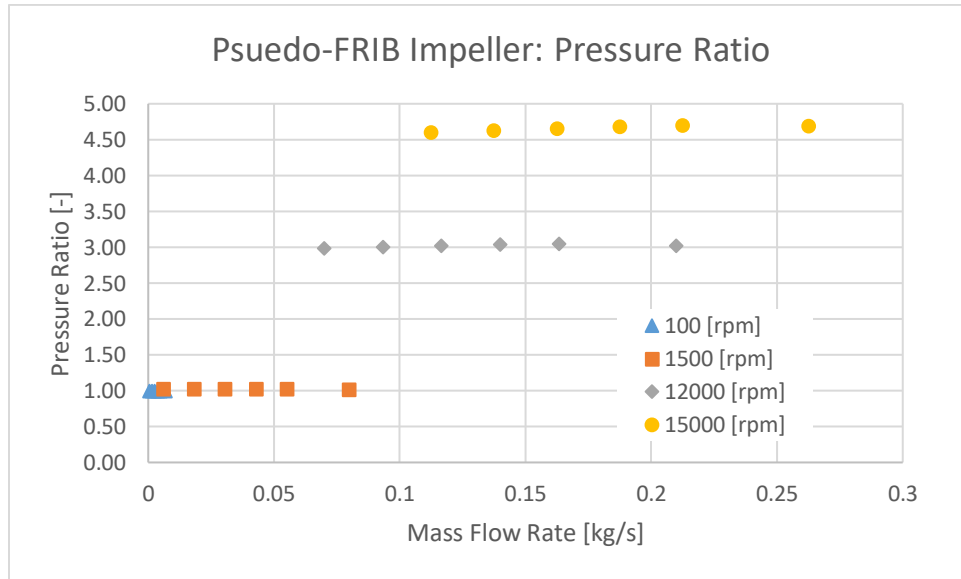
Figure 10. Pressure ratio as function of angular velocity (rpm) and mass flow rate

Regarding Figure 10 it is evident that this impeller will be able to provide the desired pressure ratios for the design case. When combined with Figure 11, the pseudo-FRIB impeller is shown to provide the mass flow rates as the desired pressure ratio and produce efficiencies at or above the expected values for the manufactured impeller.
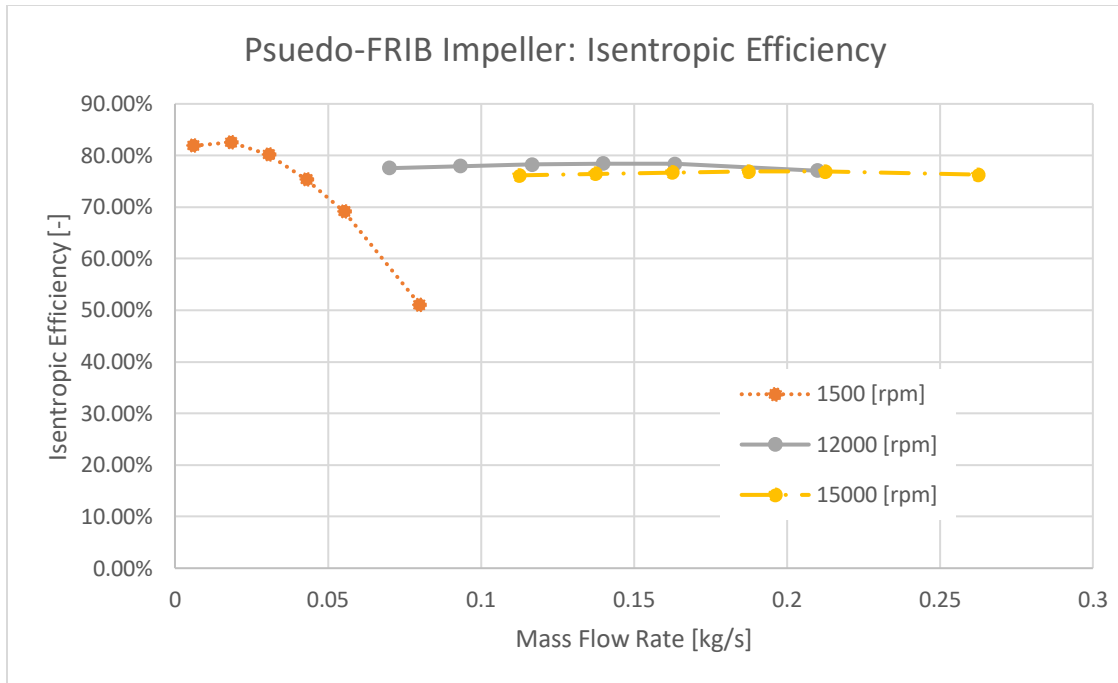
Figure 11. Isentropic efficiency of estimated FRIB impeller focused around the design conditions

This characterization of the FRIB estimated (pseudo-FRIB) impeller is valid for this modeling application.

3.2.4.5 K to 2 K Heat Exchanger

Within the cryo-modules there is a heat exchanger that is used to recover the refrigeration of the low-pressure helium that is being removed from the dewar.

The 4.5 K to 2 K Heat Exchanger (HX) model is a variable specific heat model that assumes the HX metal is in thermal equilibrium with the high and low-pressure streams. It is also assumed there is no pressure drop for either stream within the heat exchanger.

The need for a variable specific heat model in this model is due to the variation of specific heat in the application range of this heat exchanger.

52
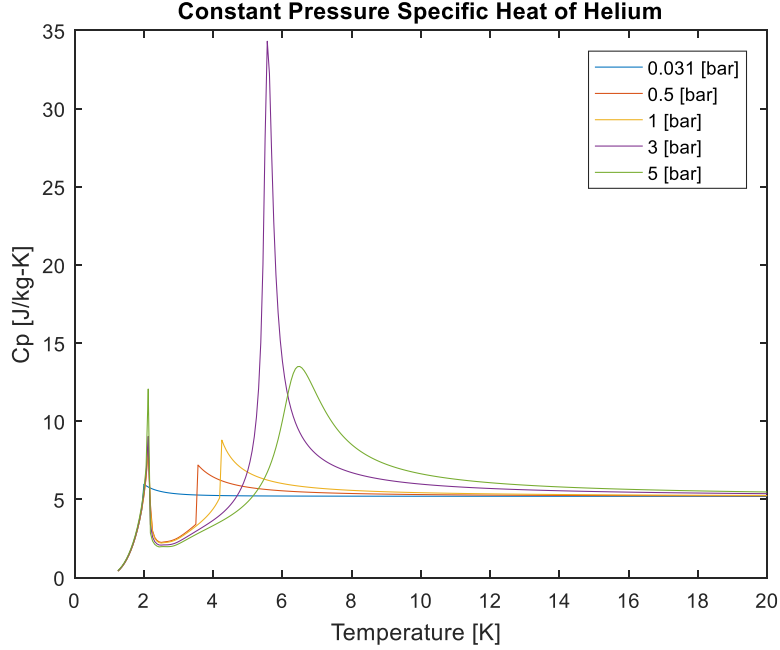
Figure 12. Variation of constant pressure specific heat for a range of pressures and temperatures

Figure 12 shows how significant the specific heat will vary over the possible application range of the sub-atmospheric heat exchanger.

The heat exchanger is divided into 10 divisions in order to model a variable specific heat component [7]. At each division, the heat rate equation and the energy balance equation are solved.

$$-\theta_i^* \cdot T_{h,i} + \theta_i^* \cdot T_{l,i} + T_{h,i+1} - T_{l,i+1} = 0 \tag{40}$$

$$-T_{h,i} + C_{R,i}^* \cdot T_{l,i} + T_{h,i+1} - C_{R,i+1}^* \cdot T_{l,i+1} = 0 \tag{41}$$

(40) shows the rate equation and (41) shows the energy equation for each heat exchanger division (i).

To initialize the solution, the stream temperature profiles are solved as though the stream is of a constant heat capacity. This is done by calculating heat capacity values for the high and low-

pressure streams based on $T_{h,1}$ and $T_{l,N}$, which are known values. The overall heat capacity ratio

for the heat exchanger is calculated using (42).

$$C_{R,est}^* = \frac{C_{\min}}{C_{\max}} \tag{42}$$

Where $C_{\min}$ is the minimum stream heat capacity and $C_{\max}$ is the maximum stream heat capacity

for the initialization problem. From there the initial dimensionless temperature difference for the

entire heat exchanger can be calculated using (43).

$$\theta_{est}^* = e^{(1-C_{R,est}^*)\cdot NTU} = e^{(1-C_{R,est}^*)\cdot\left(\frac{UA}{C_{l,N}}\right)} \tag{43}$$

This is then used to calculate the initial temperature difference of the heat exchanger, see (44).

$$\Delta T_{hl,est,1} = \frac{1-C_{R,est}^*}{\theta_{est}^*-C_{R,est}^*}\cdot\Delta T_{hl,est,i} = \frac{1-C_{R,est}^*}{\theta_{est}^*-C_{R,est}^*}\cdot\left(T_{h,1}-T_{l,N}\right) = T_{h,1}-T_{l,1} \tag{44}$$

The initial temperature difference is used to solve for the low-pressure stream temperature at the

first division (i=1). The total temperature difference of the low-pressure stream is calculated and

divided into the low-pressure temperature difference per division. This is used to initialize the

low pressure stream temperature throughout the heat exchanger. To initialize the high-pressure

stream (44) is multiplied by (45)

$$\theta_{est,i}^* = e^{(1-C_{R,est}^*)\cdot(NTU)_i} = e^{(1-C_{R,est}^*)\cdot\left(\frac{(UA)_i}{C_{l,N}}\right)} \tag{45}$$

$$\Delta T_{hl,est,i+1} = \Delta T_{hl,est,i}\cdot\theta_{est,i}^* = T_{h,i+1}-T_{l,i+1} \tag{46}$$

(45) shows the dimensionless temperature difference per division. This gives the temperature

differential at each heat exchanger division. The high-pressure stream temperature profile is then

solved for by reordering (46) and solving for $T_{h,i+1}$. Figure 1 shows the initialized strem

temperature profiles as a function of thermal length.

Figure 13. Initialized stream temperature profiles as a function of NTUs

After the temperature profile has been initialized, the system can be analyzed at the division level. The enthalpy, high to low-pressure temperature difference, forward temperature difference, heat input per division, and heat capacity were then calculated for each stream at each division. $C_{R,i}^*$ and θ_i^* were calculated for each division.

The matrix was then built to solve for temperatures at each node in the system. Figure 14 shows an example matrix used in solving the temperature profile for a 3 division heat exchanger.

$$\begin{bmatrix} \theta_1^* & 1 & -1 & & & \\ C_{R,1}^* & 1 & -C_{R,1}^* & & & \\ & -\theta_2^* & \theta_2^* & 1 & -1 & \\ & -1 & C_{R,2}^* & 1 & -C_{R,2}^* & \\ & & & -\theta_3^* & \theta_3^* & 1 \\ & & & -1 & C_{R,3}^* & 1 \end{bmatrix} \cdot \begin{bmatrix} T_{l,1} \\ T_{h,2} \\ T_{l,2} \\ T_{h,3} \\ T_{l,3} \\ T_{h,4} \end{bmatrix} = \begin{bmatrix} \theta_1^* \cdot T_{h,1} \\ T_{h,1} \\ 0 \\ 0 \\ T_{l,4} \\ C_{R,3}^* \cdot T_{l,4} \end{bmatrix}$$

Figure 14. 3 division, 4 node example system for solving the temperature profile for a heat exchanger

In Matlab the temperature vector was solved for using a direct method, *mldivide()*. The new

stream temperatures then needed to be introduced into the current solution and the solution

iterated until the maximum division heat exchange discrepancy reached 0.1% (see (47)).

$$dq_{lh,i} = abs\left[\frac{dq_{h,i}}{dq_{l,i}} - 1\right] \tag{47}$$

The current stream temperature profiles were updated using a relaxation method shown in (48).

$$\left(T_{p,i}\right)_{new,relaxed} = (1 - \alpha) \cdot \left(T_{p,i}\right)_{old} + \alpha \cdot \left(T_{p,i}\right)_{new} \tag{48}$$

Where in this solution the relaxation value α was set to 0.35. A fully iterated stream temperature

profile can be seen in Figure 15.
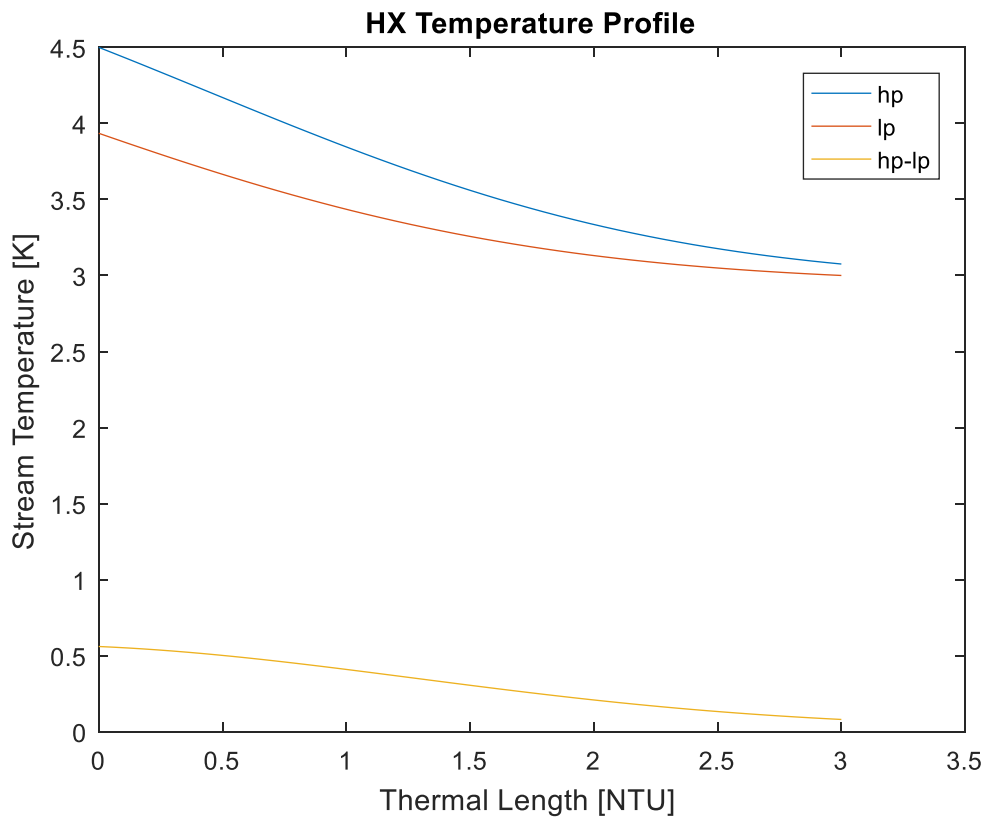
Figure 15. Fully iterated stream temperature profiles for the sub-atmospheric heat exchanger

This model solves for the outlet high-pressure and low-pressure stream temperatures while allowing stream heat capacity of each stream to vary with temperature. This is done for a single time step in the overall model solution.

3.2.1. Validation of the Sub-Atmospheric Heat Exchanger Model

The sub-atmospheric heat exchanger model is validated two fold. The first is an overall stream to stream comparison of heat transfer.

$$\frac{\Sigma q_l}{\Sigma q_h} - 1 < 0.1\% \tag{49}$$

The second validation path is the iteration to iteration change of the division stream temperatures. The maximum change in stream temperature in any division should be below 1%. That calculation can be seen in (50).

$$\frac{T_{stream,division,i+1} - T_{stream,division,i}}{T_{stream,division,i}} < 0.01 \tag{50}$$

Those two validation tactics satisfied qualifies the sub-atmospheric heat exchanger as validated.

3.3. Dewar Depressurization Model

The purpose of this study is to model system transiency as the system is "pumped-down". This "pump-down" is a depressurization of the cryo-module dewar. The need for depressurization stems from the need to move from a state where the cold compressors have been deactivated or "tripped" to desired operating conditions.

Prolonged exposure to the cold compressor "tripped" state will cause the helium bath to warm and transition to a saturated liquid/saturated vapor mixture, at a pressure near 1 bar. The subroutine models pressure with time navigating from 1.25 bar to 0.031 bar. The goal being to track the pressure profile and time to achieve the desired pressure in the dewar.

The major assumptions of this model are as follows:

- Only saturated liquid and saturated vapor are present during the process

- The volume of saturated liquid will remain constant

- Saturated vapor mass flow out of the dewar is an independent variable and constant during the process

- Heat goes solely into the liquid

There are four governing equations of the dewar depressurization model. The assumptions that the total dewar volume is constant, constant saturated liquid volume, mass continuity, energy conservation of the system produce relationships to solve for $\dot{m}_l$, $\dot{m}_v$, $\dot{m}_s$, and $\dot{p}$.

where: $\dot{m}_l$ is the rate of change of saturated liquid mass [g/s]

$\dot{m}_v$ is the rate of change of saturated vapor mass [g/s]

$\dot{m}_s$ is the supply mass flow rate [g/s]

$\dot{p}$ is the rate of change of pressure in the dewar [bar]

The cryo-module containers are rigid and therefore their volume is fixed.

$$V = V_l + V_v = const. \tag{51}$$

$$\dot{V} = 0 = m_l \dot{v}_l + m_v \dot{v}_v + \dot{m}_l v_l + \dot{m}_v v_v \tag{52}$$

Recall that both the vapor and liquid phases are saturated, temperature is dependent on pressure. Therefore, saturated liquid and saturated vapor specific volumes are a function of only pressure, in the case of this dewar depressurization.

$$\dot{v}_l = \dot{p} \left(\frac{dv_l}{dp} \right)_{sat} \tag{53}$$

$$\dot{v}_v = \dot{p} \left(\frac{dv_v}{dp} \right)_{sat} \tag{54}$$

So the volume constraint equation expands to

$$0 = \dot{p} \left[m_l \left(\frac{dv_l}{dp} \right)_{sat} + m_v \left(\frac{dv_v}{dp} \right)_{sat} \right] + \dot{m}_l v_l + \dot{m}_v v_v \tag{55}$$

To expand upon the $\left(\frac{dv}{dp} \right)_{sat}$ terms in the volume constraint equation, a derivation of the total

specific volume differential is necessary. The saturated vapor and saturated liquid evaluations

will be analogous.

$$dv_l = \left(\frac{\partial v_l}{\partial p} \right)_T dp + \left(\frac{\partial v_l}{\partial T} \right)_p dT \tag{56}$$

$$\left(\frac{dv_l}{dp} \right)_{sat} = \left(\frac{\partial v_l}{\partial p} \right)_T + \frac{\left(\frac{\partial v_l}{\partial T} \right)_p}{\left(\frac{dp}{dT} \right)_{sat}} \tag{57}$$

$$\left(\frac{dv_l}{dp} \right)_{sat} = v_l \left(\frac{\beta_l}{\left(\frac{dp}{dT} \right)_{sat}} - \kappa_{T,l} \right) \tag{58}$$

where β_l is the saturated liquid volume expansivity [1/K]

$\kappa_{T,l}$ is the saturated liquid isothermal compressibility [1/Pa]

$\left(\frac{dp}{dT} \right)_{sat}$ is the slope of the vapor pressure curve

This relationship will be used in the energy conservation equation.

The rate of change of energy of the saturated liquid and saturated vapor components in the dewar are as follows:

$$\frac{dU}{dt} = \frac{d}{dt}(m_l u_l + m_v u_v) = m_l \dot{u}_l + m_v \dot{u}_v + \dot{m}_l u_l + \dot{m}_v u_v \tag{59}$$

Similar to the volume constraint the rate of change of saturated internal energy is a function of only pressure.

$$\dot{u}_l = \dot{p}\left(\frac{du_l}{dp}\right)_{sat} \tag{60}$$

$$\dot{u}_v = \dot{p}\left(\frac{du_v}{dp}\right)_{sat} \tag{61}$$

Evaluating the total specific internal energy differential allows for the expansion of the $\left(\frac{du_l}{dp}\right)_{sat}$ terms.

$$du_l = \left(\frac{\partial u_l}{\partial T}\right)_v dT + \left(\frac{\partial u_l}{\partial v}\right)_T dv \tag{62}$$

$$\left(\frac{du_l}{dp}\right)_{sat} = \frac{\left(\frac{\partial u_l}{\partial T}\right)_v}{\left(\frac{dp}{dT}\right)_{sat}} + \left(\frac{\partial u_l}{\partial v}\right)_T \left(\frac{dv_l}{dp}\right)_{sat} \tag{63}$$

$$\left(\frac{du_l}{dp}\right)_{sat} = \frac{\left(\frac{\partial u_l}{\partial T}\right)_v}{\left(\frac{dp}{dT}\right)_{sat}} + \left[T\left(\frac{\partial p}{\partial T}\right)_{vl} - p\right]\left(\frac{dv_l}{dp}\right)_{sat} \tag{64}$$

$$\left(\frac{du_l}{dp}\right)_{sat} = \frac{c_{v,l}}{\left(\frac{dp}{dT}\right)_{sat}} + \left[\frac{T\beta}{\kappa_{T,l}} - p\right]\left(\frac{dv_l}{dp}\right)_{sat} \tag{65}$$

where $c_{v,l}$ is the constant volume specific heat of the saturated liquid

$\left(\frac{dv_l}{dp}\right)_{sat}$ is derived above

60

T is the saturation temperature at the given pressure

And again the saturated vapor is analogous to the saturated liquid. To simplify the derivation, an intermediate variable $\tilde{U}$ is created.

$$\tilde{U} = \tilde{U}_l + \tilde{U}_v \tag{66}$$

$$\tilde{U}_l \equiv m_l \left(\frac{du_l}{dp}\right)_{sat} \tag{67}$$

$$\tilde{U}_v \equiv m_v \left(\frac{du_v}{dp}\right)_{sat} \tag{68}$$

Therefore, the rate of change of internal energy of the saturated liquid and vapor components simplifies.

$$\frac{dU}{dt} = \dot{p}\tilde{U} + \dot{m}_l u_l + \dot{m}_v u_v \tag{69}$$

The energy conservation of the dewar is the difference between the enthalpy of the supply flow being added to the dewar and the enthalpy of the saturated vapor flow leaving the dewar plus the heat into the dewar.

$$\frac{dU}{dt} = \dot{m}_s h_s - \dot{m}_g h_v + q \tag{70}$$

Setting the energy conservation equal to the rate of change of internal energy yields:

$$\dot{p}\tilde{U} + \dot{m}_l u_l + \dot{m}_v u_v = \dot{m}_s h_s - \dot{m}_g h_v + q \tag{71}$$

where the specific enthalpy of the supply stream is a saturated mixture shown as:

$$h_s = (1 - x)h_l + x h_v \tag{72}$$

The mass continuity of the dewar shows that the rate of change of saturated liquid mass and the rate of change of saturated vapor mass is related to the mass flow rate of supply fluid and the mass flow rate of saturated vapor removed from the dewar.

$$\dot{m}_l + \dot{m}_v = \dot{m}_s - \dot{m}_g \tag{73}$$

This equation is used to solve for supply enthalpy and then substitute into the energy equation to yield:

$$p\tilde{U} + \dot{m}_l(u_l - h_s) + \dot{m}_v(u_v - h_s) = -\dot{m}_g(h_v - h_s) + q \tag{74}$$

Solving for $\dot{p}$:

$$\dot{p} = \frac{\dot{m}_g\big(h_v(x-1) - h_l(x-1)\big) + \dot{m}_l(xh_v - h_l(x-1) - u_l) + \dot{m}_v(h_v u_v(x-1) - h_v u_v x) + q}{\tilde{U}} \tag{75}$$

Using the following relationships, the solution for $\dot{p}$ can be simplified.

$$\lambda = h_v - h_l \tag{76}$$

Returning to the constraint that liquid level must remain constant

$$\dot{V}_l = 0 = m_l \dot{v}_l + \dot{m}_l v_l \tag{77}$$

$$\dot{m}_l = -m_l \frac{\dot{v}_l}{v_l} = -\widetilde{m}_l \frac{\dot{p}}{p} \tag{78}$$

where $\widetilde{m}_l$ is an intermediary variable defined as:

$$\widetilde{m}_l \equiv m_l \frac{p}{v_l}\left(\frac{dv_l}{dp}\right)_{sat} \tag{79}$$

Similarly, because the total volume of the dewar is constant in conjunction with the saturated liquid level being constant, the saturated vapor volume must remain constant.

$$\dot{V}_v = 0 = m_v \dot{v}_v + \dot{m}_v v_v \tag{80}$$

$$\dot{m}_v = -m_v \frac{\dot{v}_v}{v_v} = -\widetilde{m_v} \frac{\dot{p}}{p} \tag{81}$$

Again, where $\widetilde{m_v}$ is an intermediary variable defined as:

$$\widetilde{m_v} \equiv m_v \frac{p}{v_v} \left(\frac{dv_v}{dp}\right)_{sat} \tag{82}$$

Lastly by creating the intermediary variables $\widetilde{V}_l$ and $\widetilde{V}_v$ below the final equation is generated.

$$\widetilde{V}_l = \widetilde{m_l}\widetilde{v_l} \tag{83}$$

$$\widetilde{V}_v = \widetilde{m_v}\widetilde{v_v} \tag{84}$$

where,

$$\widetilde{v_l} = v_l + \frac{x\lambda}{p} \tag{85}$$

$$\widetilde{v_v} = v_v - \frac{(1-x)\lambda}{p} \tag{86}$$

The final equation for the time rate of change of pressure in the dewar is:

$$\dot{p} = \frac{q - \dot{m}_g(1-x)\lambda}{\widetilde{U} + \widetilde{V}_l + \widetilde{V}_v} \tag{87}$$

The major goal of this model is to prescribe initial conditions of pressure $p(t \to 0)$, mass flow

rate of vapor out of the dewar $\dot{m}_g$ and solve for the time rate of change of pressure in the dewar

$\dot{p}$, the time rate of change of saturated liquid mass in the dewar $\dot{m}_l$, the time rate of change of

saturated vapor mass in the dewar $\dot{m}_v$, and the supply fluid flow rate into the dewar $\dot{m}_s$.

The method used to solve for pressure with respect to time is to numerically integrate the $\dot{p}$

equation.
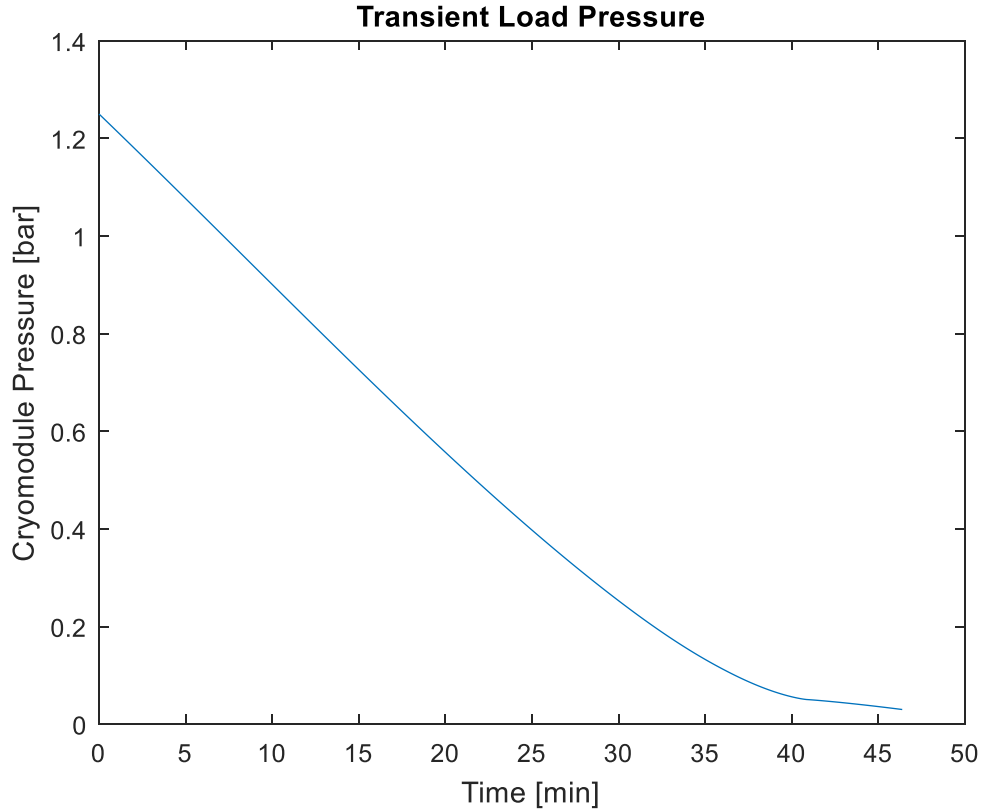
Figure 16. SRF cavity (cryo-module) load pressure as a function of time given constant 15 [g/s] mass removal process path

Figure 16 shows an example solution for pressure as a function of time using the initial conditions of $\dot{m}_g = 15$ [g/s], $p(t \to 0) = 1.25$ [bar], a load of $q_{load} = 21$ [W], and a time step of 0.5 [s]. This process path is solved independently of other components in the system and serves only to show how the dewar will respond in isolation.

3.3.1. Cryo-module Dewar Validation

The cryo-module dewar model is validated by solving the equations that form the three primary equations of the model. The volume constraint, mass balance, and energy balance are solved using the calculated variables. Those can be referenced in the following equations.

$$\frac{dV}{dt} = m_l \dot{v}_l + m_v \dot{v}_v + \dot{m}_l v_l + \dot{m}_v v_v = 0 \tag{88}$$

$$\dot{m}_l + \dot{m}_v - \dot{m}_s + \dot{m}_g = 0 \tag{89}$$

$$\left(\dot{m}_s h_s - \dot{m}_g h_v + q \right) - \left(m_l \dot{u}_l + m_v \dot{u}_v + \dot{m}_l u_l + \dot{m}_v u_v \right) = 0 \tag{90}$$

Recall that the rate of vapor being removed from the cryo-module ($\dot{m}_g$) is different from the rate

of change of vapor mass in the cryo-module ($\dot{m}_v$). Also any derivative value that was solved for

using closed form solutions in the derivation are checked using values calculated through finite

difference of the component values.

$$\left(\frac{dv}{dp} \right)_{sat} = v \left(\frac{\beta}{\left(\frac{dp}{dT} \right)_{sat}} - \kappa_T \right) \cong \frac{v_{p+\Delta p} - v_p}{\Delta p} \tag{91}$$

$$\left(\frac{du}{dp} \right)_{sat} = \frac{c_v}{\left(\frac{dp}{dT} \right)_{sat}} + \left(\frac{T\beta}{\kappa_T} - p \right) \left(\frac{dv}{dp} \right)_{sat} \cong \frac{u_{p+\Delta p} - u_p}{\Delta p} \tag{92}$$

The discrepancy of the closed form derivative solutions seen in (91) and (92) to the finite

difference solution should be below 0.01%.

3.4. Return Volume Model

The purpose of this portion is to model the changing superheated vapor helium conditions within

the transfer lines of the distribution system. The transfer lines in question are vacuum jacketed

pipes containing a series of helium supply and return lines to the LINAC.

The model this paper will discuss handles changing conditions of helium traveling from the

LINAC tunnel, returning to the 2K cold box. The major concern of the transfer line is: how much

heat will leak into the helium vapor traveling back from the cryo-modules how different will the

conditions be at the outlet of the transfer lines. The cold compressor model reads in geometry,

operating, and fluid conditions for the compressor and solve the system in a series of three subroutines.

Table 9. Geometry and operating inputs to transfer line model

Symbol	Unit	Description
GC	[m^{-4}]	Geometry constant for return line volume
dt	[s]	Differential time for integration
$\dot{m}_{out}$	[kg/s]	Mass flow through cold compressor
T_{TL}	[K]	Initial temperature of transfer line
P_{TL}	[bar]	Initial pressure of transfer line
q_s	[W]	Heat flux into transfer line
p_{in}	[bar]	Pressure of fluid leaving S.A.H.E. entering return volume
T_{in}	[K]	Temperature of fluid leaving S.A.H.E. entering return volume

Table 9 shows the geometric, fluid condition, and thermal condition inputs to the return line model.

3.4.1. Friction Model and Control Volume Analysis

The major assumptions of this model are as follows:

- The transfer line is of a constant diameter

- The thermal boundary condition is a constant heat in-leak over the transfer line

The mass flow rate is derived from the Darcy-Weisbach equation. Which relates pressure drop to mass flux in pipe flow.

$$\Delta p = \frac{1}{2}\frac{G_{in}^2}{\rho_{in}}\frac{4fL}{D_h} \tag{93}$$

$$\Delta p = \frac{1}{2}\frac{\left(\frac{\dot{m}_{in}}{A_{in}}\right)^2}{\rho_{in}}\frac{4fL}{D_h} \tag{94}$$

$$\dot{m}_{in} = \sqrt{\frac{2\Delta p D_h \rho_{in} A^2}{4fL}} \tag{95}$$

In (93), Δp is the initial difference in pressure of the inlet to the transfer line model to the pressure in the transfer line control volume. The geometric values of length, diameter, and friction factor found in equations (93) through (95) are simplified using (96).

$$\Delta p = GC \times \dot{m}_{in}^2 / \rho_{TL} \tag{96}$$

This equation coupled with the knowledge that the return line volume is split into three lengths, each handling 50 g of helium at 0.030 bar, with a 1 mbar pressure drop at the steady state condition and an estimated temperature change of:

$$\Delta T = \frac{q_s}{\dot{m}_{in} c_p} \tag{97}$$

By solving (97) for the steady state temperature at node 5, equation (96) can now be solved. That is, the density in equation (96) is taken at the inlet pressure and the average temperature (average of node 4 and node 5 temperatures).

$$GC = \frac{\Delta p \times \rho_{TL}}{\dot{m}_{in}} \tag{98}$$

Equation (98) allows for the solution of the return line geometric constant GC. With this, equation (96) can be reordered to solve for mass flow into the return line volume.

From here, a control volume analysis takes place to evaluate the changing conditions in the transfer line. Mass conservation is applied to solve for the rate of change of density in the transfer line volume.

$$\dot{m}_{cv} = \dot{m}_{in} - \dot{m}_{out} \tag{99}$$

$$\dot{m}_{cv} = \frac{d}{dt} [\rho_{cv} V_{cv}] = \dot{\rho}_{cv} V_{cv} + \rho_{cv} \dot{V}_{cv} = \dot{\rho}_{cv} V_{cv} + \rho_{cv} \cdot 0 = \dot{\rho}_{cv} V_{cv} \tag{100}$$

$$\dot{\rho}_{cv} = \frac{\dot{m}_{in} - \dot{m}_{out}}{V_{cv}} \tag{101}$$

The energy rate and energy conservation equations allow for the rate of change of internal

energy in the transfer line control volume to be calculated.

$$\frac{dU_{cv}}{dt} = \dot{m}_{in} h_{in} - \dot{m}_{out} h_{out} + q_{leak} \tag{102}$$

$$\frac{dU_{cv}}{dt} = \frac{d}{dt}[m_{cv} u_{cv}] = \frac{d}{dt}[\rho_{cv} V_{cv} u_{cv}] = V_{cv} \dot{\rho}_{cv} u_{cv} + V_{cv} \rho_{cv} \dot{u}_{cv} \tag{103}$$

$$\dot{u}_{cv} = \frac{\dot{m}_{in} h_{in} + q_{leak} - u_{cv}(V_{cv} \dot{\rho}_{cv} + \dot{m}_{out}) - (\dot{m}_{out} p_{cv} v_{cv})}{V_{cv} \rho_{cv}} \tag{104}$$

(101) and (104) are then integrated to solve for the density and internal energy at the next time.

With these two intensive properties known we can calculate pressure and temperature of the next

time. The flow chart for return transfer line operation can be found in Figure 17.
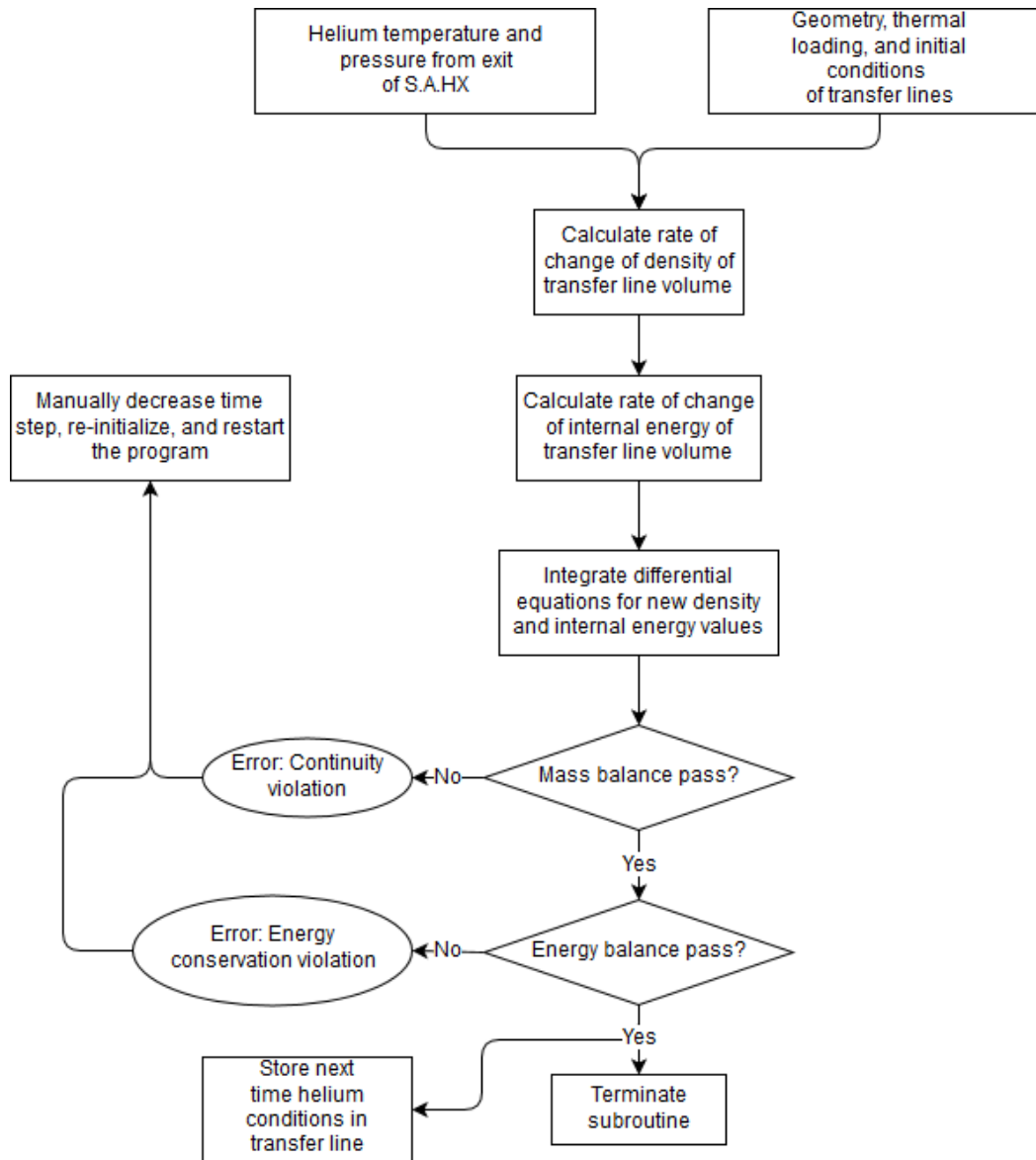
Figure 17. Return Line Flow Chart

3.4.2. Return Line Volume Validation

The return line volume is validated by checking for mass balance and energy balance in the

system.

$$\dot{m}_{depressurization} + \dot{m}_5 + \dot{m}_4 = \dot{\rho}_5 V_{RL} + \dot{m}_5 + \dot{m}_4 = 0 \tag{105}$$

Checking (105) for a solution equal to zero satisfies the mass balance condition.

The balance of energy through the system means equating the rate of change of energy equation to the energy conservation equation.

$$\dot{m}_{in}h_{in} - \dot{m}_{out}h_{out} + q_{leak} - V_{TL}\dot{\rho}_{TL}u_{TL} - V_{TL}\rho_{TL}\dot{u}_{TL} \cong 0 \qquad (106)$$

(106) shows the equation used to check for energy balance. The balance passes if the equation balances to an absolute error below $\pm 10^{-6}$.

3.5. Initialization of Program Variables

The initialization of the program requires defining the mass flow rates, pressures, temperatures, and operational values of all components and nodes in the system. The model map has been included again here for reference purposes.
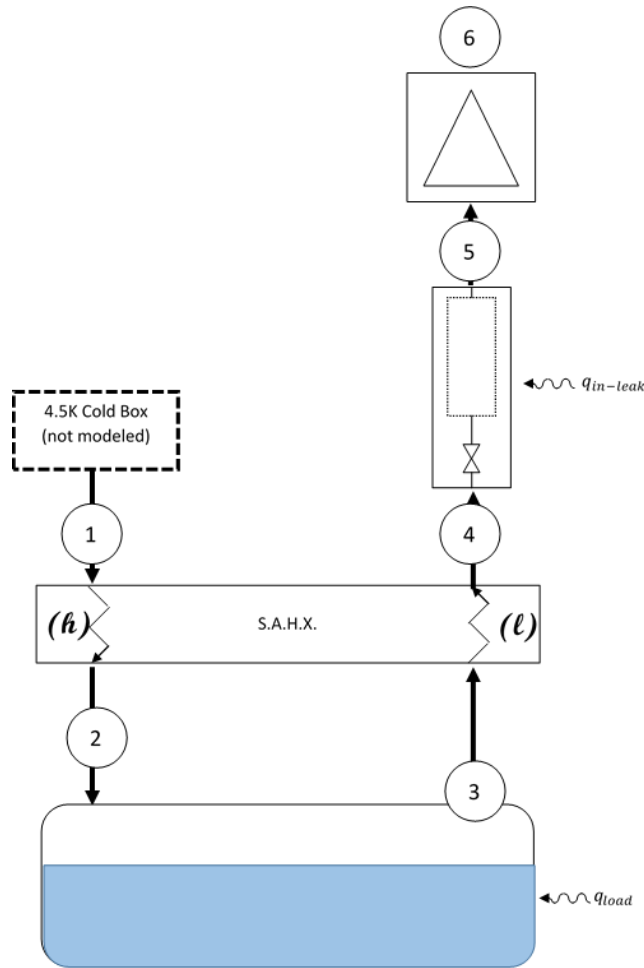
Figure 18. Model map for initialization reference

The initialization requires working from dewar outward to determine the conditions throughout

the system. The assumption for initialization is that there is no mass accumulation in the system.

That is, $\dot{p}(0) = 0$ and the cold compressor is spinning fast enough to remove the mass leaving

the dewar due to evaporation. This means that mass flow into the dewar is equal to mass flow out

of the dewar.

From there the sub-atmospheric heat exchanger can be solved. By solving this the fluid

conditions at nodes 2 and 4 are now known.

With those conditions known the return line volume can be solved. However, the return line volume is not analyzed in the same method used during the time step solutions. (93) is solved for the pressure drop across the transfer line, $\dot{m}_4$ is equal to $\dot{m}_5$ because there is no depressurization of the return line volume.

$$h_5 = h_4 + \frac{q_{in-leak}}{\dot{m}_4} \tag{107}$$

(108) shows how the specific enthalpy of the fluid at node 5 simplifies. With the pressure and specific enthalpy known from solving (93) and (107) the fluid conditions at node 5 are known.

Finally the cold compressor model is solved and the fluid conditions at node 6 are known. With all conditions through the system solved, serving the basis for the transient solution, the initialization is complete and the program transitions into the transient calculation stage.

3.6. Process Time Step Calculation

The purpose of the integrated system pump-down process model (process model) is to model the downstream components of the 2K cold box effectively and evaluate the transient nature of the cryo-module (load) pressure and the transient polytropic efficiency of the cold-compressor as a function of different vapor removal mass flow rate profiles (pump-down paths).

This document serves to give detail on the initialization of the coupled model sub-routines and outline the operation path of the process level program. Furthermore, it details the validation methods used to check the solution of each sub-routine in the integrated program.

The driving factor for the overall process is a dictated *pump-down process path.* This process path is in fact the desired mass flow rate through the cold compressor as a function of load pressure. The mass flow through the cold-compressor creates a net loss of mass in the cryo-

module and thus a time rate of change of pressure, which approaches the desired pressure: on the

order of 0.030 [bar]. This sub-atmospheric helium condition in the load allows for the

production of helium below 4.5 [K].

3.6.1. Flow chart for program operation

The chronology of subroutine execution is stated in Figure 19.
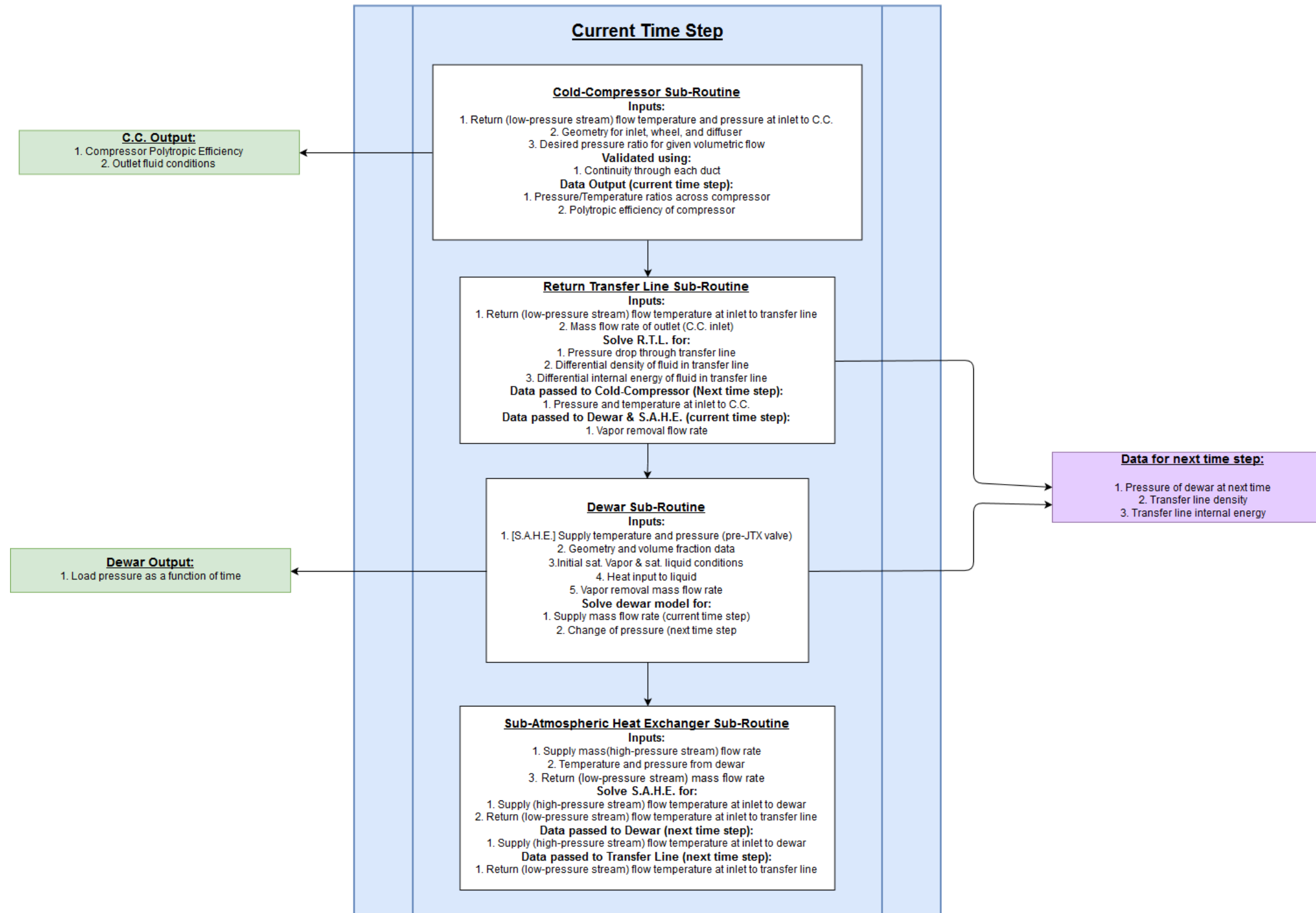
Figure 19. Overall System Model Flow Chart

3.6.2. Cold Compressor Subroutine

The first subroutine is the cold compressor model. This subroutine reads in the temperature and pressure from node 5 and begins determining the operational parameters of the machine. First the pressure ratio must be set. This is prescribed for the cold compressor as a function of volumetric flow rate.
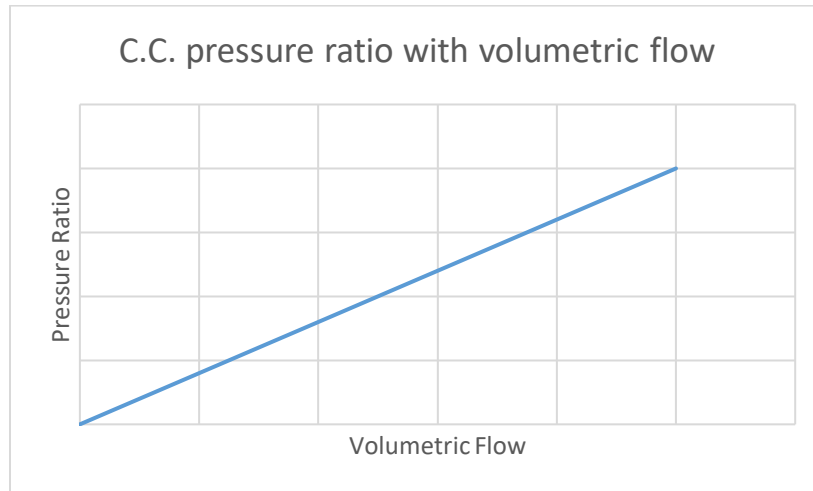
Figure 20. Pressure ratio prescribed as a function of volumetric flow for the cold compressor

Figure 20 shows how the pressure ratio will be set during the pump-down. As the density of incoming flow changes the prescribed pressure ratio will adjust to accommodate the new flow conditions.

With pressure ratio prescribed the next step is to solve the cold compressor and determine the wheel speed needed to support the desired pressure ratio. An approximate wheel speed is solved for using an isentropic radial bladed relationship, derived from equating the work done by the impeller wheel to the work done compressing the fluid.

$$\text{Pr} = \left(\frac{P_4}{P_1}\right) = \left[1 + \left(\frac{\gamma-1}{\gamma}\right)\left(\frac{1}{P_1 v_1}\right)(\omega r_T)^2\right]^{\frac{\gamma}{\gamma-1}} \tag{108}$$

75

By solving (108) for an approximate ω value, the first solution from the cold compressor subroutine will be below the desired pressure ratio. From there the program will increase the angular velocity of the wheel and resolve until the desired pressure ratio is achieved. Once the desired solution is obtained, the polytropic efficiency is solved and the subroutine terminated.

3.6.3. Return Transfer Line Subroutine

The second subroutine to be solved is the Return Transfer Line model. This model is used to solve for the mass flow rate out of the dewar and the inlet temperature and pressure to the cold-compressor for the next time step.

The return line volume begins with the initial conditions or the previous time step pressure and temperature. With $\dot{m}_{5 \to 6}$, $p(4, t-1)$ and $p(5, t-1)$ known $\dot{m}_{4 \to 5}(t)$, $p(5, t+1)$, and $T(5, t+1)$ can be solved. After those values are determined, the subroutine terminates.

3.6.4. Dewar Subroutine

The third subroutine to be solved is the saturated dewar depressurization model. This model makes use of a constant dewar volume constraint equation, the mass balance equation, and energy balance equation to calculate $\dot{m}_{2 \to 3}$ and $\dot{p}(t)$. After these values are calculated, the rate of change of pressure is integrated and $p(3, t+1)$ and $T(3, t+1)$ are calculated and stored for the next time step. The mass flow rate of the supply fluid (at the current time step) and the current time step dewar saturated vapor fluid conditions are then passed to the sub-atmospheric heat exchanger model. After the data for the next time step is calculated the subroutine terminates.

3.6.5. Sub-Atmospheric Heat Exchanger Subroutine

The sub-atmospheric heat exchanger is modeled solving the heat exchanger division rate equation and the heat exchanger division energy balance equation. This model accepts the supply mass flow rate and the low-pressure inlet temperature $T_{l,N}$ from the dewar model, which is the saturation temperature of the vapor. The high-pressure inlet temperature $T_{h,0}$ and pressure p_h is a prescribed value from the 4.5K cold box. The output of this model is the low-pressure stream outlet temperature $T_{l,0}$ and the high-pressure stream outlet temperature $T_{h,N}$. These data points are for the next time step. $T_{l,0}$ is passed to the Return Transfer Line Subroutine to solve for pressure and temperature at the inlet to the cold compressor model. After the *(h)* and *(ℓ)* stream outlet temperatures are solved the subroutine terminates.

3.6.6. March with time

After the load pressure, density and internal energy in the transfer line for the next time step are stored, the time step increments and the calculations begin again from the beginning. This continues until the system has reached the target load pressure. At that point the load pressure will be graphed against time, along with polytropic efficiency graphed against process path mass flow and time.

4. Results

The integrated model was analyzed for two mass flow profiles. The first was a constant $250 \frac{g}{s}$ mass flow rate across the cold compressor. The second was a load dependent mass flow profile with a maximum mass flow rate of $150 \frac{g}{s}$ across the cold compressor. The different mass flow profiles were used to isolate the effects of changing mass flow rates from changing time step and conditions in the system.

4.1. Constant Cold Compressor Mass Flow Solution

This was used primarily to study the overall pump-down process and gain an understanding of the system while limiting complexity of component response due to mass flow rate variation. Neglecting the ramp-up portion of the pump-down, the mass flow rate across the cold compressor was held at $250 \frac{g}{s}$.

4.1.1. Cold Compressor Constant Mass Flow Response
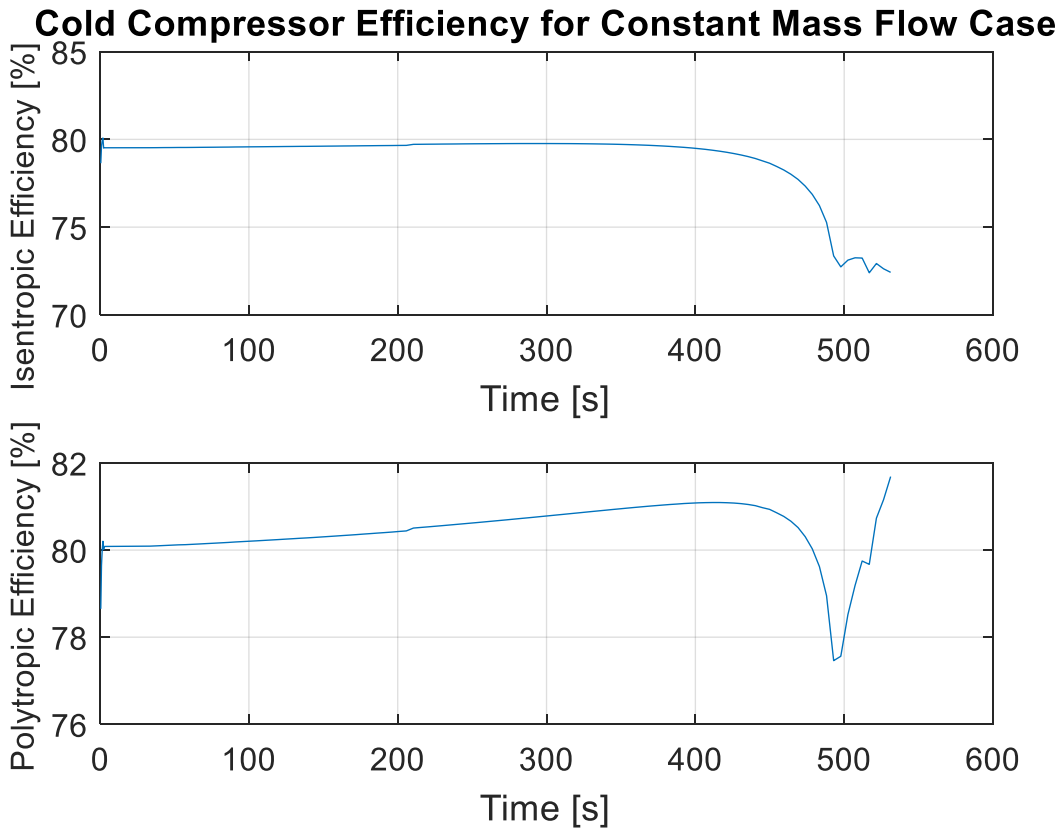
Figure 21. Cold compressor isentropic polytropic efficiencies over pump-down duration

Figure 21 shows the isentropic and polytropic efficiencies over the duration of the pump-down for the constant mass flow case. The isentropic efficiency is nearly constant over the duration. However, in the last 20% of the pump-down, the rapid increase in speed to keep up with the changing volumetric flow rate lead to a drop in the isentropic efficiency. The polytropic efficiency reacted similarly. When the pressure ratio reached the necessary level ($Pr > 5$) for the mass flow rate at the given input conditions, the polytropic efficiency increased to operational levels around 81-82%, where the isentropic efficiency remained around 73%.
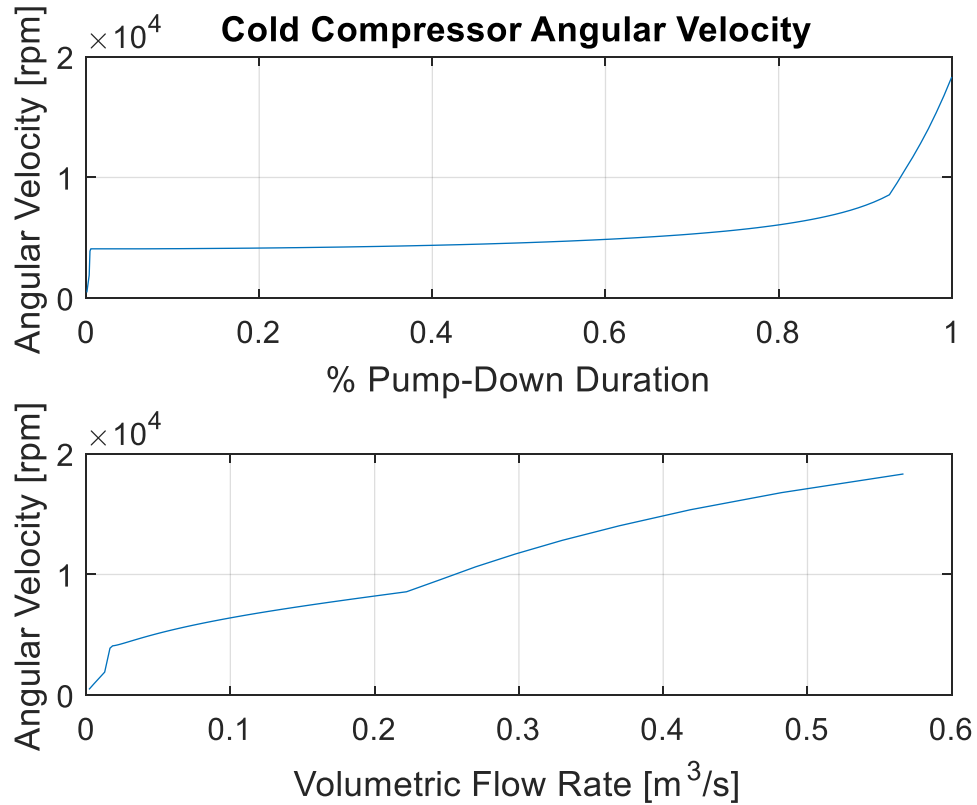
Figure 22. Cold compressor impeller angular velocity as a function of volumetric flow rate through the compressor

The increased mass flow rate for the constant mass flow case results in the need for higher impeller angular velocities to produce stable operation. As a result, the output pressure ratios as the low pressures and temperatures are significantly higher than expected for the design case. Specifically, the design case expected a pressure ratio of 3 at a mass flow rate of $150 \frac{g}{s}$.
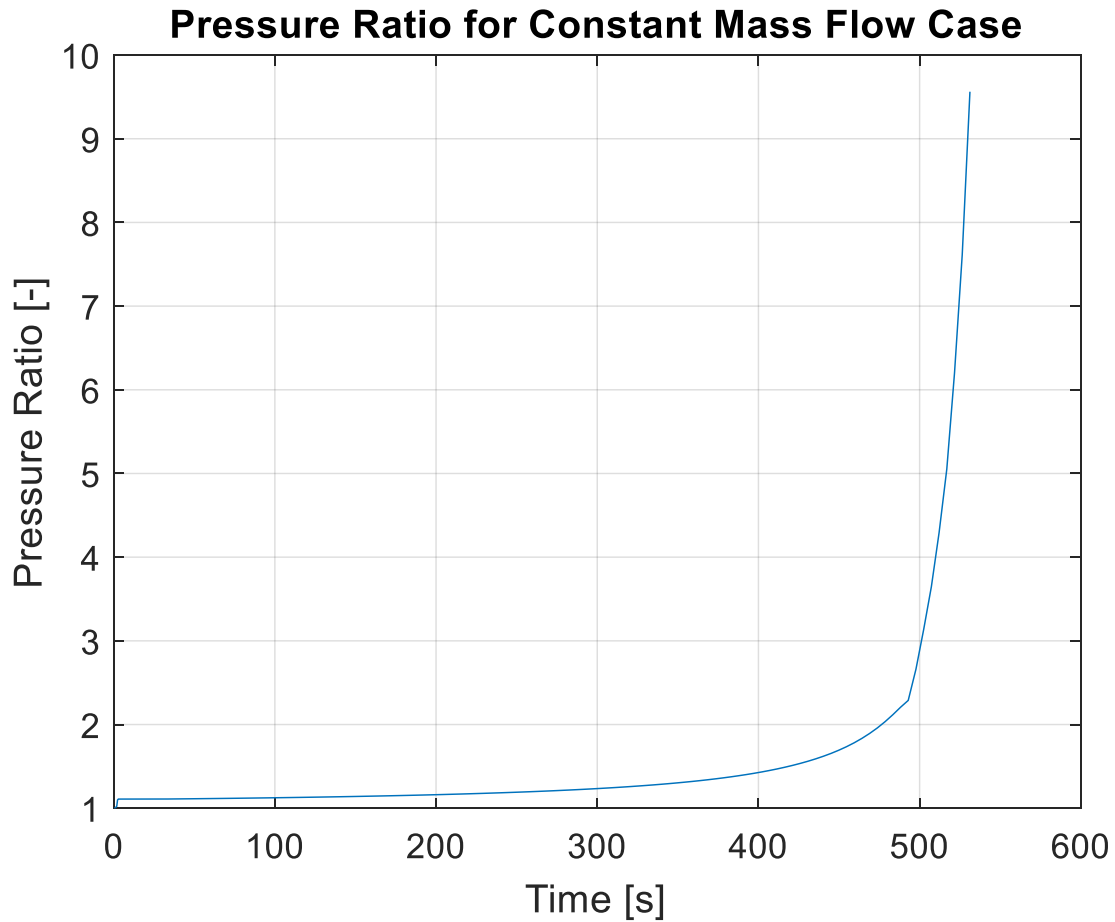
Figure 23. Cold compressor pressure ratio over constant mass flow pump-down

An important distinction for the constant mass flow rate pump down case is that it deviates

significantly from the design case. The impeller was sized for $150 \frac{g}{s}$ mass flow rate at the

operating condition (0.029 bar and roughly 3.8 K). The pressure ratios necessary to produce

stable compressor operation at the $250 \frac{g}{s}$ mass flow rate are significantly higher than expected for

the design case.
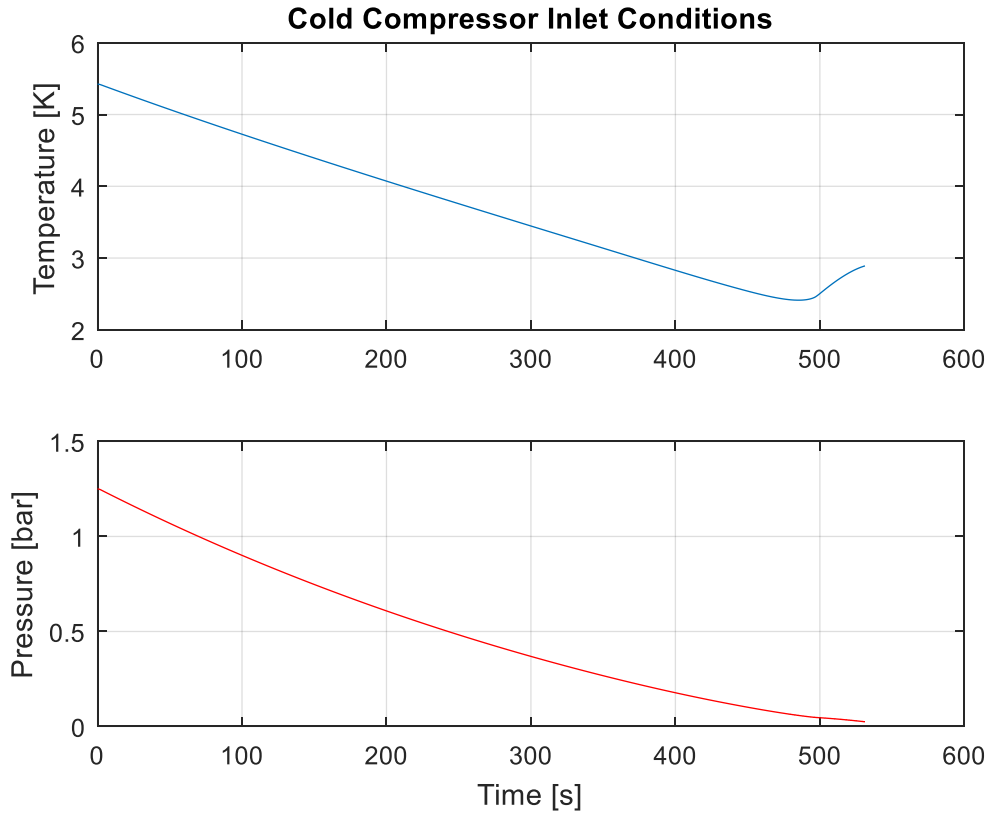
Figure 24. Cold compressor inlet pressure and temperature during constant mass flow pump-down

The cold compressor inlet pressure and temperature graphs are reasonable through the pump-down. The only area of interest would be the last 10% of the cold compressor inlet temperature. The inlet temperature increased by roughly 0.5 K in the last portion of the pump-down. This is likely due to the lambda point transition in the cryo-module. The change in the saturated vapor (not a superfluid) specific heat at this temperature and pressure is such that the helium temperature returning from the cryo-module, past the sub-atmospheric heat exchanger is elevated.
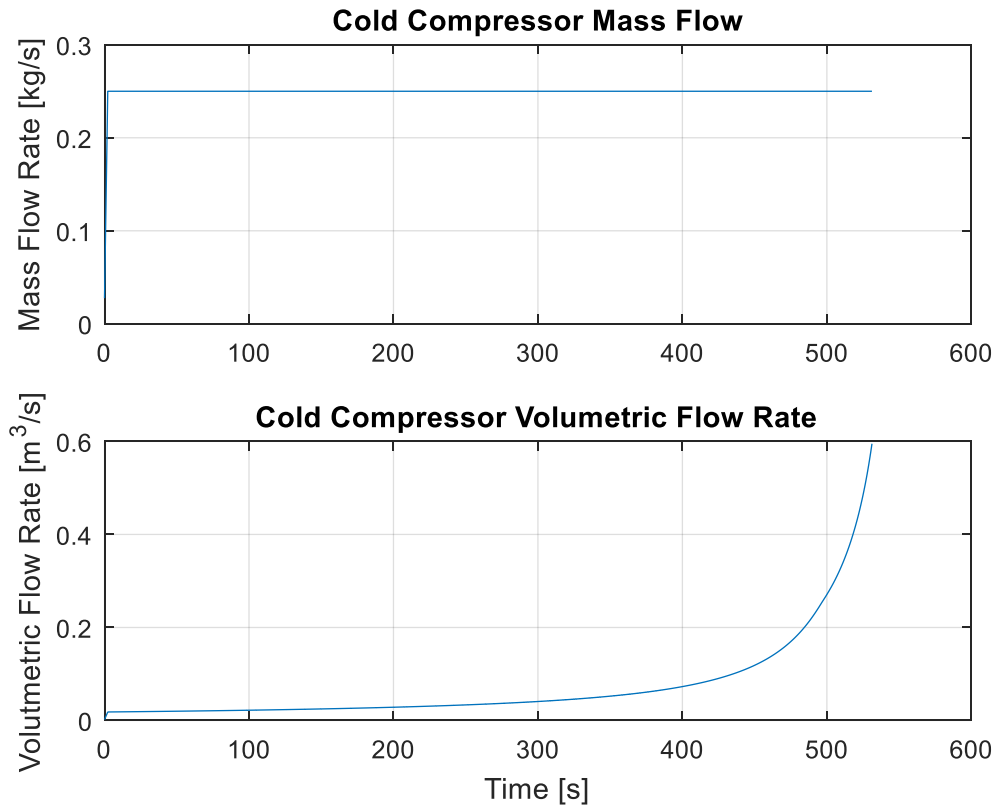
Figure 25. Mass and volumetric flow rates during constant mass flow pump down

As the system depressurizes there is an increase in volumetric flow rate through the cold compressor.

4.1.2. Return Line Volume Constant Mass Flow Response

The mass flow into the return line volume is not constant relative to the mass flow rate across the cold compressor (see Figure 26).
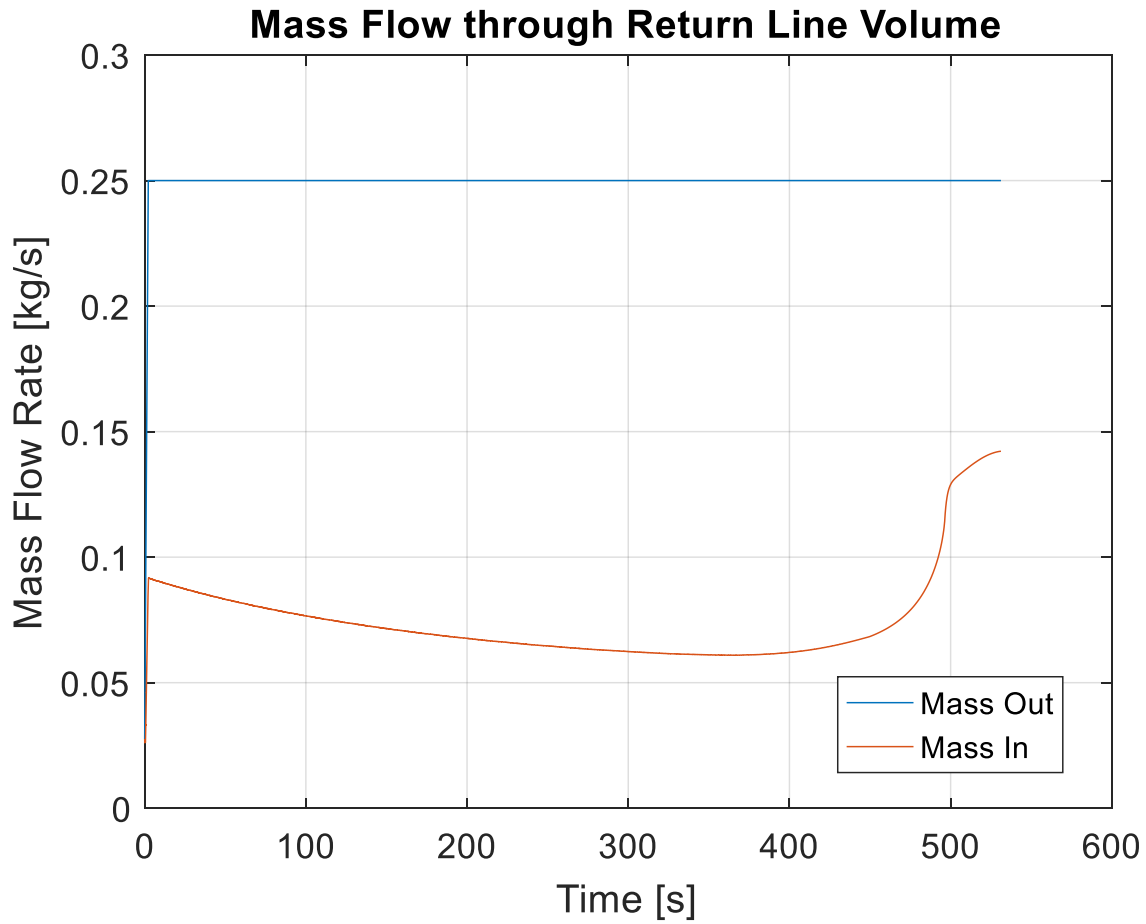
Figure 26. Mass flow rate into and out of the return line volume

As the conditions, on the warm end of the *(ℓ)* stream from the sub-atmospheric heat exchanger change, the flow rates change in kind. Again, there is an increase in mass flow rate into the return line volume toward the end of the pump-down.
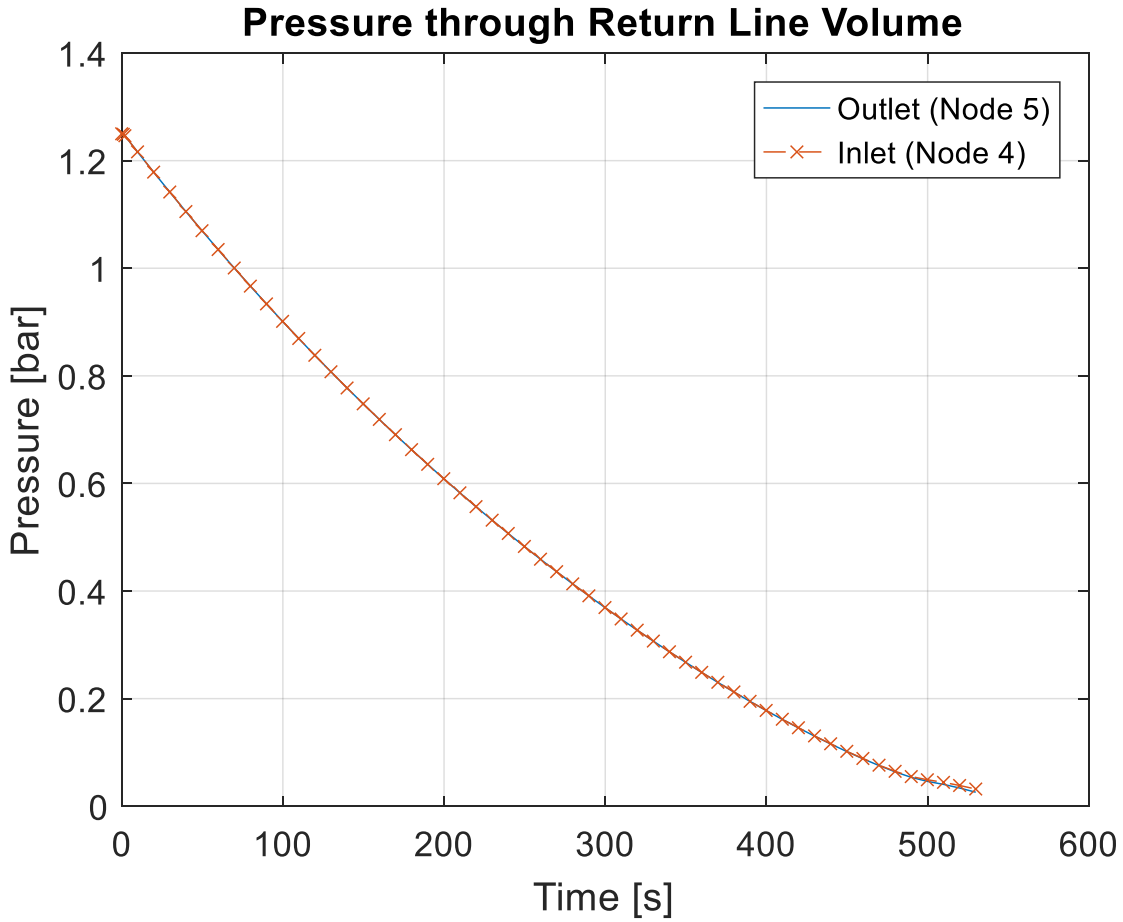
Figure 27. Return line volume and cryo-module pressures plotted over constant mass flow pump-down

Figure 27 shows how the return line volume (node 5) pressure compares to the cryo-module pressure during the pump-down. Recall that the cryo-module and return line volume inlet pressures are equal (node 3 pressure equal to node 4 pressure). There is a divergence in the pressures toward the end of the pump-down. This is illustrated in Figure 28.
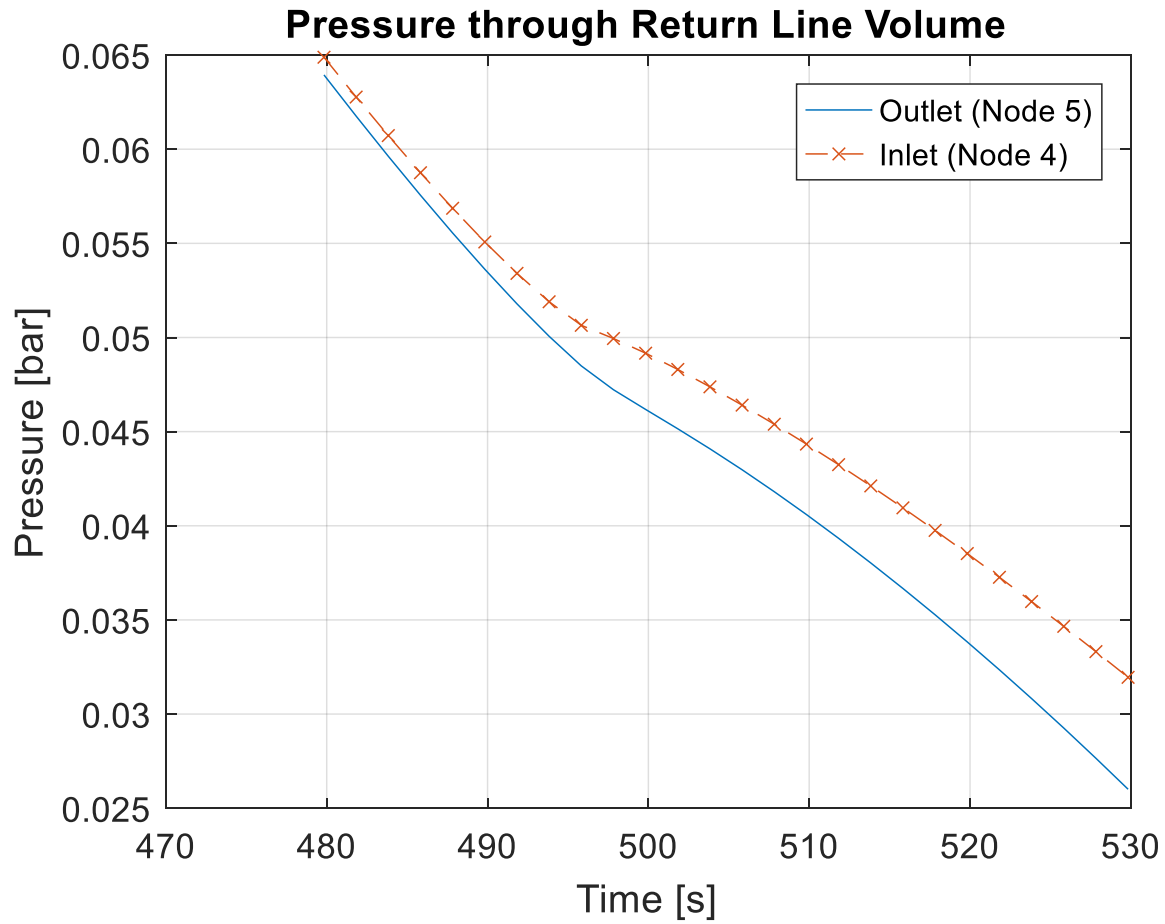
Figure 28. Return line volume and cryo-module pressures focused on final portion of pump-down

Figure 28 shows how the pressure differential between node 4 and node 5 increases toward the end of the pump-down. This increased pressure differential would drive significantly more mass through the return line volume as seen in Figure 26.
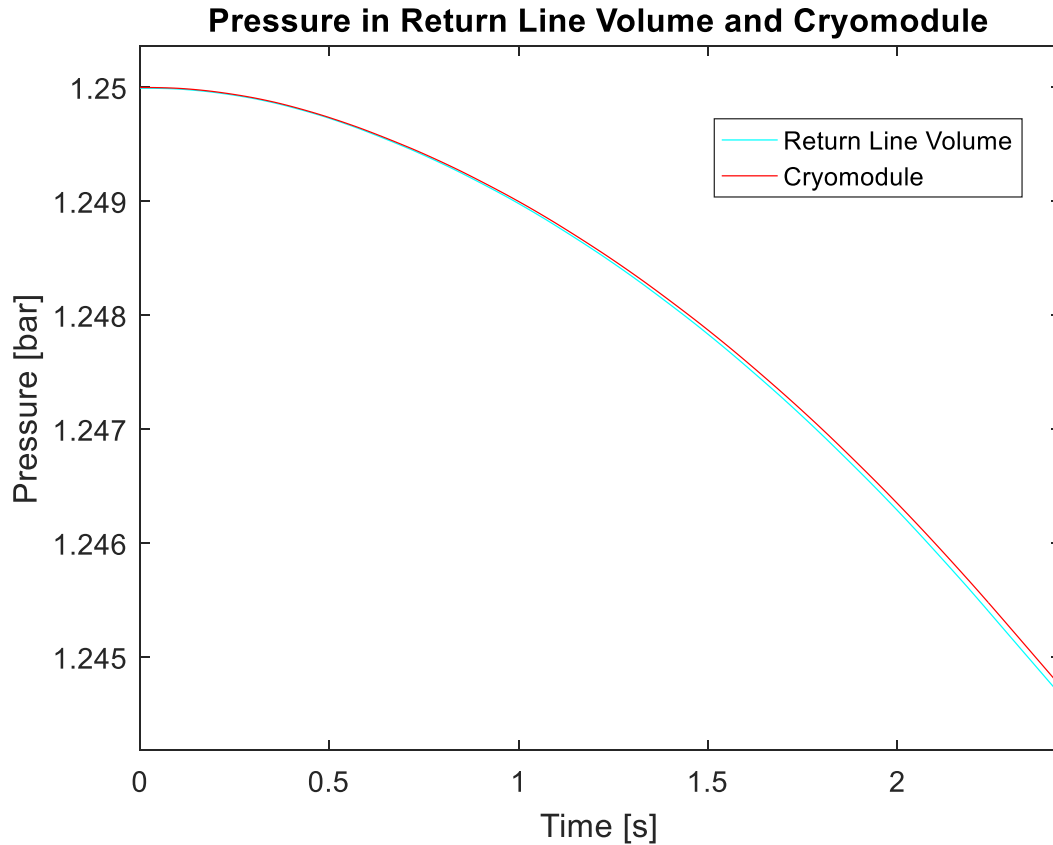
Figure 29. Return line volume and cryo-module pressure focusing on the initial ramp-up of the pump-down process

Figure 29 shows how the pressure response curves develop in the beginning seconds of the solution. As the pressures drop, mass removal from the components is implied, as such there is a pressure differential that is developing between nodes 4 and 5.
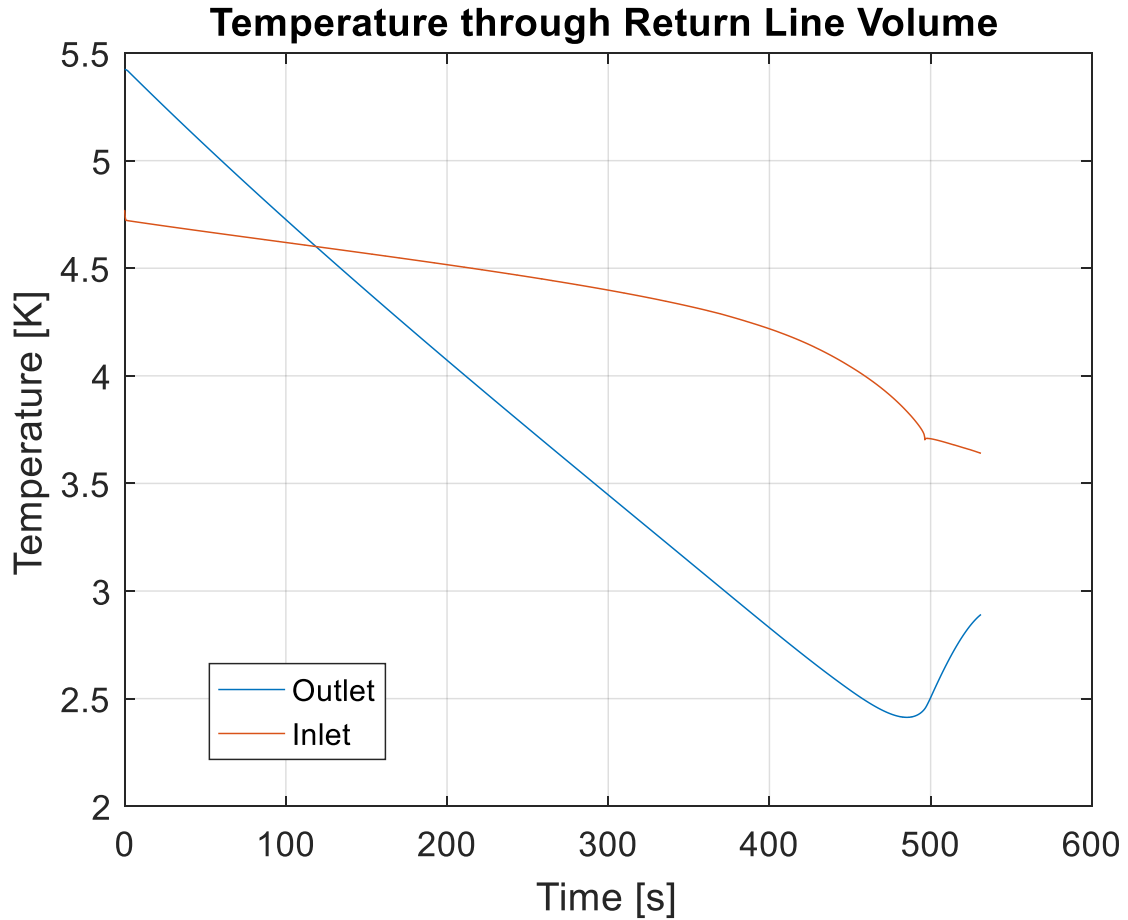
Figure 30. Return line volume inlet and outlet temperatures during pump-down

Figure 30 shows node 4 (inlet) and node 5 (outlet) temperatures during the pump-down. The

node 4 temperature is driven by the refrigeration recovery (heat input) from the 4.5 to 2 K heat

exchanger. However, it is interesting to note that the node 5 temperature drops below that of the

input. This is a non-physical response and will be discussed further in Appendix A.

4.1.3. Cryo-module Constant Mass Flow Response

The cryo-module depressurizes nearly linearly with respect to time, as seen in Figure 31.
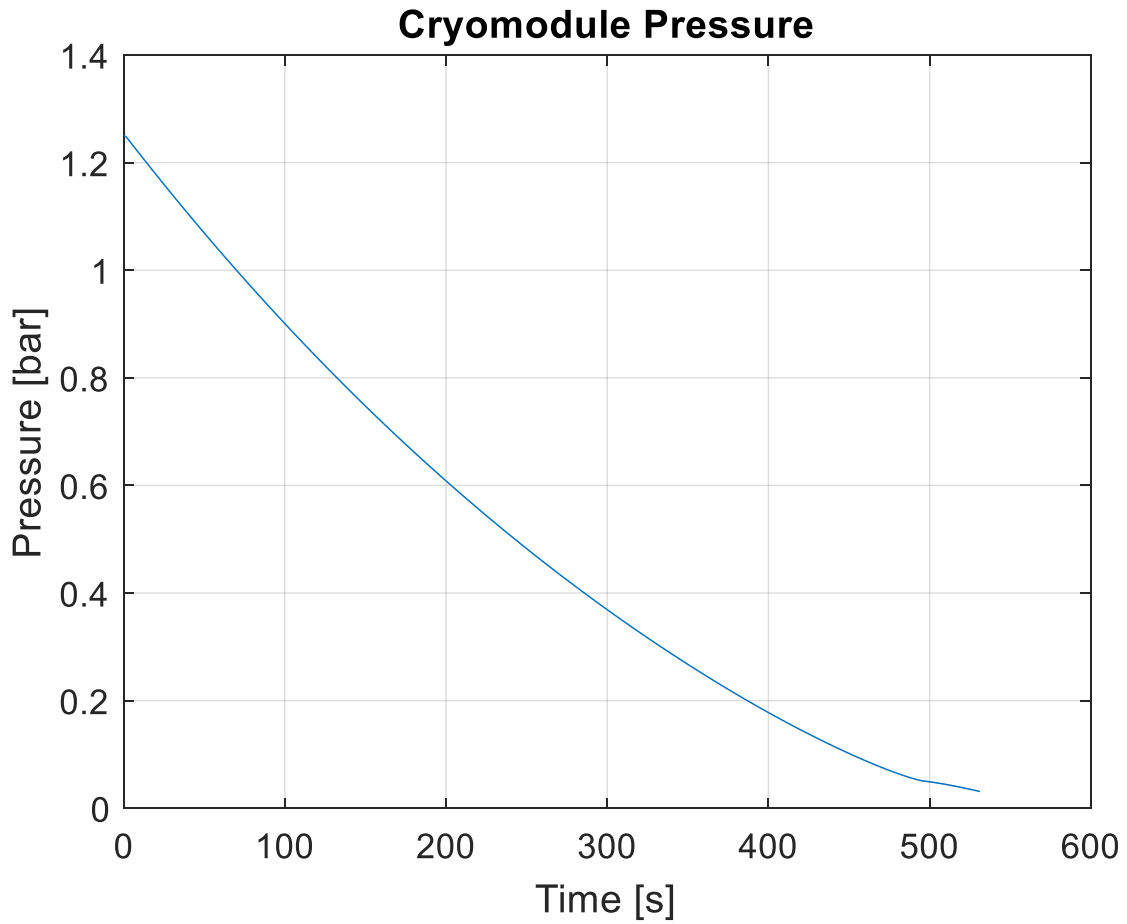
Figure 31. Cryo-module pressure during constant mass flow pump-down

The constant mass flow rate across the cold compressor leads to a consistent mass flow removal

from the cryo-module. There is a step change at the bottom of the curve that would correspond to

the lambda point transition.
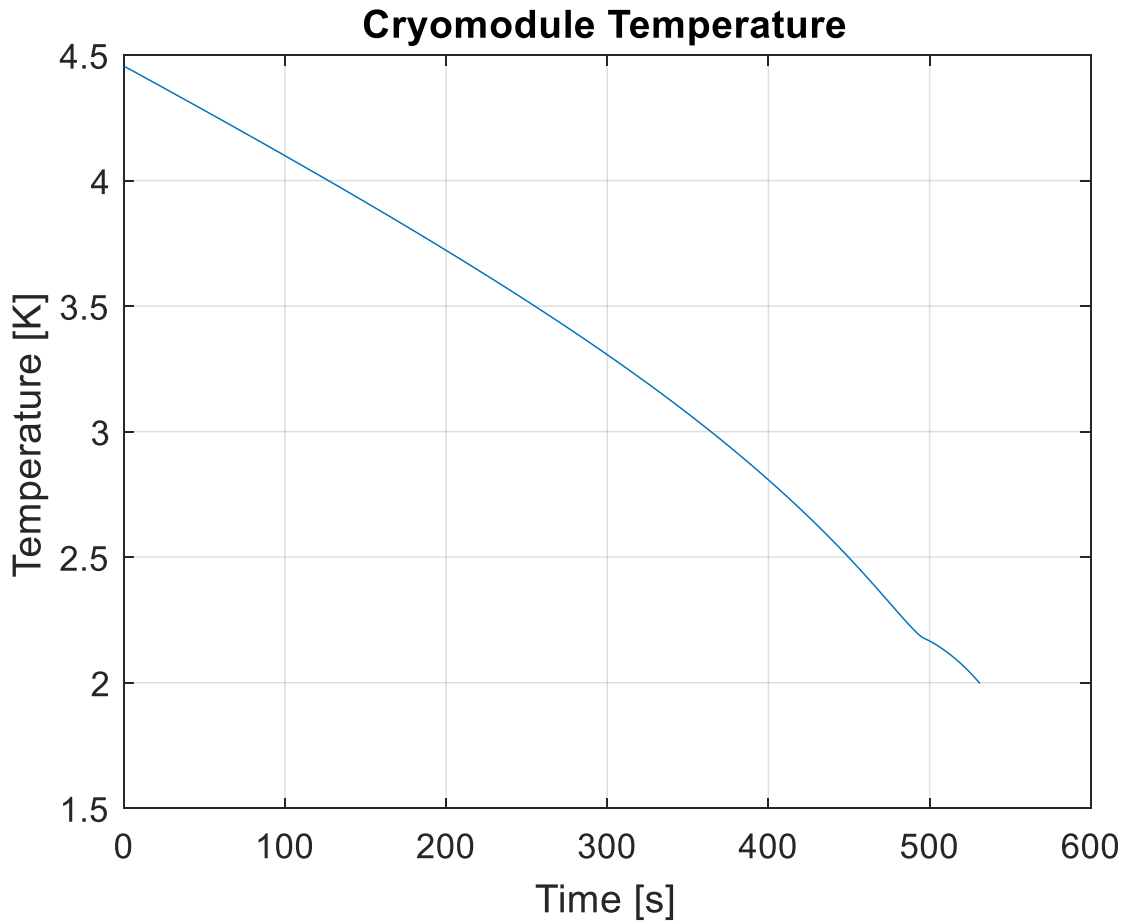
Figure 32. Cryo-module temperature over pump-down duration

Figure 32 shows how the cryo-module temperature varies over the pump-down. As expected, the

temperature in the cryo-module varies with the pressure, as the temperature is simply the

saturation temperature at the given cryo-module pressure. Again there is a slight change in the

curvature at the end of the pump-down. This corresponds to a lambda-point transition.
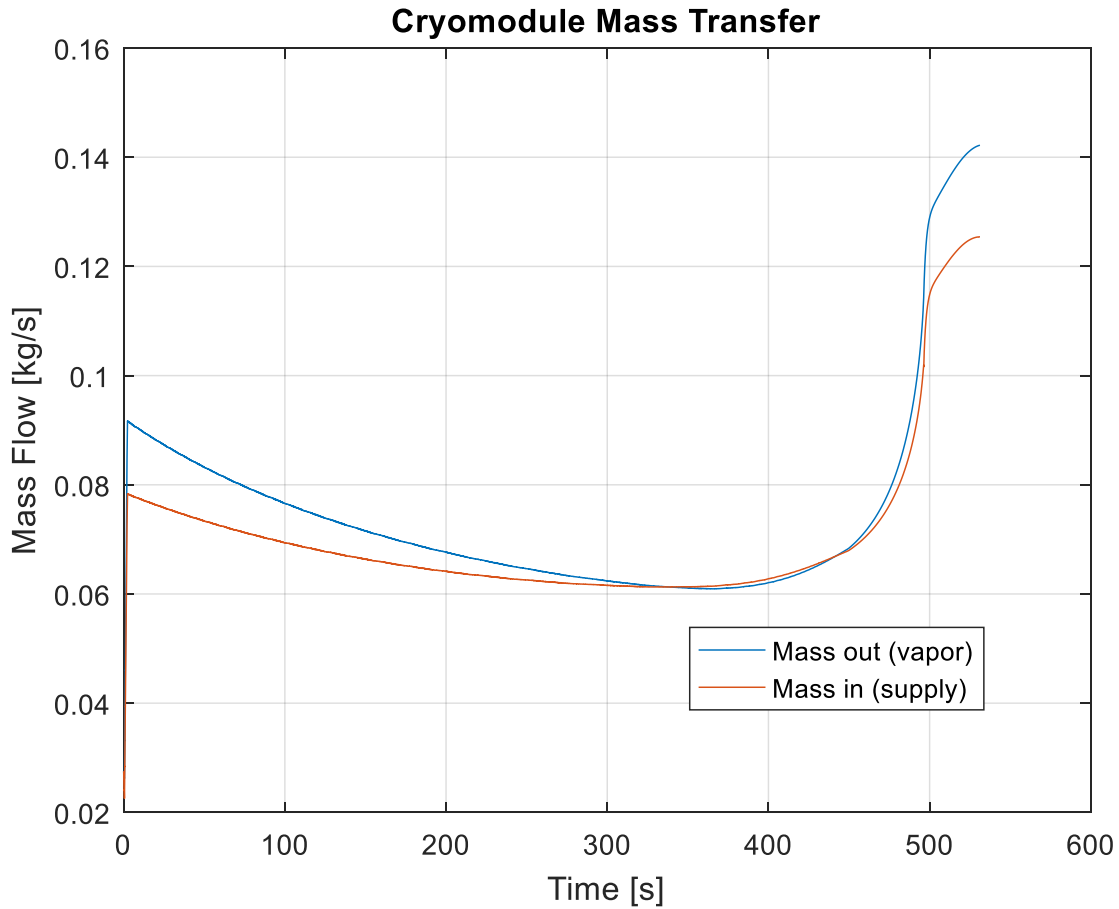
Figure 33. Supply (mass in) and vapor removal (mass out) mass flow rates during pump down

At roughly 350 s to 450 s the cryo-module is depressurizing while there is mass accumulating.

Specifically, the cryo-module is accumulating mass for 104.12 s. It is doing so with a maximum

mass accumulation of $0.72 \frac{g}{s}$. This translates to the mass flow into the system being at most

1.16% more than the mass out of the cryo-module.
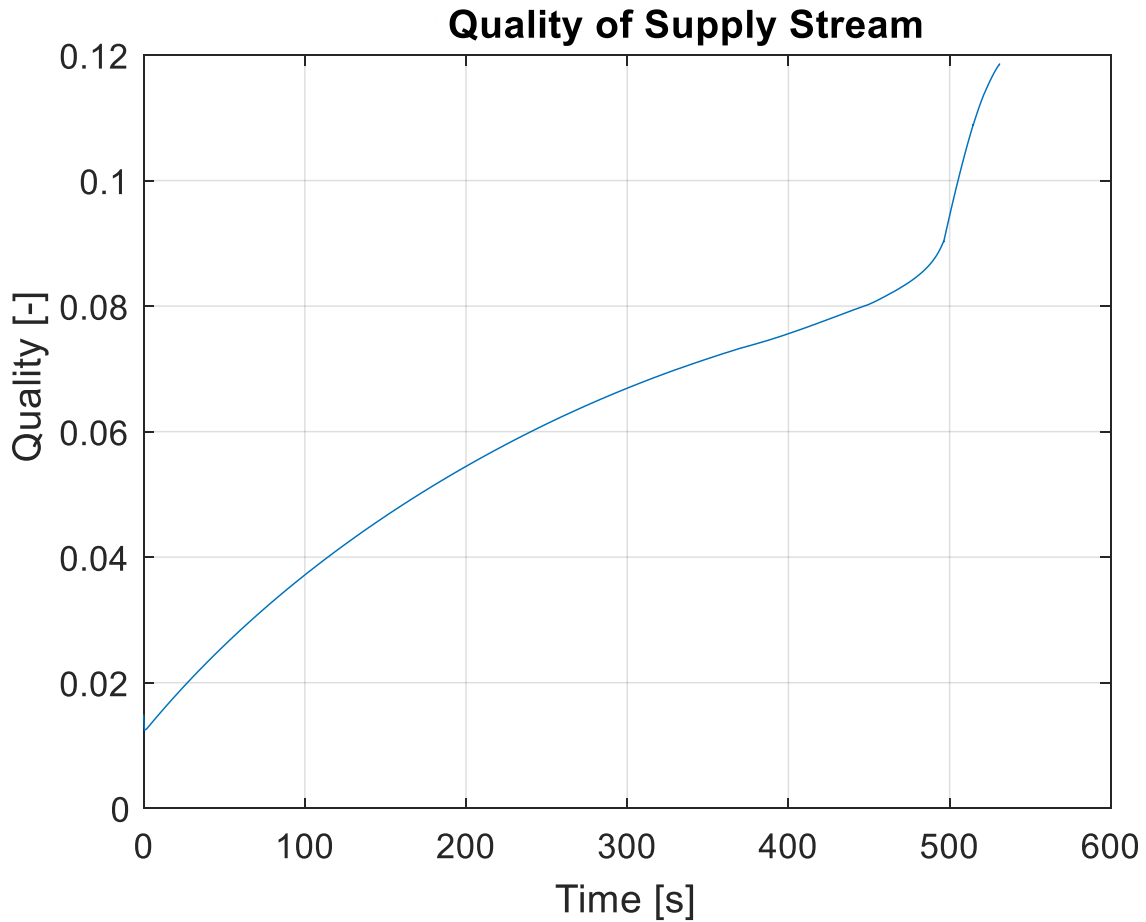
Figure 34. Cryo-module supply flow quality over pump-down duration

Figure 34 shows the quality of the flow incoming to the cryo-module. This is interesting as the system approaches depressurization, there is a significantly higher amount of vapor flashing off the supply stream. This vapor mass flashing off could be what is forcing the mass flow rate to jump at the later times in the pump-down.

4.1.4. Sub-Atmospheric Heat Exchanger Constant Mass Flow Response

Refer to Figure 33 for *(ℓ)* and *(ℏ)* stream mass flow rates. Figure 35 shows node 1 and node 2 temperatures. Node 1 is shown simply to give reference for node 2 that is recovering

refrigeration from the low-pressure stream. As the cryo-module is pump-down and cooled, it

shows how the high-pressure supply stream temperature reacts as a function of the recovery.
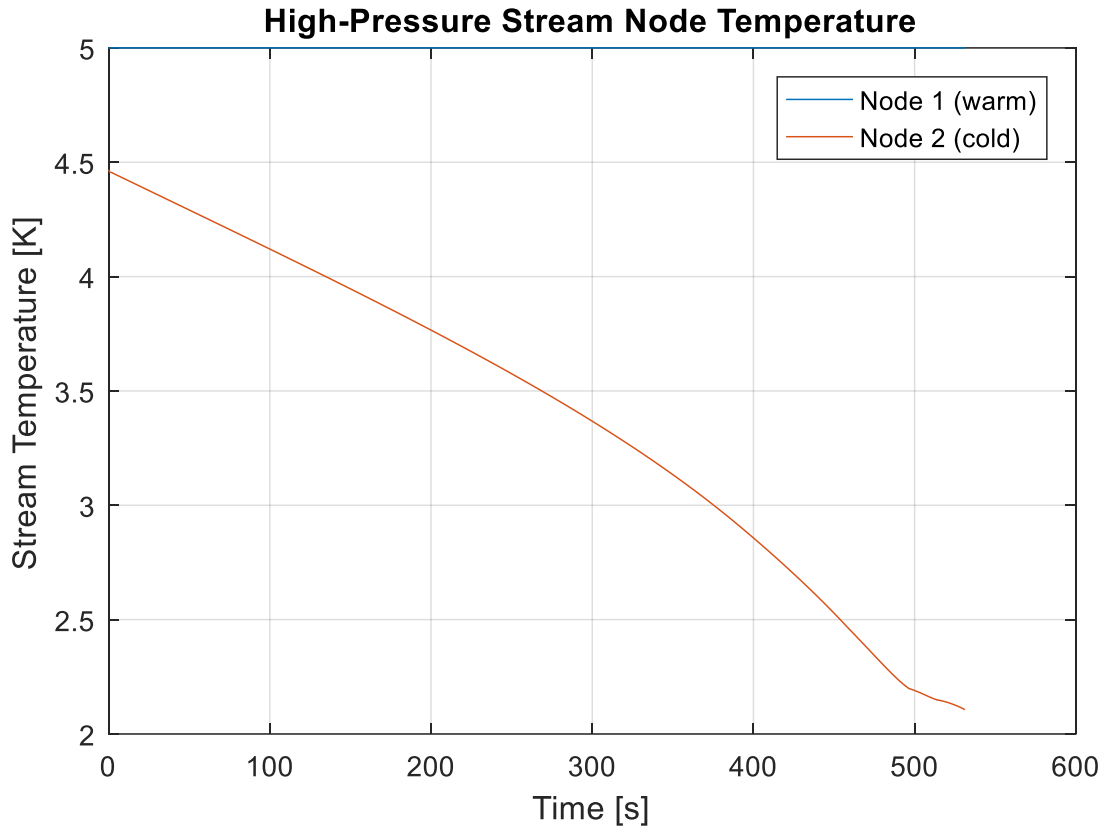
Figure 35. Sub-atmospheric heat exchanger high-pressure warm and cold end stream
temperatures

Conversely, Figure 36, shows how the low-pressure stream reacts to the refrigeration recovery.

Node 3 showing the cryo-module temperature and node 4 showing the temperature of the outlet

of the sub-atmospheric heat exchanger.
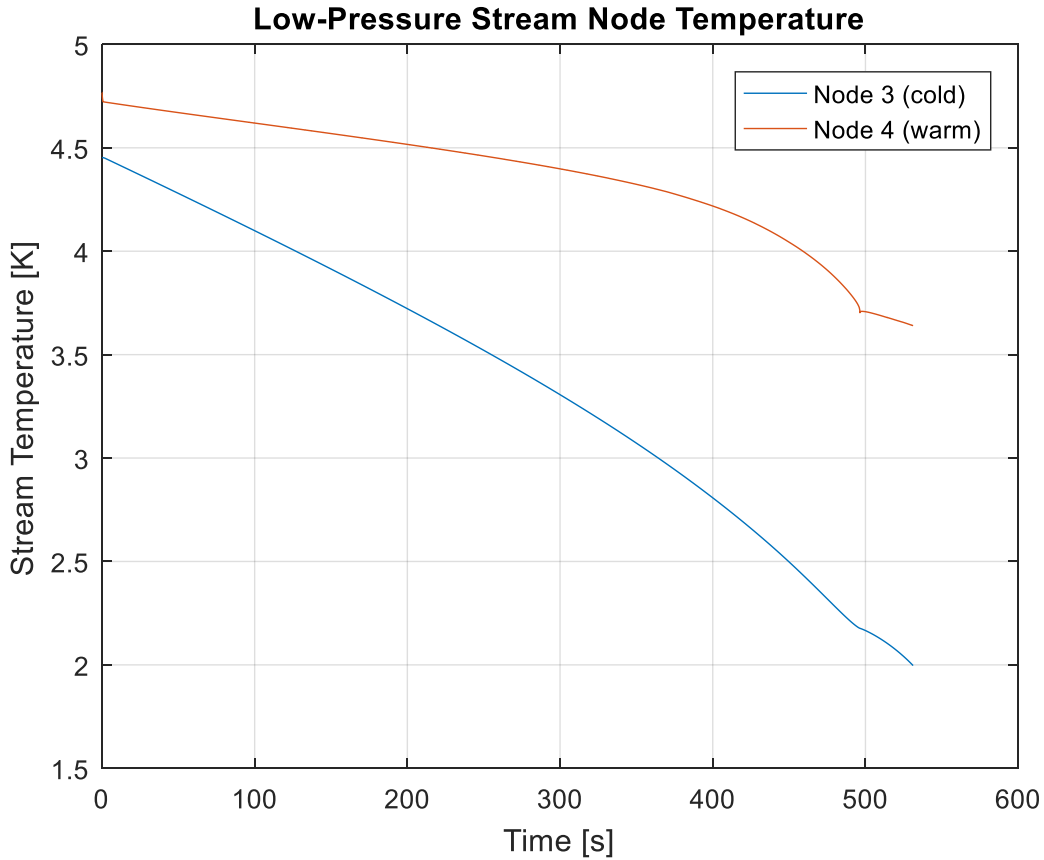
Figure 36. Sub-atmospheric heat exchanger low-pressure warm and cold end stream temperatures

4.2. Load Pressure Dependent Cold Compressor Mass Flow Solution

The load pressure dependent path was generated using the constant cold compressor process path as inspiration. This was done simply to explore the results using non-constant paths. As the cryo-module pressure was pumping down consistently, it made sense to reference. The polynomial was generated to limit the number of piecewise functions and discontinuities in the mass flow rates of the system solution. Similarly to the constant mass flow case, there is a ramp-up section and a near constant (between 130 and 150 $\frac{g}{s}$) mass flow rates during the pump-down. The major difference in this case was limiting the rate of change of mass flow across the cold compressor

during the ramp-up process. The process path was defined by a fourth order polynomial, seen in equation (109).

$$\dot{m}_{cc} = -0.5341500 \times p_3^4 + 1.0631190 \times p_3^3 - 0.6350860 \times p_3^2 \\ + 0.0872060 \times p_3 + 0.1477530 \tag{109}$$

This produces a process path curve seen in Figure 37.
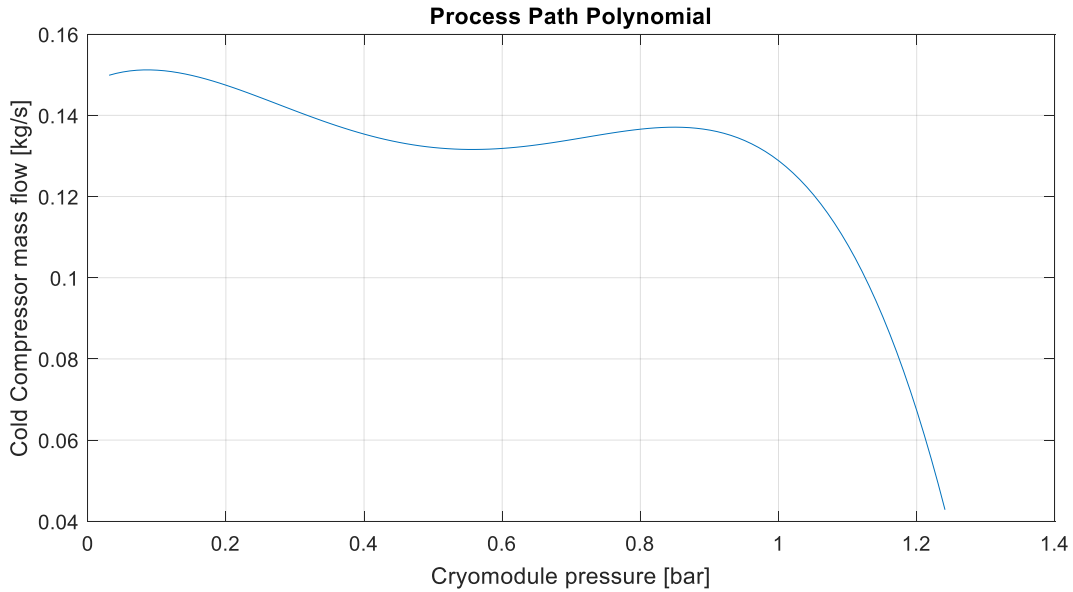
Figure 37. Process path polynomial plotted against load pressure

4.2.1. Cold Compressor Load Dependent Response

The isentropically solved angular velocity for the impeller was at times insufficient to produce valid flow in the subroutine. As a result for those areas the impeller speed was set manually.
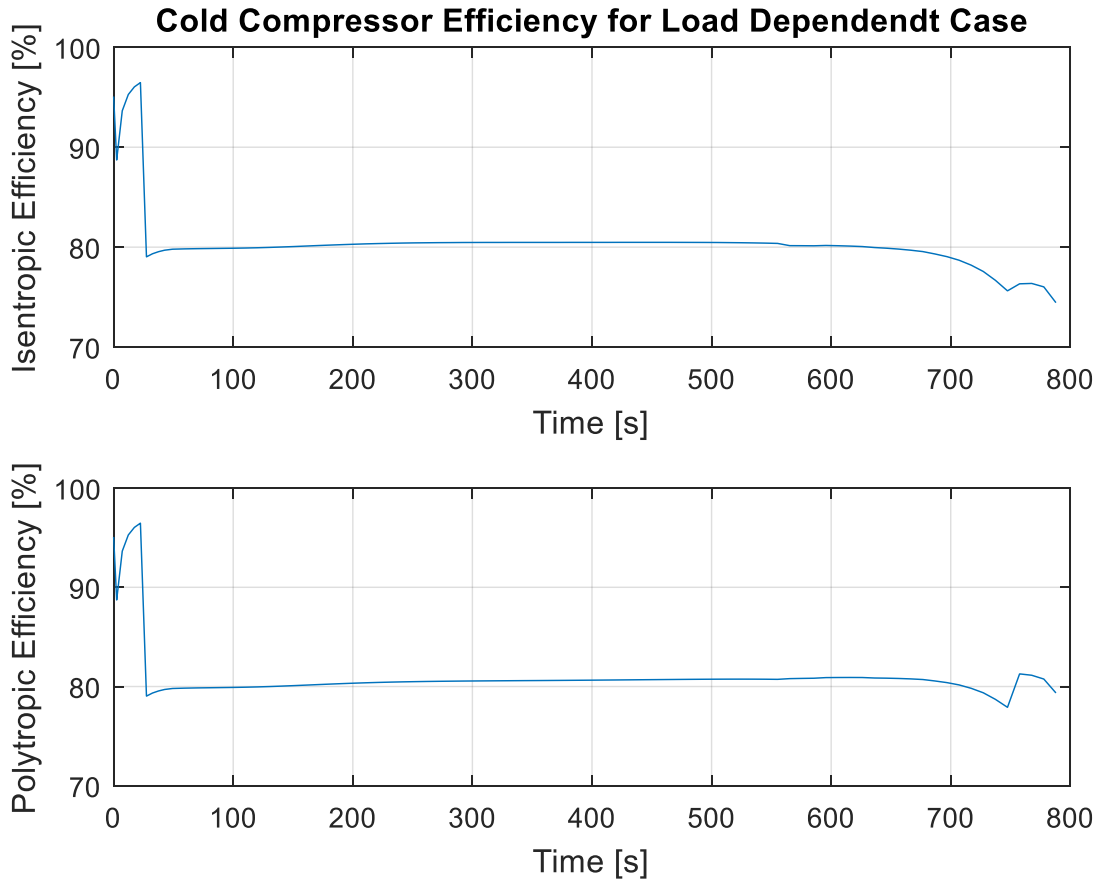
Figure 38. Cold Compressor Efficiency Metrics for load dependent case

Figure 38 shows the cold compressor isentropic and polytropic efficiencies. The load dependent case is largely similar to the higher mass flow constant case shown previously. However, with lower pressure ratios and corresponding angular velocities the efficiencies are lower.

This manual setting of the impeller speed produced some sporadic speeds and efficiencies seen in Figure 38 and Figure 40.
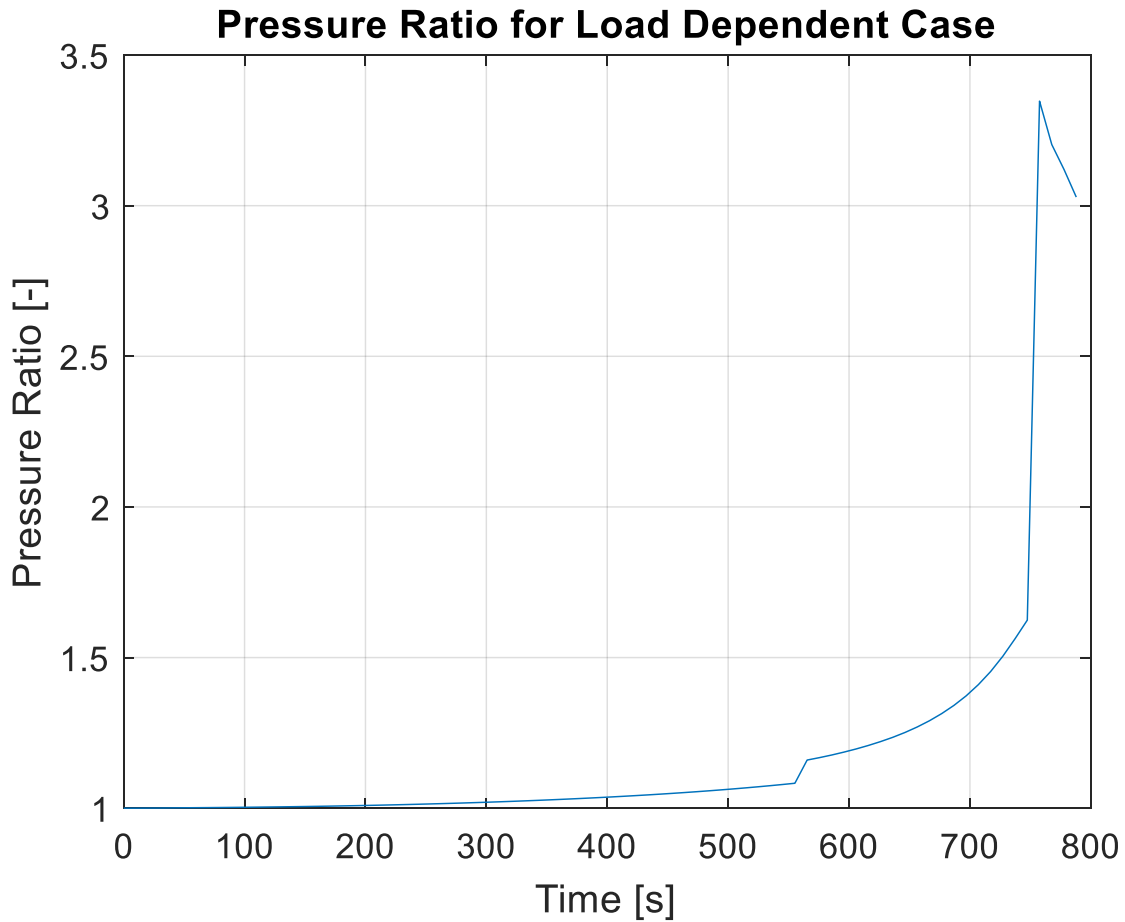
Figure 39. Cold compressor pressure ratio for load dependent case

A major contributor to the low speed issues with the compressor subroutine had to do with how

the pressure ratios were set. The pressure ratio was a linear function of volumetric flow rate see

equation

$$Pr = 5.845247 * \dot{V} + 1.01 \qquad (110)$$

This equation produced low pressure ratios at the low volumetric flow rates and as such, the

speeds needed to be adjusted.
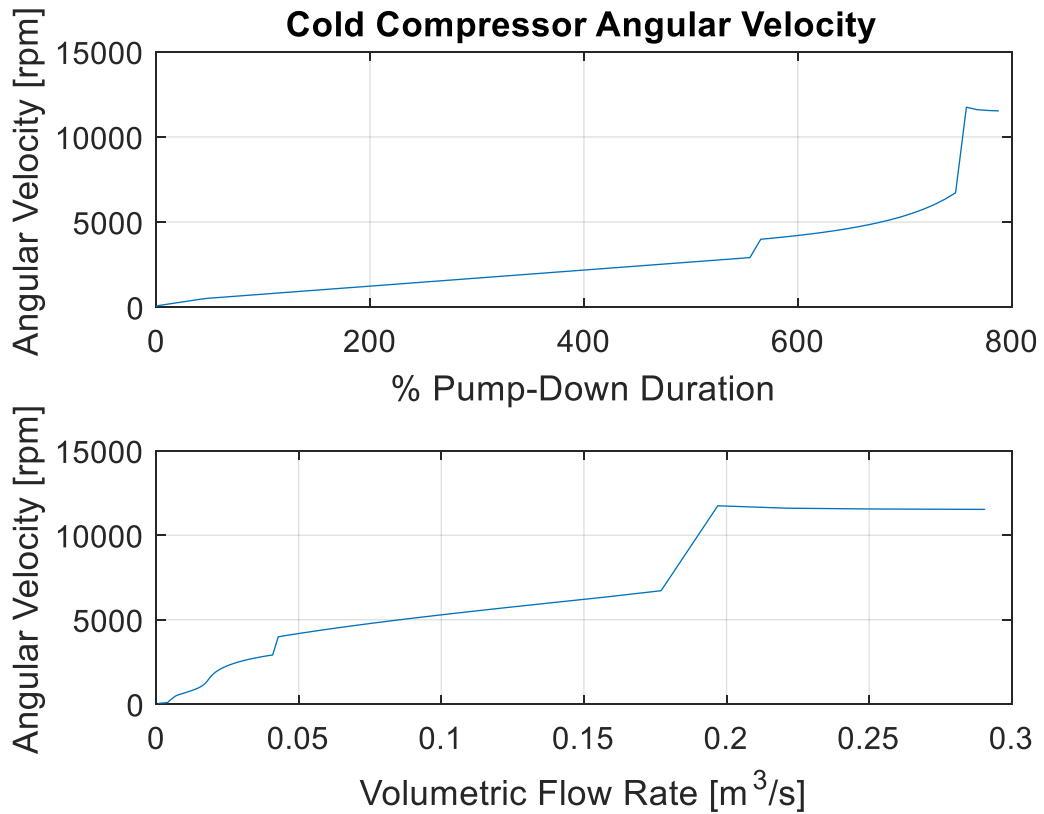
Figure 40. Cold Compressor frequency for load dependent case

Figure 40 shows how the impeller frequency varied with time. There are step changes in the solution as the settings of the impeller speed were modified in order to achieve the design conditions of the system. As the mass flow rate increased the impeller speed needed to increase sharply.

4.2.2. Return Line Volume Load Dependent Response
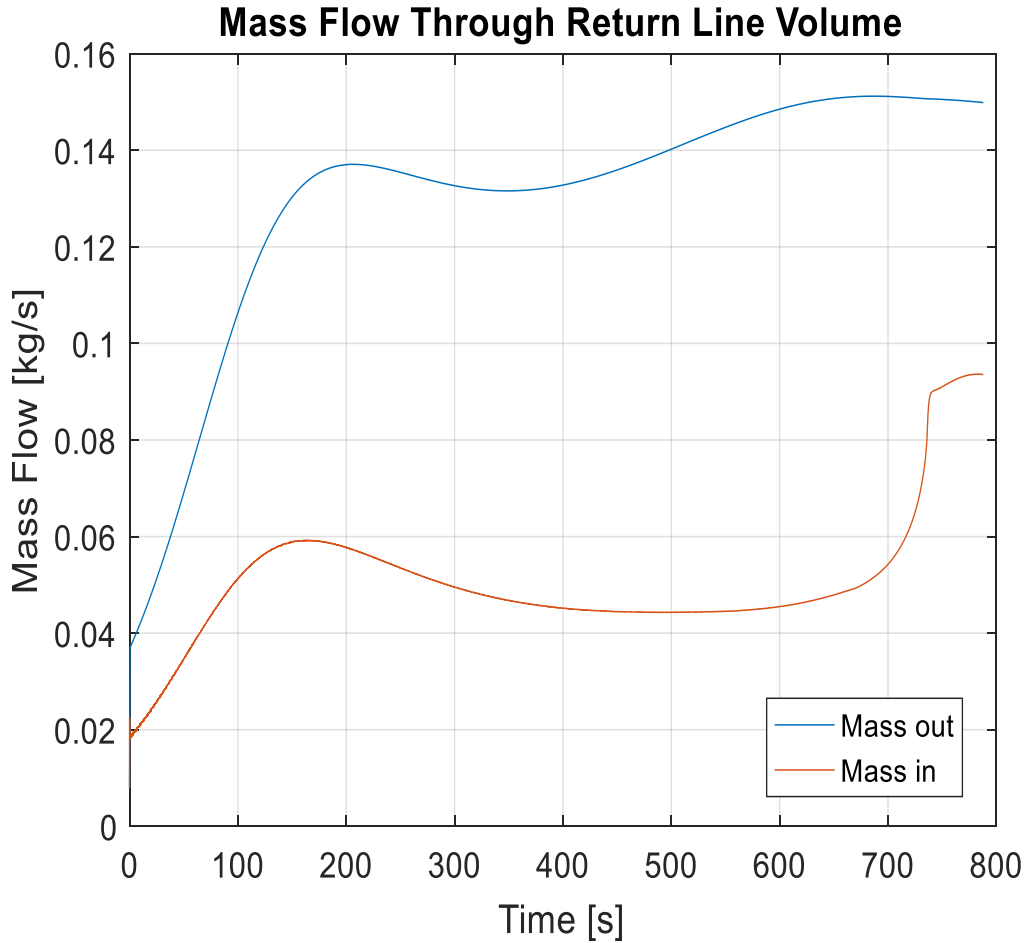
Figure 41. Return line volume flow at inlet and outlet during load dependent case

Figure 41 details how the mass flow into and out of the return line volume changed with time. This mimics what was seen in the constant mass flow case, and also shows the sharp increase of mass flow into the return line volume toward the end of the pump down.
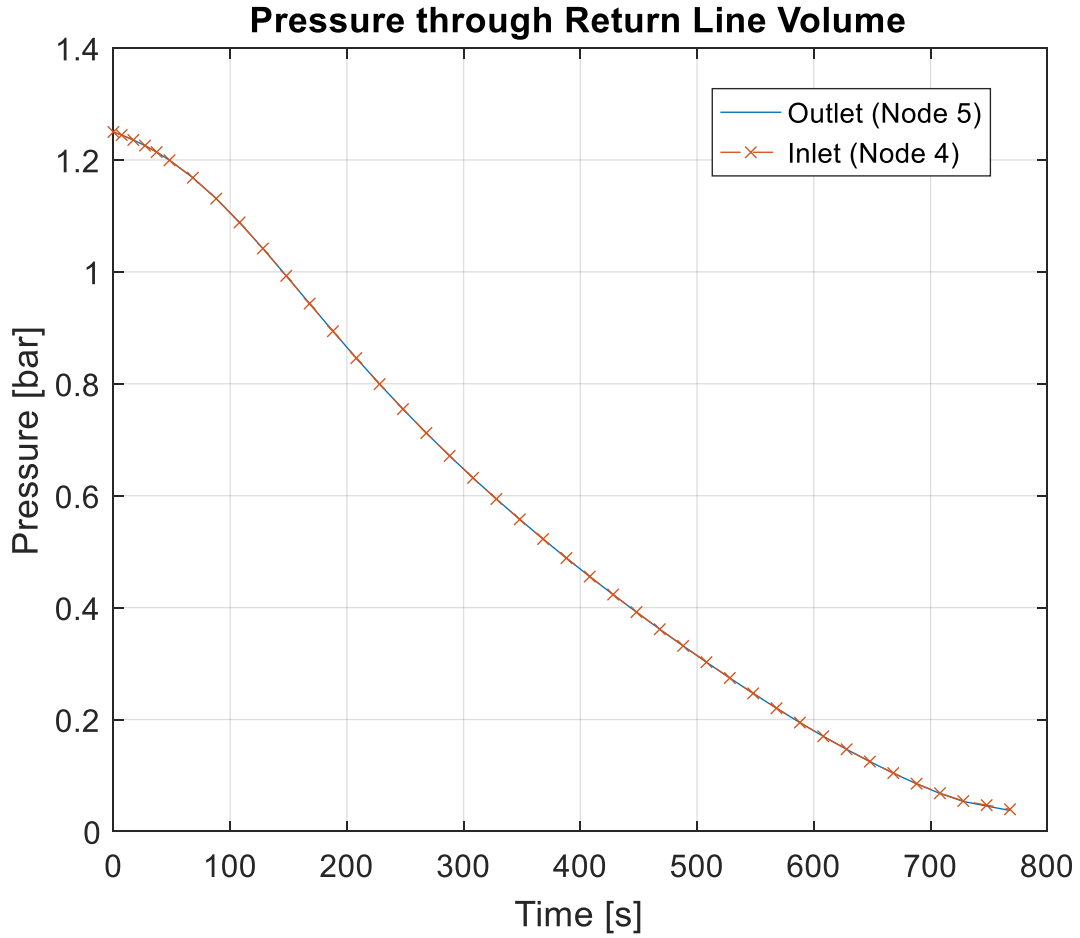
Figure 42. Return line volume inlet and outlet pressure

Figure 42 shows how the pressure at node 4 (inlet) and node 5 (outlet) varied over time. The

pressure drop across the line was sufficient to return mass to the inlet of the cold compressor.
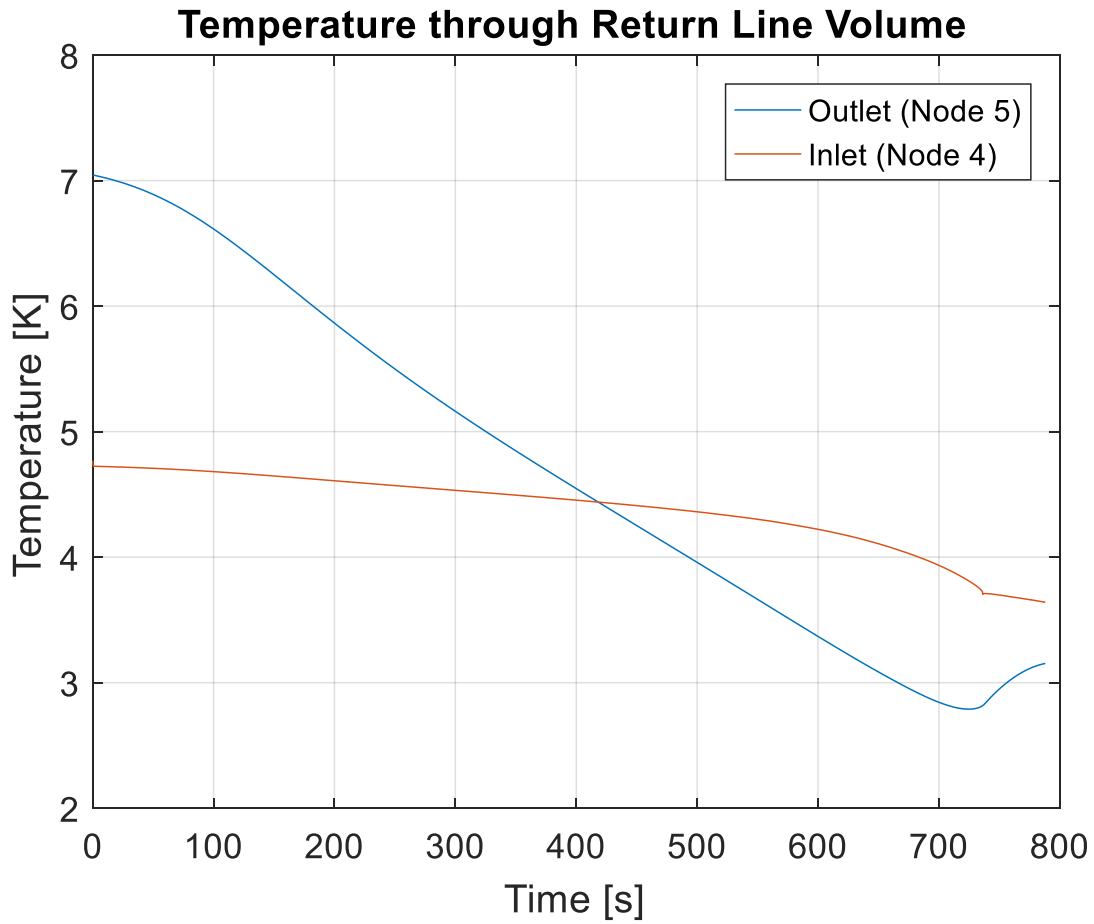
Figure 43. Return line volume inlet and outlet temperatures

Figure 43 shows how the temperatures at nodes 4 (inlet) and 5 (outlet) varied over time. Again, the return line volume cooled below that of the inlet fluid. This is a non-physical response and will be discussed further in Appendix A.

4.2.3. Cryo-module Load Dependent Response

Cryomodule Pressure
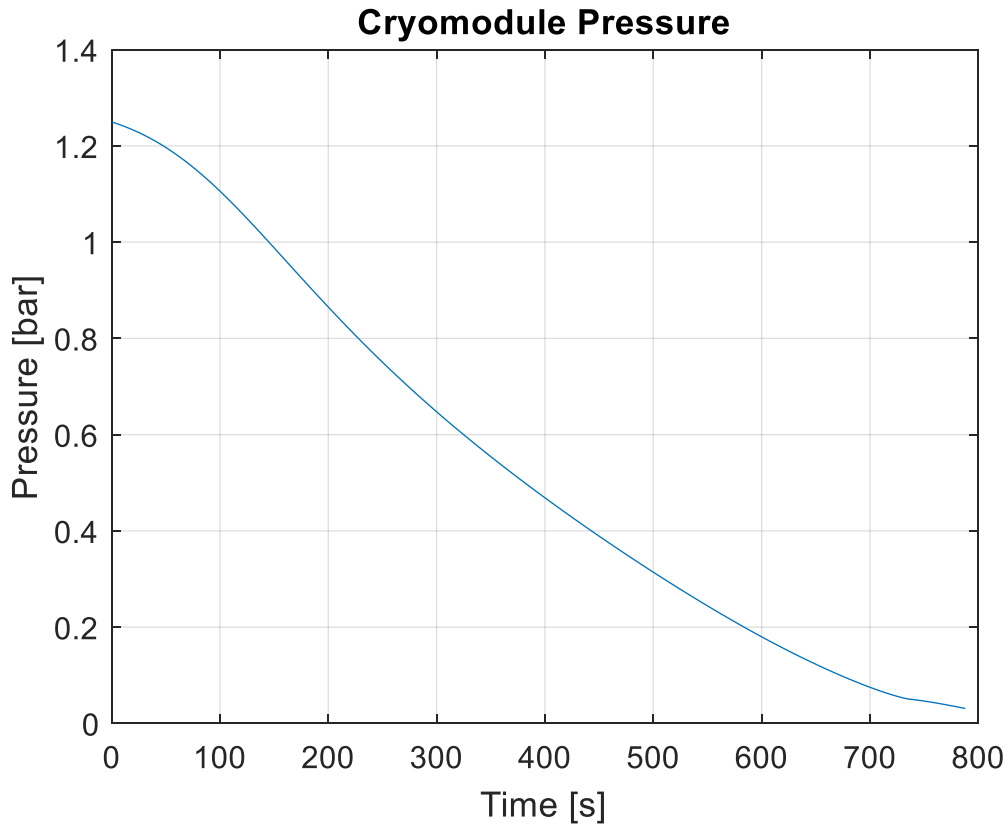
Figure 44. Cryo-module (node 3) pressure during the pump-down for the load dependent case

Figure 44 details the cryo-module load pressure over the pump-down. The pump-down duration is significantly longer than the constant mass flow case, as the maximum mass flow of the load pressure dependent case was roughly 40% less. However, the curve is comparable to the constant mass flow case, seen in Figure 31. In concert with the load pressure dependent pressure curve, the cryo-module (node 3) temperature curve follow suit in that it is nearly identical to the constant mass flow rate case (see Figure 45).
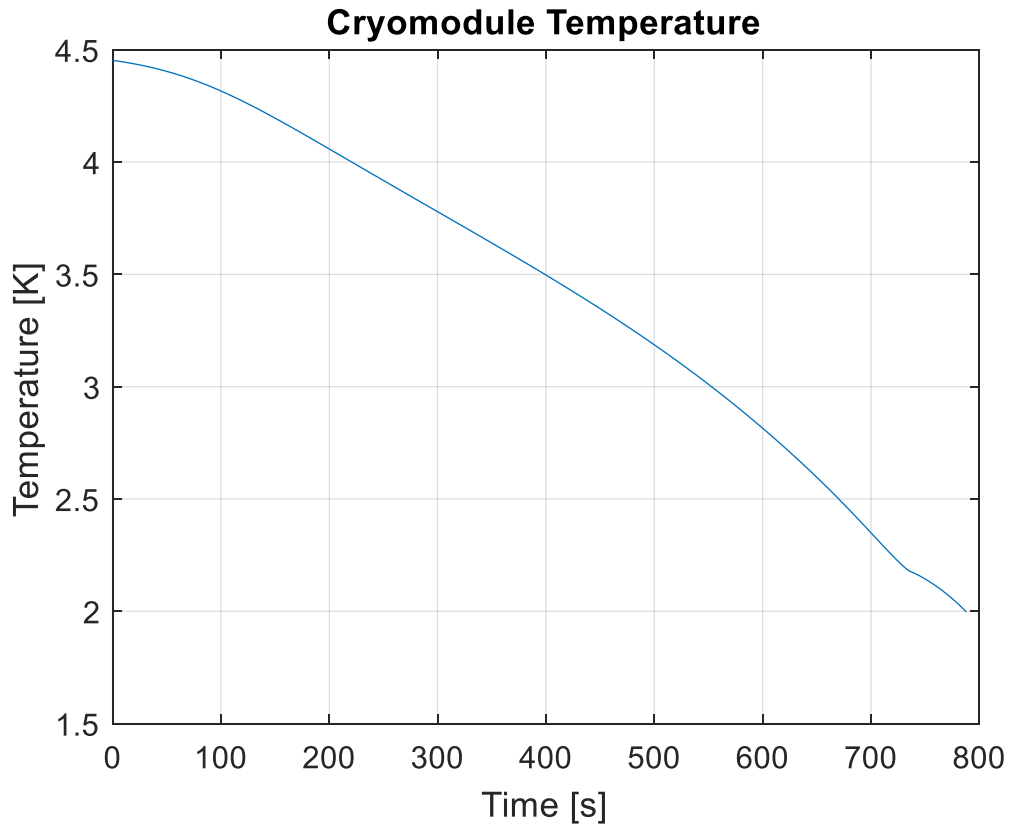
Figure 45. Cryo-module temperature during the load dependent case
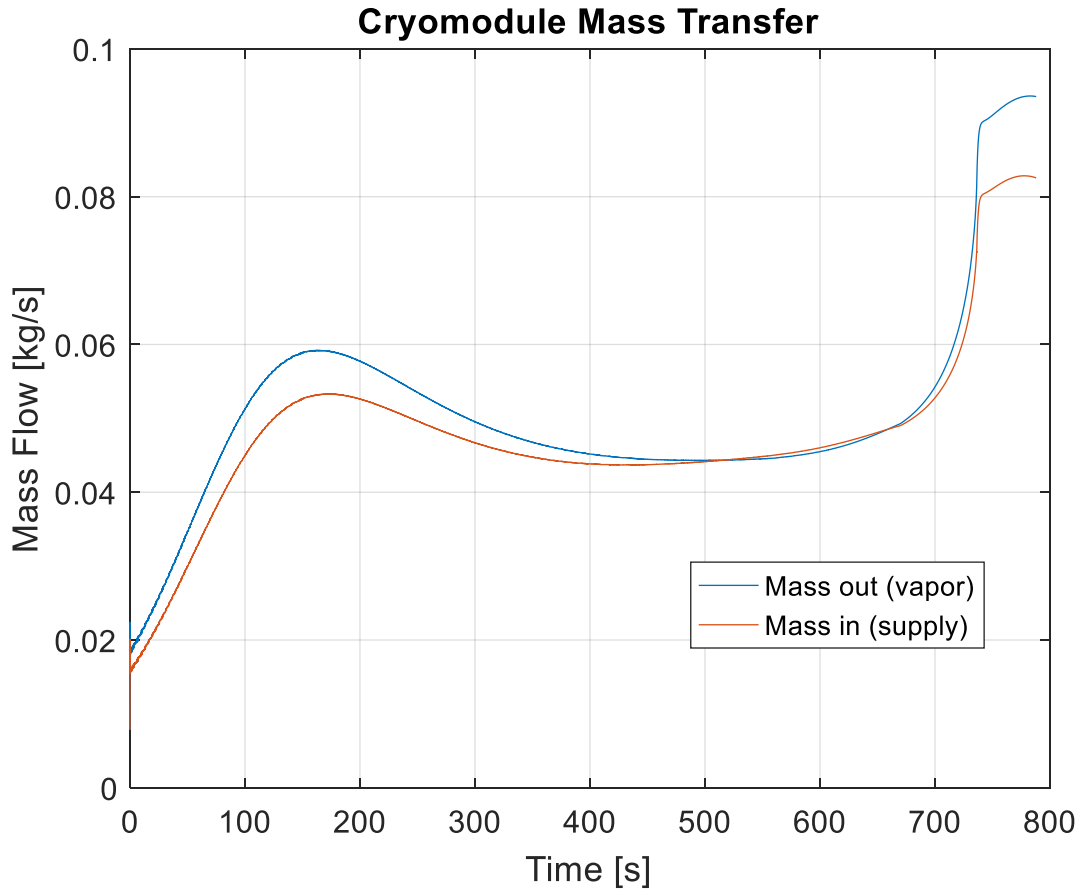
Figure 46. Cryo-module mass flow in and out during the pump down

Again, the mass flow into the cryo-module exceeded the mass flow out of the cryo-module and it still produced a depressurization condition. Figure 46 shows 142.62 s of mass accumulation in the cryo-module, at a maximum accumulation discrepancy between mass in and mass out of 1.15% over the course of 142.62 s.
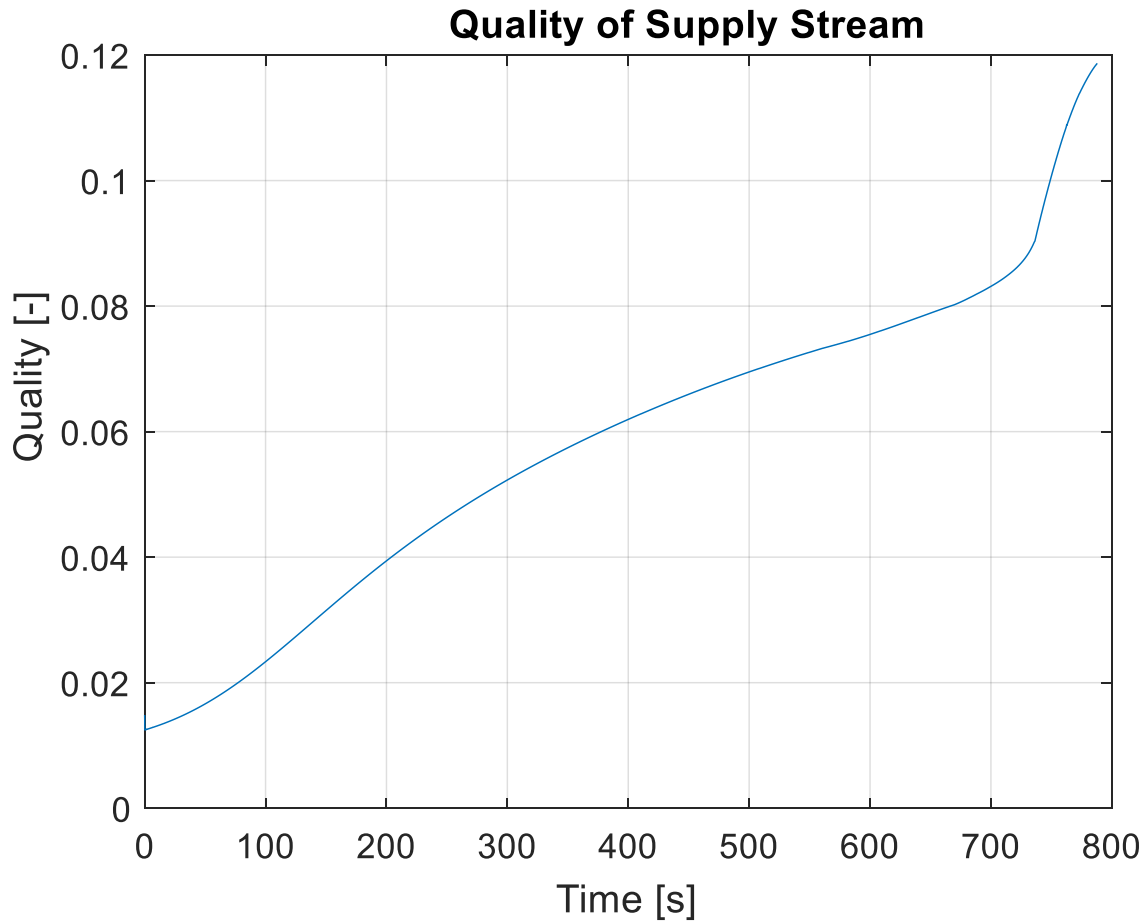
Figure 47. Quality of supply flow during pump down

Figure 47 shows two different perspectives on the supply mass flow quality. The load pressure shows that once the cryo-module is below the pressure corresponding to a saturation temperature at the lambda point, the quality increase dramatically.

4.2.4. Sub-Atmospheric Heat Exchanger Load Dependent Response

High-Pressure Stream Node Temperature
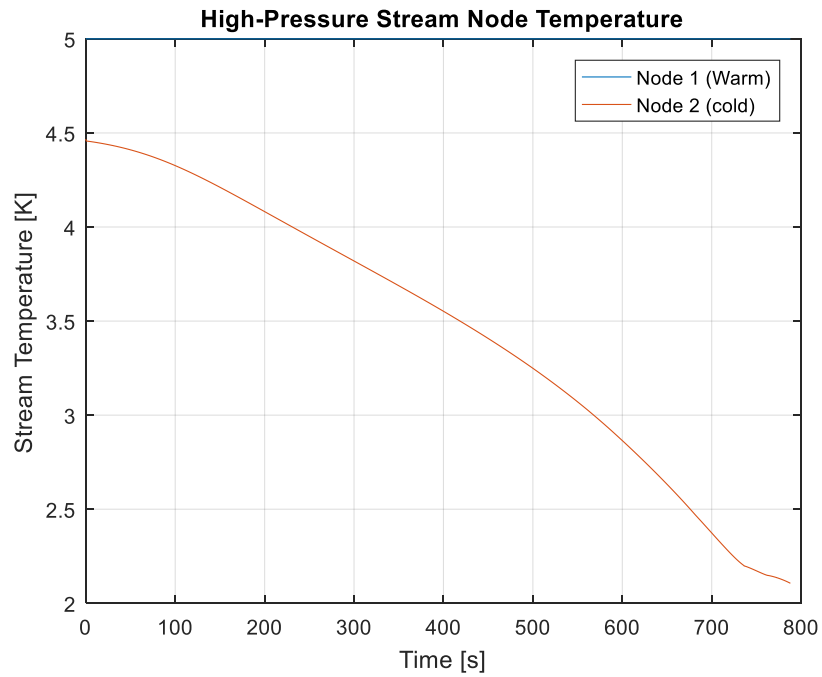
Figure 48. Sub-atmospheric heat exchanger high-pressure stream temperatures during load dependent pump-down

Figure 48 shows how significant the refrigeration recovery can be on the high-pressure stream that is supplying flow to the cryo-module.

Low-Pressure Stream Node Temperature
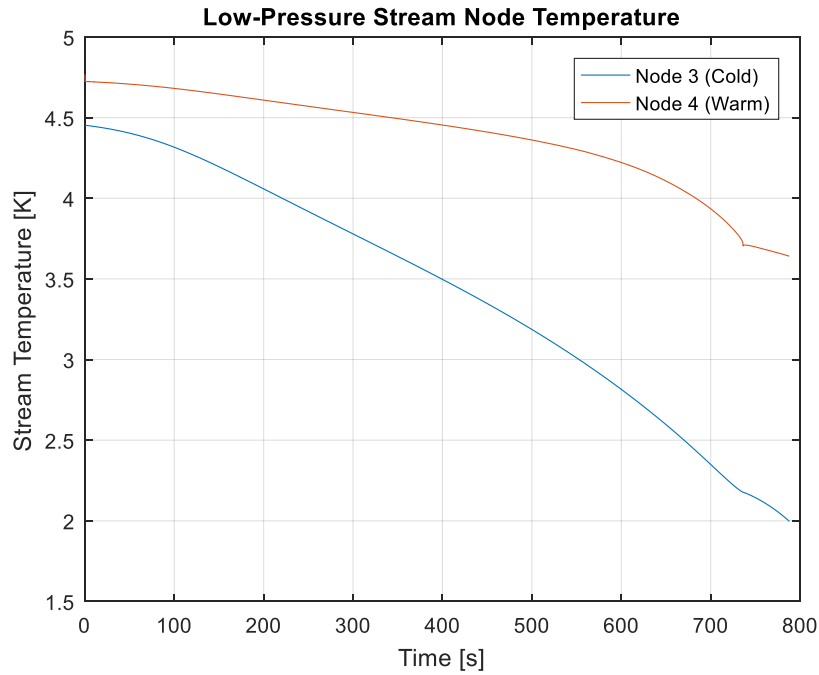
Figure 49. Sub-atmospheric heat exchanger low pressure stream temperatures during pump down

Figure 49 shows how, for the load pressure dependent process case, the low-pressure stream returning mass to the cold compressor inlet reacts to the refrigeration recovery. As the cryo-module depressurizes, the specific heat of the helium vapor drops. As a result, there is a larger temperature differential between node 3 and node 4.

5. Discussion

The solution of the model presented an array of challenges to be overcome. These challenges were the product of simplifications during modeling. As a result there were modifications that had to be made to achieve a solvable system.

In this paper, terms are used to refer to different portions of the pump-down process. Ramp-up is referred to when the portion of the pump-down where the mass flow rate across the cold compressor is increasing with time. Constant pump-down is referred to when the mass flow across the cold compressor is not changing. Ramp-down is referred to when the mass flow rate is decreasing with time. Mass flow profiles as function of time will be provided and portions commented for context when discussing specific sections.

5.1. Low Mass Flow at Initialization

In holding to the FRIB design case, the integrated model was initialized with a 21 [W] heat load on the cryo-modules. This produced a balanced mass flow rate through the cryo-module of roughly 1.1 $\left[\frac{g}{s}\right]$. When initializing the system this relatively small mass flow rate produces higher than expected temperatures at the inlet to the cold compressor, on the order of 50 K. While this calculated mass flow rate and temperature satisfied the model calculations, it is not an expected condition of the FRIB system. To avoid these high temperatures, the FRIB system employs bypass lines on the far side of the return transfer lines, band heaters in the cryo-module, and variable Joule-Thomson valves on the supply side to increase the mass flow rate while the system is at pressures and temperatures above the operating condition.
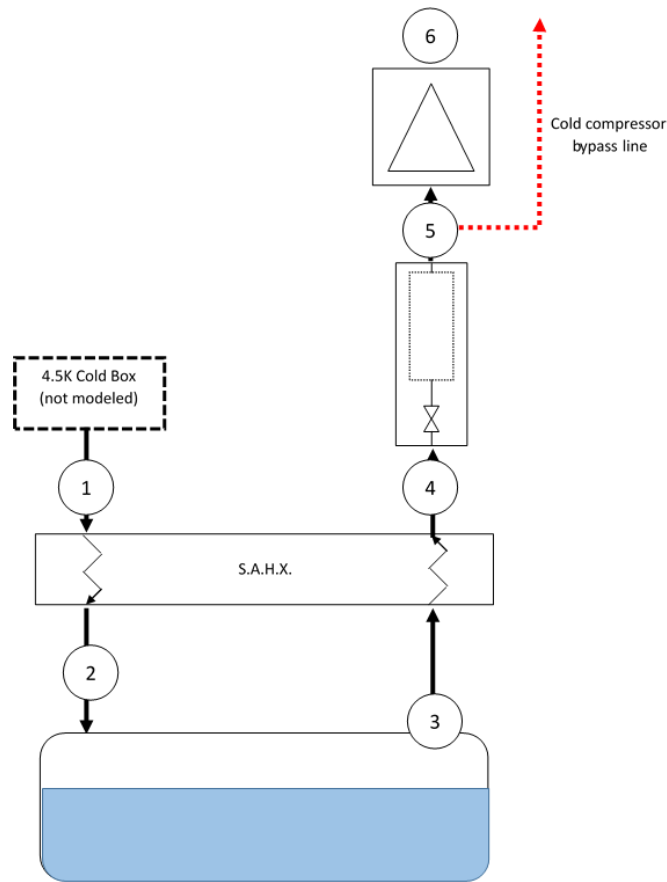
Figure 50. System model with dotted line showing non-modeled bypass line

Figure 50 shows the non-modeled bypass lines installed in the FRIB sub-atmospheric system. The model was modified to support mass flow rates at the initial condition that more appropriately reflect the FRIB system. The initial load heat was increased from the design case of 21 W to an artificially higher initial value of 521 W. At this point, the mass flow rate through the system was such that temperatures at the initialization were roughly 5.5 K. After it was determined that the system temperatures were appropriate at the initial condition, the program was started at the higher load and during the first steps of the solution the load heat was tapered during the solution from the artificially high 521 W to the design case 21 W. The load heat was reduced at a rate of $10 \frac{kW}{s}$ during the ramp up of the pump-down.

Load Dependent Return Line Pressure
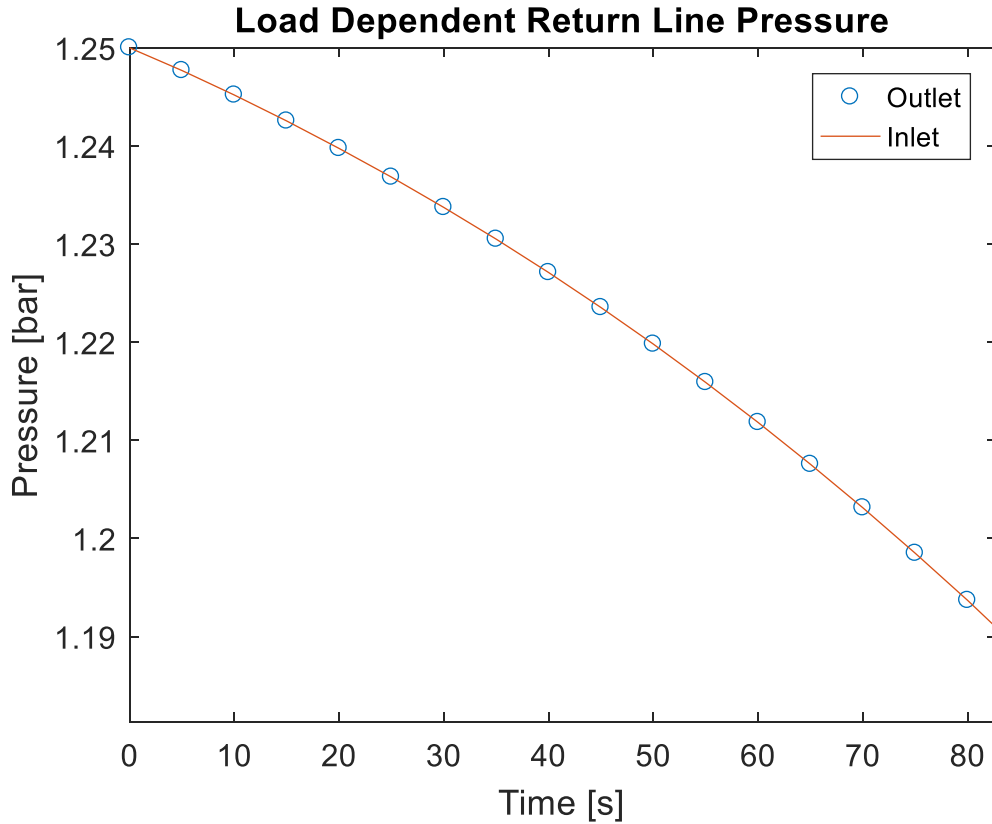
Figure 51. Load dependent return line pressure at outlet and inlet during ramp up

Over the majority of the pump down there is no appreciable effect on the return line volume pressures. Figure 51 shows the load dependent case pressures at the inlet and outlet during the initialization of the pump-down and the first few minutes of the process. The pressure curves are unaffected and showing smooth stable depressurization. However, reducing the resolution of the pump-down to the first two seconds (see Figure 52), the reduction in load heat has a significant effect on the return line volume pressure drop. That is, the rate at which the cryo-module depressurizes per a change in mass removal from the cryo-module, reduces. As such there is a lower pressure differential between the cryo-module and the far side of the return line volume.
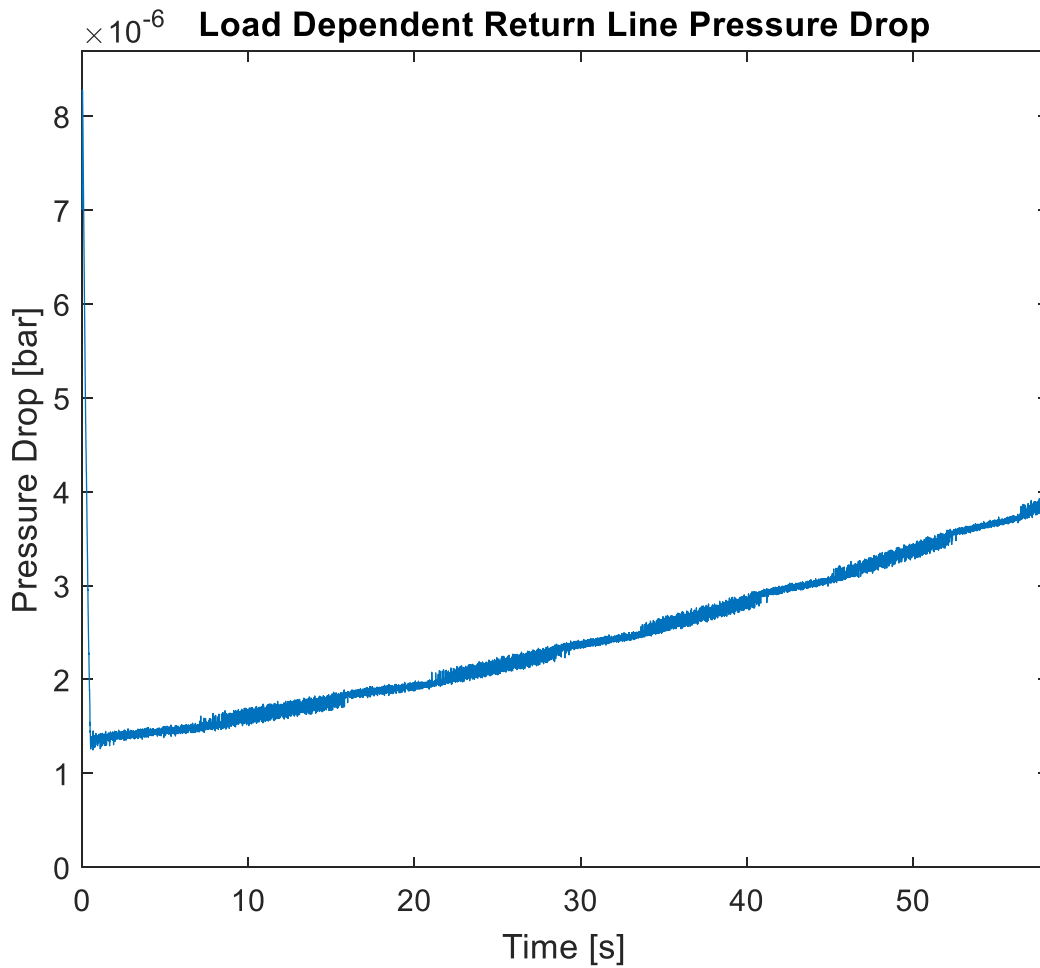
Figure 52. Load dependent return line pressure drop across component during ramp-up

5.2. Stability of Solution during Ramp Up

During the ramp up of the system the pressure drop across the return line volume must be stable. The time rate of change of pressure in the return line volume (node 5) must be such that the pressure drop is preserved between the dewar and the return line volume. However, large time steps and large changes in mass flow at node 5 can lead to high changes in density in the return line volume.

5.2.1. Solution time step change

The initial time step was reduced by an order of magnitude. Then at the time that the constant mass flow rate value was achieved, the time step was returned to the original value. It was expected at this point that the mass flow rates had stabilized and the system solution could be accelerated.
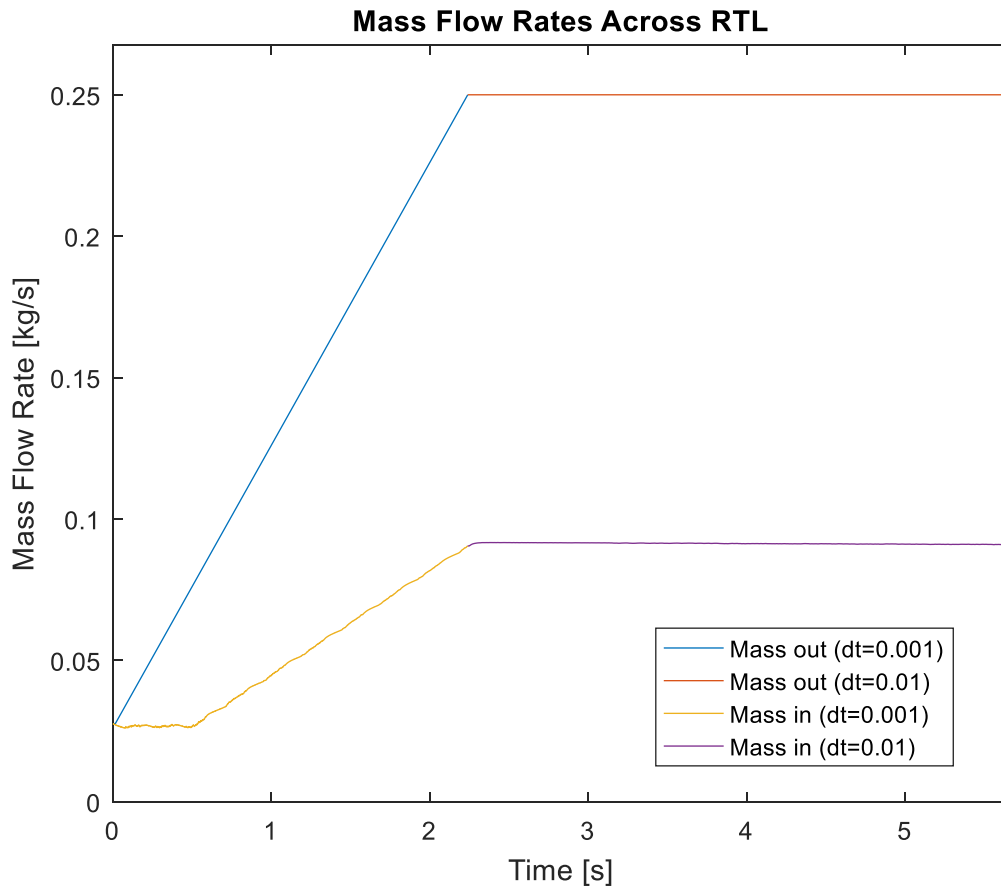
Figure 53. Effect of time step change on mass flow rate calculations in the Return Line Volume

Figure 53 shows how the mass flow into the return line volume (node 4) responds to a change in time step. It is evident that there is no appreciable change in system response at this moment in the solution. An evaluation of the pressure at the inlet and outlet of the return line volume can be seen in Figure 54. Again the inlet and outlet pressures do not react significantly in this instance.
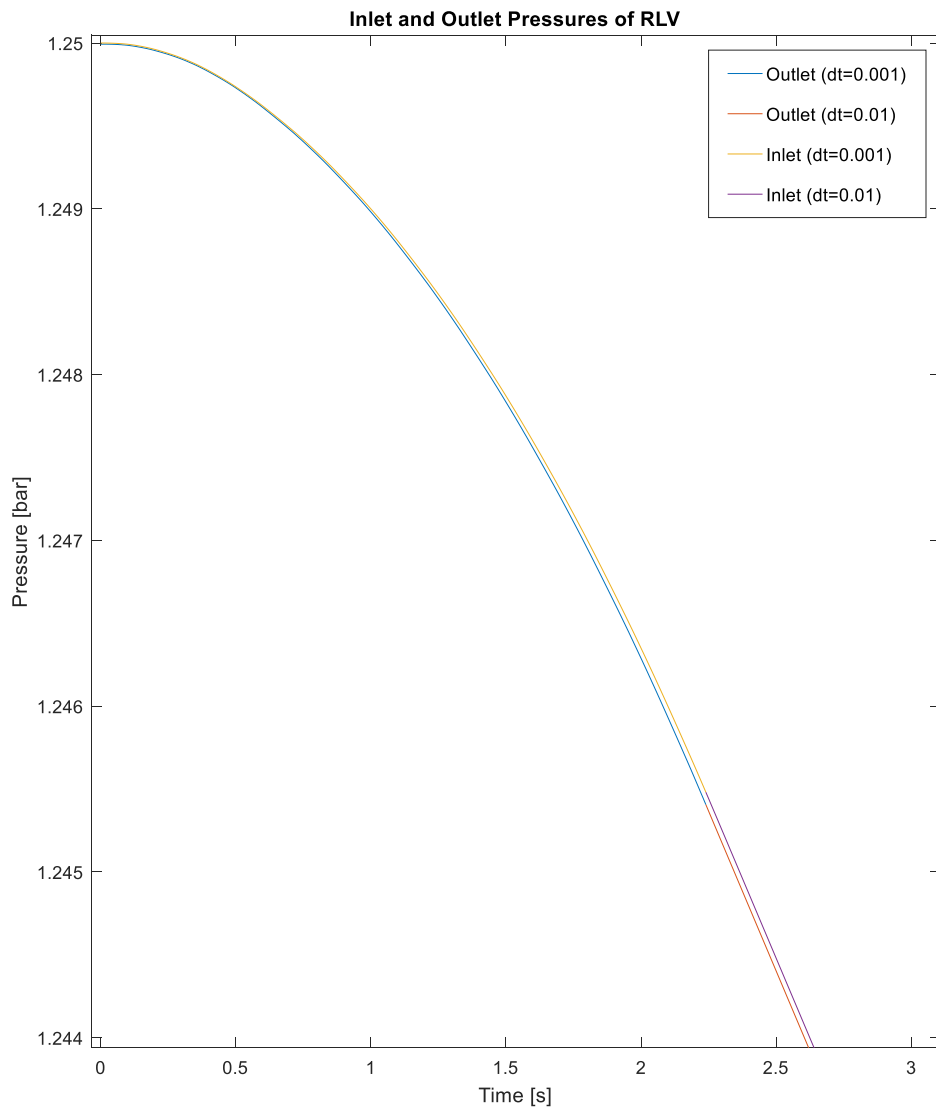
Figure 54. Inlet and outlet pressures of the return line volume during the initial depressurization of the cryo-module and the RLV

5.2.2. Rate of change of ramp-up mass flow during constant mass flow

case

As stated previously, rate of change of mass flow rate across the cold compressor and the

resulting mass flow rate at the inlet of the return line volume are unstable at the initial ramp-up

of the system. Therefore the rate of change of mass flow is increased slowly at the beginning of the simulation. This allows for a gradual change of pressure of the return line volume in concert with the depressurization of the cryo-module. Once there is a difference in the mass flow into and out of the return line volume of $0.4 \frac{g}{s}$ the time rate of change of mass flow across the cold compressor is increased from $1 \frac{g}{s^2}$ to $10 \frac{g}{s^2}$.
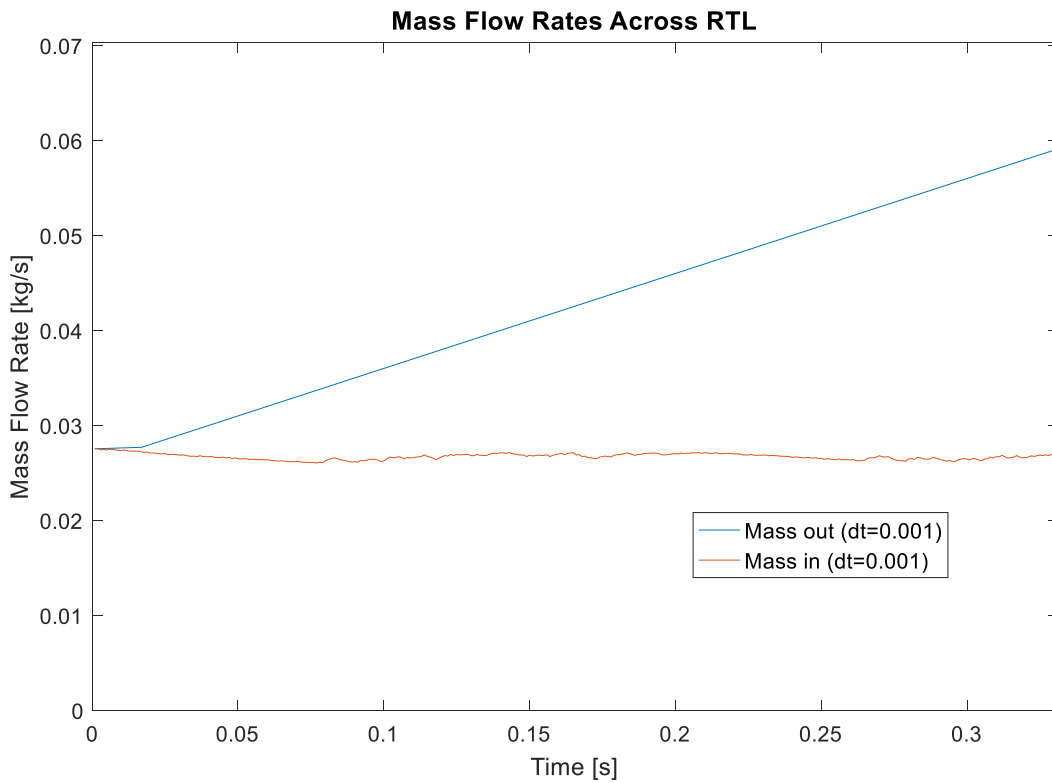
Figure 55. Effect of time rate of change of mass flow rate across the cold compressor on mass flow rate into and out of the return line volume

The effect of the step change in the rate of change of mass flow rate during the ramp up of the constant mass flow rate solution can be seen in Figure 55. It is interesting that the mass flow rate into the return line volume is decreasing at the initial stages of the solution. However, Figure 54 shows that the pressure differential between the inlet and the outlet of the return line volume at the beginning of the pump-down is very small. As the flow rates develop past the non-linear

portion of the pressure curves and the pressure differential begins to widen, the tracking of the inlet mass flow rate with the outlet flow rate begins to follow.

This was similar in the load dependent case. There was a discontinuous jump in mass flow from the initial time step to solving the process path polynomial. The discontinuity can be seen with the reduction in load heat. The initial pump-down was solved with a time step of 0.005 s and then incremented to 0.01 s once the mass flows at the inlet and outlet of the return line volume were developed (Mass out of return line volume 92% higher than mass in).

Load Dependent Return Line Volume Mass Flow
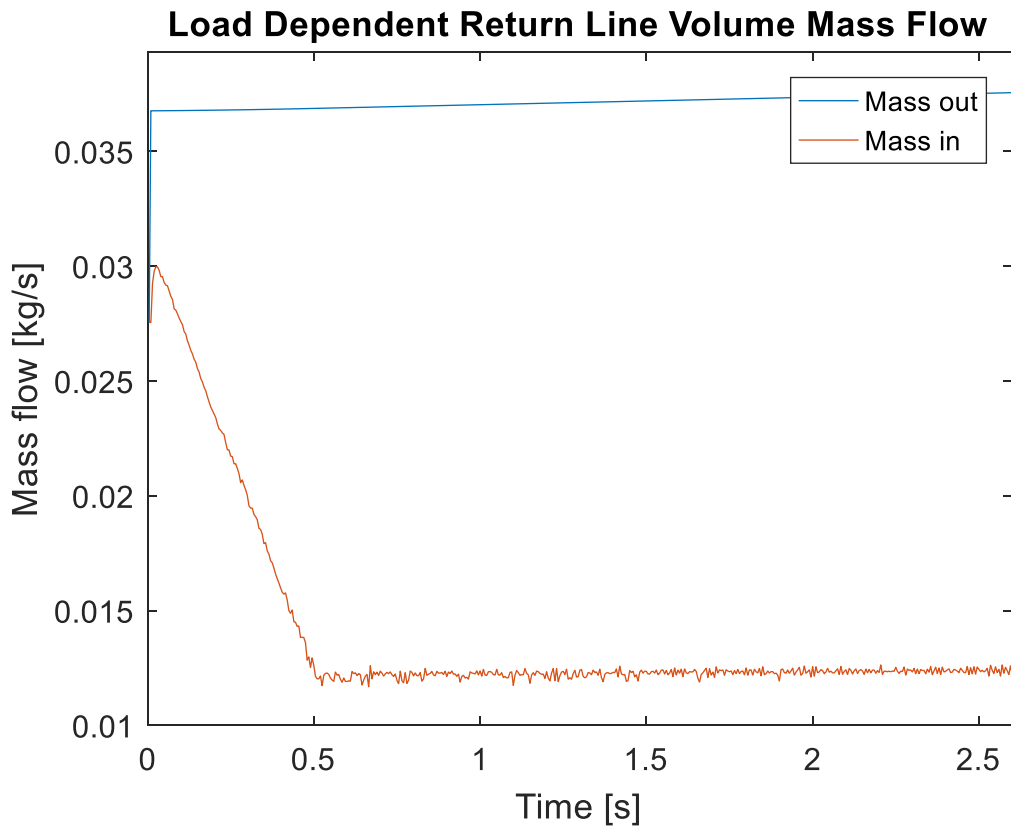
Figure 56. Return line volume mass flow at inlet and outlet during initialization of pump-down with a time step of 0.005 s

5.3. Cryo-module Mass Accumulation during Constant Mass Flow Rate Depressurization

The constant mass flow rate case produced a condition where the cryo-module was depressurizing while mass was accumulating in the vessel. The major model assumptions for the cryo-module depressurization was that the volume of the saturated liquid was held constant, the load heat goes solely into the saturated liquid, and there is only saturated liquid and vapor present in the cryo-module. These constraints do not preclude the possibility of an unbalanced mass flow. Furthermore, due to the discrepancy being relatively small (1.15%), it was determined acceptable for the constant mass flow rate simulation.

5.4. System Model Transience

The system model pump-down is never modified to approach a steady state. This would be accomplished by increasing the heat load on the SRF cavities in the cryo-module. That could be accomplished by either turning on the beam or increasing the heat output from the installed heaters in the cryo-module headers.

5.5. Cold Compressor Frequency Settings

There was an issue that resulted in the need to set the impeller frequency differently during the load dependent solution. The isentropically solved angular velocity was not sufficient to produce the desired pressure ratios. As a result impeller angular velocities were manually selected to achieve the desired pressure ratios at node six. This resulted in a sporadic curve for efficiency.

5.6. Removal of simplifications for further study at FRIB

The removal of model simplifications is a tedious subject; however, a series of simplifications for each model was presented in this paper and must be addressed in order to more accurately model the sub-atmospheric system as designed at FRIB.

5.6.1. Cold Compressors

The existing model only has one cold compressor. FRIB will utilize a cold compressor train of five cold compressors. Each cold compressor will have to be modeled and an algorithm designed for simulated operation given changing boundary conditions (inlet and outlet pressures and temperatures).

5.6.2. Return line volume

The existing model does not account for the numerous components installed in the transfer lines. Bayonets, distribution boxes, geometry changes at headers, and other complicating factors that would affect the return line volume model have been simplified into a simple geometry constant. That constant also omits the reality of a changing Reynold's number that would affect the pressure drop through the volume.

5.6.3. 4.5 K to 2 K heat exchanger

The modeling of the heat exchanger metal as in thermal equilibrium with the streams was a major simplification. In elaborating the heat exchanger model, allowing the construction material to be cooled by the streams during the pump-down would be necessary to more accurately simulate the heat exchanger performance. There are other 'parasitic' loads such as heat in-leak into each stream that would improve the accuracy of the model. Furthermore, the component was

modeled with no pressure drop. The addition of that effect would work in concert with the return line volume to remove mass from the cryo-module.

5.6.4. Dewar

The saturation condition that was modeled in this study only allows for a small number of initial conditions that the system could see. Furthermore, the two-phase non-interacting container model that was derived here is a simplification. A more elaborate model that allows heat to affect the saturated vapor, thus allowing superheated vapor as a component phase in the cryo-module dewar process, would be necessary to more accurately reflect the reality of the system.

6. Conclusions

The transient nature of the LINAC depressurization needs extensive modeling to predict the behavior of the downstream components. The sub-atmospheric refrigeration system at FRIB was divided into discrete component models and combined to model the effects of the overall system. The cold compressor was modeled as a series of ducts with correlations to model real system losses. The return line volume was modeled as a two part system, a friction based pressure drop and a depressurizing control volume with a constant heat load on the volume. The cryo-module was modeled as a vessel containing a two saturated phases (vapor and liquid) with a heat load that varied initially to support mass flowing through the system and quickly tapered back to a constant load heat. Lastly the sub-atmospheric heat exchanger was modeled as a variable specific heat device by dividing the heat exchanger up into a series of constant specific heat divisions that have a local specific heat at each division.

A major parameter for system response during depressurization is the refrigeration process path which must be prescribed to the system by the operators. The FRIB 2 K refrigeration system commissioning will include process path studies to empirically determine acceptable paths; however, the modeling proposed in this study is limited to the simplified models as stated above with initial conditions limited to the saturated liquid/vapor combination. Different initial conditions lead to different thermal loading, mass flow conditions, and system dynamics that relate to whether or not there exists a condition that could produce large cold vapor surges through the system. The separation of the four component models and evaluation with time produced solutions of cryo-module dewar load pressure with time and cold compressor polytropic efficiency with time. These models were solved for different prescribed process paths to prove the process model. Using efficiency information and observing the effect of different

process paths on the component models and the overall system model, it is possible to find theoretically optimal process paths. That is, mass flow rates that ensure efficient operation of the cold compressor and limit stress on the system.

In large scale cryogenics systems such as FRIB, a low stress and efficient operating state is desirable. Models such as this, in conjunction with diligent experimentation and careful validation can provide a 'road map' to an optimal operating condition.

7. Recommendation for Future Study

Elaboration on the component models and integrated system model will be necessary to accurately describe the conditions throughout the system and adequately model the process(es) for use as a predictive tool at FRIB. Four cold compressors must be added to the system and correct impeller geometries for each must be used in the compressor performance prediction sub-routines. The return transfer line volume must be elaborated upon to more accurately describe the effects of heat in-leak, friction related pressure drop, and complex fluid dynamics of vapor through the volume. The sub-atmospheric heat exchanger must be modeled as a Collins type heat exchanger which FRIB utilizes. This type of heat exchanger uses a relatively larger pipe to handle the low pressure return flow from the LINAC and a smaller tube, handling the high pressure supply flow to the LINAC, wound helically around the larger pipe to recuperate refrigeration from the return flow. The cryo-module must be modified to track the mass fraction of superfluid helium and the normal helium component, accurately describe the physical helium distribution in the system, as well as introduce ancillary systems that are present in the cryo-module such as band heaters, helium vents, and others.

8. Appendix A: Return Line Temperature Inversion

In both the constant mass flow case and the load dependent mass flow case, the outlet temperature (node 5) decreased below the inlet temperature (node 4). This is recognized as a non-physical response. It is unclear why this occurred in these simulations. In order to produce results free of this non-physical behavior, a second set of studies was performed that modeled the return line volume as steady pipe flow. This omitted any return line volume mass and energy accumulation.

In this case the mass flow rate across the cold compressor was equal to the mass flow leaving the cryo-module dewar. The pressure drop was assessed as before and the effect of the heat in-leak was established using (111).

$$dh = h_5 - h_4 = \frac{\dot{m}_5}{q_{in-leak}} \tag{111}$$

This produced temperature curves that do not show the outlet temperature decreasing below the inlet temperature.
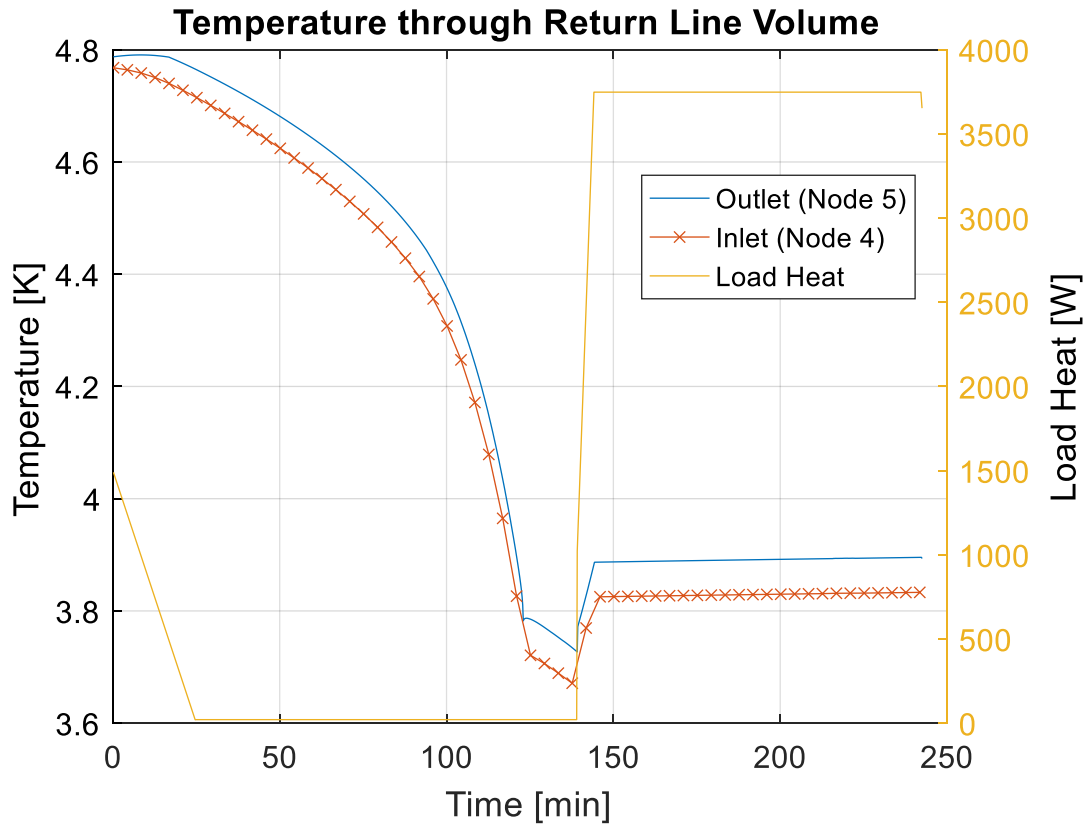
Figure 57. Constant mass flow rate case return line volume temperatures and load heat

The constant mass flow rate case with the steady-pipe modeled return line volume (Figure 57) shows how the system will pump down and be stabilized at the pump-down final condition. The stabilization of the system at the pump-down final condition is achieved by increasing the load heat at the cryo-module.
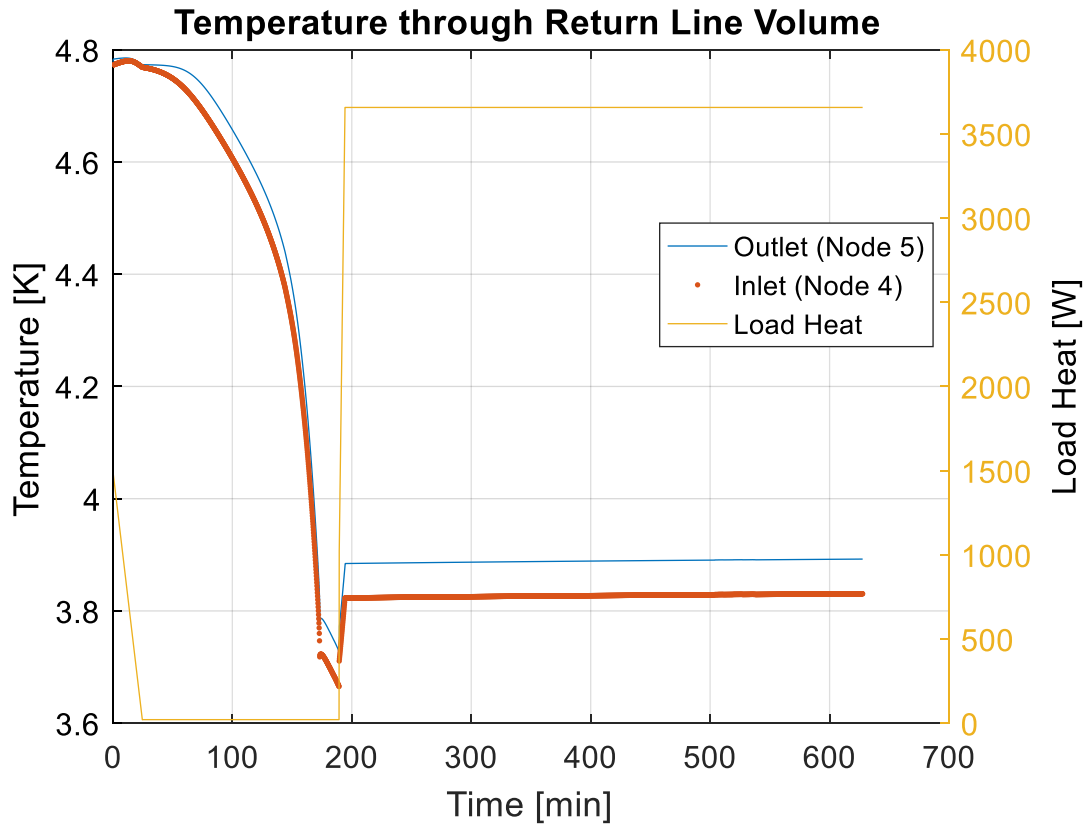
Figure 58. Load dependent case return line volume temperatures and load heat

Both the constant mass flow case (Figure 57) and the load dependent case seen in Figure 58 show how the return line volume temperatures react during the pump-down. Again, it is shown that the temperatures never cross during the pump-down or the stabilization.

Future research is necessary, in order to more accurately model the return line volume mass and energy accumulation.

9. Appendix B: Code

9.1. Integrated System Code

```matlab
%% Initialize the Program
%Clears the command window and workspace
clc;
clear all;

%Open an ActiveX server to handle running HePak macros
ExcelApp = actxserver('Excel.Application');
%Link the worksheet containing the macros (Must tie to spreadsheet
%containing Prop_functions module)
ExcelApp.Workbooks.Open('C:\Users\dinger\Desktop\Sat Dewar Depressurization.xlsm');

%% Set solution and model specific constants
T = zeros(6,100000);
p = zeros(6,100000);
x = zeros(1,100000);
ms= zeros(1,100000);
mg = zeros(1,100000);
mdot = zeros(1,100000);
Vdot = zeros(1,100000);

%Solution related values
dt       = 10;              %time step                    [s]
dmdt     = 0.0001;          %change in mass flow per unit time [kg/s^2]
dqdt     = 1;              %change in load heat for ramping [W/s]
dqdtstabilize = 100;
loadcheck = 0;

FID      = 10;              %Helium fluid ID for HePAK   [-]
q_load(1)   = 1500;    %heat input into liquid           [W]
p(3,1)   = 1.25; %initial pressure of the dewar      [bar]
pfinal   = 0.031;         %final pressure            [bar]
z        = 1;              %iterating parameter         [-]


%subatmospheric heat exchanger model related values
P_supply    = 3;     %supply flow pressure at inlet to SAHX      [bar]
T_supply    = 5;     %supply flow temperature at inlet to SAHX    [K]
HX_NTU       = 3;  %heat exchanger coefficient                 [-]
relax_fact  = 0.35; %relaxation factor                         [-]

%return volume model related values
qTL      = 500;       %heat leak into transfer line [W]

%% System Initialization
%{
The initialization of the pump-down model begins by assuming that the cold
compressor is spinning fast enough to remove the vapor begin generated by
heat in the dewar. That is: pressure is constant in the dewar. Therefore
the initial mass flow rate leaving the dewar is calculated
%}
P_dewar = p(3,z);
hv= ExcelApp.Run('hv_p',FID,P_dewar);        %sat vapor enthalpy     [J/g]
hl= ExcelApp.Run('hl_p',FID,P_dewar);        %sat liq enthalpy       [J/g]
```

```matlab
lambda = hv-hl;                                 %latent heat of mix      [J/g]
mg(1) = (q_load(1)/lambda)/1000;                %vapor removal mass flow[kg/s]


%{
with the mass flow due to evaporation of the liquid in the dewar known, it
is assumed that the supply mass flow is equal to the mass flow out of the
dewar. Therefore the enthalpy of the fluid before isenthalpic expansion
can be calculated using the SAHX model.
%}
i=1;
ms(1) = mg(1); %setting the supply mass flow equal to the vapor removal mass flow

p(1,z)  = P_supply;
T(1,z)  = T_supply;

[Tl_0,Ts_PX]     =
HX_Calc_NTU(FID,HX_NTU,ms(1),mg(1),P_supply,P_dewar,T_supply,relax_fact,ExcelApp);
hs                = ExcelApp.Run('h_pT',FID,P_supply,Ts_PX);
quality           = ExcelApp.Run('x_ph',FID,P_dewar,hs); %quality of fluid pre-
isenthalpic expansion
[check,msdotnew,dp(1)]     = D_Calc(FID,P_dewar,mg(1),q_load(1),quality,dt,ExcelApp);
dm               = mg(1)-msdotnew;



%this loop iterates until the differential between the mass flow rate into
%or out of the dewar are equal to with 10^-6
while or((dm > 10^-6), (dm<0))
    if dm > 10^-6
        while and(dm > 10^-10,mg(1)>0)
            mg(1)               = mg(1) - 0.001/i;
            ms(1)               = mg(1);
            [Tl_0,Ts_PX]     =
HX_Calc_NTU(FID,HX_NTU,ms(1),mg(1),P_supply,P_dewar,T_supply,relax_fact,ExcelApp);
            hs                = ExcelApp.Run('h_pT',FID,P_supply,Ts_PX);
            quality           = ExcelApp.Run('x_ph',FID,P_dewar,hs); %quality of fluid
pre-isenthalpic expansion
            [check,msdotnew,dp(1)]    =
D_Calc(FID,P_dewar,mg(1),q_load(1),quality,dt,ExcelApp);
            dm                = mg(1)-msdotnew;
        end
    else
        while or(dm < 0 ,mg(1)<0)
            mg(1)               = mg(1) + 0.001/i;
            ms(1)               = mg(1);
            [Tl_0,Ts_PX]     =
HX_Calc_NTU(FID,HX_NTU,ms(1),mg(1),P_supply,P_dewar,T_supply,relax_fact,ExcelApp);
            hs                = ExcelApp.Run('h_pT',FID,P_supply,Ts_PX);
            quality           = ExcelApp.Run('x_ph',FID,P_dewar,hs); %quality of fluid
pre-isenthalpic expansion
            [check,msdotnew,dp(1)]    =
D_Calc(FID,P_dewar,mg(1),q_load(1),quality,dt,ExcelApp);
            dm                = mg(1)-msdotnew;
        end
    end
    i=i+1;
end

p(2,z)  = P_supply; %set pressure at node2 equal to supply pressure
T(2,z)  = Ts_PX;    %set temperature at node2 equal to the output temperature of the
heat exchanger
ms(z)   = ms(1);    % set the supply mass flow rate at the first time step
```

```matlab
x(z)     = quality;   % set the quality at the first time step

p(4,z)   = p(3,1);    %set the pressure at node4 equal to the dewar pressure
T(4,z)   = Tl_0;      %set the temperature at node4 equal to the output of the low
pressure stream of heat exchanger

T(3,z)   = ExcelApp.Run('Tsat_p',FID,p(3,z)); %set the temperature in the dewar
dp(z)    = 0;     %set the dp value for the initial time step (dewar)

%initialize the return line volume knowing there is no depressurization of
%the volume
[Tcci,Pcci,Vdot(z)] = ReturnLineInitialize(Tl_0,P_dewar,mg(1),qTL,ExcelApp);

p(5,z)   = Pcci; %set pressure at node5
T(5,z)   = Tcci; %set the pressure at node5

z=z+1;   %increment iteration parameter

drho = 0; %initialize the rate of change of density value
du = 0;       %initialize the rate of change of internal energy value
dprv(1) = 0;      %initialize the rate of change of pressure value

%calculate the discrepancy between inlet and outlet flows in Return Volume
%mdisc(1) = (mg(1)-mdot(1))/mdot(1);
%calculate the disc. between inlet and outlet pressures in return volume
%pdisc(1) = (p(4,1)-p(5,1))/p(5,1);

%set the mass flow across the cold compressor equal to the mass flow rate
%along the return side of system for initialization
mdot(z-1) = mg(1);
n=1;
complete = 0;
stabilize = 0;
dpdirection = 0;
loaddir = 0;
below = 0;
above = 0;
%% System Solution Post-Initialization
while complete < 1
    tic %begin iteration timer for debugging

    %set cold compressor mass flow rate
     if mdot(z-1) < 0.180
        mdot(z) = mdot(z-1) + dmdt*dt;
     else
        mdot(z) = 0.180;
     end

    %Taper load heat for initial
    if and(and(loadcheck ==0 ,q_load(z-1) > 21),stabilize==0)
        q_load(z)  = q_load(z-1)-dqdt*dt;
    end

    if and(or(loadcheck > 0 , or(q_load(z-1)==21,q_load(z-1)<21)),stabilize==0)
        q_load(z)   = 21;
        loadcheck = 1;
    end

    if stabilize == 1
        if and(above==1,0.031-p(3,z-1) < 0)
```

```matlab
            above = 0;
            below =1;
            n = n+10;
        elseif and(0.031-p(3,z-1) > 0,below == 1)
            above = 1;
            below = 0;
            n=n+10;
        end
    end

    if and(stabilize ==0,p(3,z-1) < 0.031)
        stabilize = 1;
        above = 0;
        below = 1;
    end




    if stabilize == 1
        if and(and(p(3,z-1)<pfinal,dpdirection == 0),dpdt<0.025/n)
            q_load(z) = q_load(z-1)+dqdtstabilize*dt/n;
        elseif and(and(p(3,z-1)>pfinal,dpdirection > 0),dpdt>-0.025/n)
            q_load(z) = q_load(z-1)-dqdtstabilize*dt/n;
        else
            q_load(z) = q_load(z-1);
        end
    end

    %Solve return line volume for next time step pressure temperature and
    %current time step mass flow rate
    [p(5,z),T(5,z),Vdot(z)] = RL_Calc_NO(FID,p(4,z-1),T(4,z-1),qTL,mdot(z),ExcelApp);

    mg(z) = mdot(z);

    %calculate new return line volume discrepancies (mass balance and
    %pressure)
%      mdisc(z) = (mg(z)-mdot(z))/mdot(z);
%      pdisc(z) = (p(4,z-1)-p(5,z-1))/p(5,z-1);

    %calculate the next time step dewar pressure and current time step supply mass
flow rate
    [p(3,z),ms(z),dp(z),dpdt]   = D_Calc(FID,p(3,z-1),mg(z),q_load(z),x(z-
1),dt,ExcelApp);

    if dpdt > 0
        dpdirection = 1;
    else
        dpdirection = 0;
    end

    if stabilize >0
        if and(below == 1, dpdirection == 1)
            loaddir = 1;
        elseif and(above == 1,dpdirection==0)
            loaddir = 0;
        end
    end


    %calculate next time step dewar temperature
```

```matlab
    T(3,z)  = ExcelApp.Run('Tsat_p',FID,p(3,z));

    %set temperature and pressure of node 1 for next time step
    T(1,z)  = T_supply;
    p(1,z)  = P_supply;

    %Solve for high/low-pressure stream outlet temperatures for next time
    %step
    [T(4,z),T(2,z)]    = HX_Calc_NTU(FID,HX_NTU,ms(z),mg(z),P_supply,p(3,z-
1),T_supply,relax_fact,ExcelApp);

    %set pressure at inlet to return line volume (node4) for next time step
    p(4,z)  = p(3,z);

    %set pressure at high pressure stream outlet for next time step
    p(2,z)  = P_supply;

    %solve for quality of supply stream as it is expanded into dewar
    x(z)    = qualitycalc(FID,T(2,z),p(2,z),p(3,z),ExcelApp);

    if and(abs(p(3,z)-p(3,z-1))/p(3,z-1)<0.0001,abs(p(3,z)-pfinal)/pfinal <0.0001)
        complete = 1;
    end

    %increment iterating parameter
    z=z+1;

    toc%lap of execution stopwatch

end


ccp = p(5,1:250:z-1);
ccT = T(5,1:250:z-1);
ccmdot = mdot(1:250:z-1);
ccVdot = Vdot(1:250:z-1);

for i = 1:length(ccp)
[Pr(i),omega(i),isen(i),poly(i)] =
CC_Calc_Const(FID,ccp(i),ccT(i),ccmdot(i),ccVdot(i),i,ExcelApp);
end

%% Function declarations
function x = qualitycalc(FID,Ti,Pi,Pd,ExcelApp)
    hs  = ExcelApp.Run('h_pt',FID,Pi,Ti);
    x   = ExcelApp.Run('x_ph',FID,Pd,hs);
end
```

9.2. Return Line Volume Subroutine Code

```matlab
function [ P5_new,T5_new,mdotin,Vdot] =
RL_Calc(FID,dt,pin,Tin,p_tl,T_tl,q,mdotout,GeomConst,ExcelApp)

%      mdotout = mdotout;
    dp      = ((pin-p_tl)*10^5);

    V_tl    = 7; %[m^3]

    %Inlet properties
    hin     = ExcelApp.Run('h_PT',FID,pin,Tin)*1000;    %[J/kg]
    mu      = ExcelApp.Run('visc_PT',FID,pin,Tin)*10^-6; %[Pa-s]
    rho_in  = ExcelApp.Run('rho_pt',FID,pin,Tin);    %[kg/m^3]

    %Transfer line properties
    u_tl    = ExcelApp.Run('u_pt',FID,p_tl,T_tl)*1000; %[J/kg]
    rho_tl  = ExcelApp.Run('rho_pt',FID,p_tl,T_tl); %[kg/m^3]
    v_tl    = ExcelApp.Run('vol_pt',FID,p_tl,T_tl); %[m^3/kg]


    %Calculation of inlet mass flow rate
    rho = ExcelApp.Run('rho_pt',FID,pin,(Tin+T_tl)/2);    %[kg/m^3]
    mdotin      = sqrt((dp/GeomConst)*rho);

    drho    = (mdotin-mdotout)/V_tl; %[kg/m^3-s]
    du_tl   = (mdotin*hin + q - u_tl*(V_tl*drho + mdotout)-
mdotout*p_tl*100000*v_tl)/(V_tl*rho_tl);%[J/kg-s]

    rhonew  = dt*drho + rho_tl ; %[kg/m^3]
    unew    = dt*du_tl + u_tl; %[J/kg]


    P5_new  = ExcelApp.Run('p_ru',FID,rhonew,(unew/1000)); %[bar]
    T5_new  = ExcelApp.Run('T_ru',FID,rhonew,(unew/1000)); %[K]

%     dPrtl    = P5_new-p_tl;

    Mass = mdotin-mdotout-drho*V_tl;

    Vdot= mdotout/rho_tl;

    Energy = mdotin*hin - mdotout*ExcelApp.Run('h_pt',FID,p_tl,T_tl)*1000 + q -
V_tl*drho*u_tl - V_tl*rho_tl*du_tl;

    if abs(Mass) > 10^-5
        msgbox('Mass conservation failure in RTL');
        return;
    end

    if abs(Energy) > 10^-5
        msgbox('Energy conservation failure in RTL');
        return;
    end

end
```

9.3. Dewar Subroutine Code

```
function [pnew,msdot] = D_Calc(He,p,mg,q,quality,dt,ExcelApp)
%DEWARSOLVER Summary of this function goes here
%   Detailed explanation goes here

    mgdot = mg*1000;            %mass flow rate of gas leaving the dewar    [g/s]
    Vd   = 1200;           %total volume of dewar                    [L]
    Vlfrac = 0.4;          %liquid volume fraction                   [-]
    x = quality;             %supply flow quality                        [-]


    T   =   ExcelApp.Run('Tsat_p',He,p);    %Dewar temperature        [K]

    hv= ExcelApp.Run('hv_p',He,p);        %sat vapor enthalpy      [J/g]
    hl= ExcelApp.Run('hl_p',He,p);        %sat liq enthalpy        [J/g]
    lambda = hv-hl;                       %latent heat of mix      [J/g]

    LVol = Vd*Vlfrac;                     %liquid volume           [L]
    VVol = Vd*(1-Vlfrac);                 %vapor volume            [L]

    hsupply = (1-x)*hl+x*hv;              %supply flow enthalpy    [J/g]

    vv = ExcelApp.Run('volv_p',He,p);    %vapor specific volume   [L/g]
    vl = ExcelApp.Run('voll_p',He,p);    %vapor specific volume   [L/g]

    mv = VVol/vv;                         %vapor mass              [g]
    ml = LVol/vl;                         %liquid mass             [g]

    uv = ExcelApp.Run('uv_p',He,p);      %vapor s.internal energy  [J/g]
    ul = ExcelApp.Run('ul_p',He,p);      %liquid s.internal energy [J/g]

    cvv = ExcelApp.Run('cvv_p',He,p);    %vapor const. vol. specific heat capacity
[J/g-K]
    cvl = ExcelApp.Run('cvl_p',He,p);    %liquid const. vol. specific heat capacity
[J/g-K]

    kv = ExcelApp.Run('isoKv_p',He,p);   %sat. vapor isothermal compressibility
[1/Pa]
    kl = ExcelApp.Run('isoKl_p',He,p);   %sat. liq isothermal compressibility
[1/Pa]

    Bv = ExcelApp.Run('Vexpv_p',He,p);   %sat. vapor volume expansivity      [1/K]
    Bl = ExcelApp.Run('Vexpl_p',He,p);   %sat. liquid volume expansivity     [1/K]

%     dT = 0.0000001;                                %differential temperature for
differencing  [K]
%     psata = ExcelApp.Run('psat_T',He,T-dT/2);    %LHS, saturation pressure   [Pa]
%     psatb = ExcelApp.Run('psat_T',He,T+dT/2);    %RHS, saturation pressure   [Pa]
    dpdTsat = ExcelApp.Run('dp_dt_satp',He,p);                       %central difference
calculation to produce slope of vapor pressure curve [Pa/K]

    dvldpsat = vl*(Bl/dpdTsat - kl);
    dvvdpsat = vv*(Bv/dpdTsat - kv);

    duvdpsat = cvv/dpdTsat + 100*dvvdpsat*(T*Bv/kv - p);
    duldpsat = cvl/dpdTsat + 100*dvldpsat*(T*Bl/kl - p);
```

```matlab
    mltilda = ml*p/vl * dvldpsat;
    mvtilda = mv*p/vv * dvvdpsat;


    Ultilda = ml*duldpsat;
    Uvtilda = mv*duvdpsat;
    Utilda = Ultilda + Uvtilda;


    vvtilda = 100*vv - ((1-x)*lambda)/p;
    vltilda = 100*vl + (x*lambda)/p;


    Vltilda = mltilda*vltilda;
    Vvtilda = mvtilda*vvtilda;


    dpdt = (q-mgdot*(1-x)*lambda)/(Utilda + Vltilda + Vvtilda);


    mldot = -mltilda * dpdt/p;
    mvdot = -mvtilda * dpdt/p;
    msdot = (mldot+mvdot+mgdot);


    Mass = mldot+mvdot-(msdot-mgdot);
    Energy = (dpdt*(Uvtilda+Ultilda)+mldot*ul+mvdot*uv) - (q +msdot*hsupply-mgdot*hv);
    Kinematic = (dpdt*ml*dvldpsat+vl*mldot)+(dpdt*mv*dvvdpsat+(vv*mvdot));


    if abs(Mass) > 10^-5
        msgbox('Mass conservation failure in Dewar');
        return;
    end


    if abs(Energy) > 10^-5
        msgbox('Energy conservation failure in Dewar');
        return;
    end


    if abs(Kinematic) > 10^-5
        msgbox('Kinematic conservation failure in Dewar');
        return;
    end



    msdot=msdot/1000;

    difp = dpdt;
    pnew = difp*dt + p;
end
```

9.4.4.5 to 2 K Heat Exchanger Subroutine Code

```matlab
function [T_low_out,T_high_out] =
HX_Calc_NTU(He,NTU,mhd,mld,pres_h,pres_l,Th,sigma,ExcelApp)
% Heat Exchanger Model - Solve for Temperature of helium leaving cryomodule toward
cold compressors
Check = 1;
disccheck = 1;

N           = 10;                           %number of HX divisions
mfhdot      = mhd*1000;                      %flow rate of high pressure fluid
[g/s]
mfldot      = mld*1000;                      %Flow rate of low pressure fluid
[g/s]
ph          = pres_h;                        %pressure of high pressure fluid [bar]
pl          = pres_l;                        %pressure of low pressure fluid [bar]
Th(1)       = Th;                            %inlet temperature of hp fluid [K]
Tl(N+1)     = ExcelApp.Run('tsat_p',He,pl) + 0.0001;    %inlet temp of lp fluid [K]

NTUi        = NTU/N;        %differential UA per subdivision of HX

Ch(1)       = mfhdot*ExcelApp.Run('Cp_pT',He,ph,Th(1));     %Heat capacity at constant
pressure of hp fluid [J/K]
Cl(N+1)     = mfldot*ExcelApp.Run('Cp_pT',He,pl,Tl(N+1));   %Heat capacity at constant
pressure of lp fluid [J/K]

CRest       = min(Ch(1),Cl(N+1))/max(Ch(1),Cl(N+1));    %Heat capacity ratio for
initializing the temperature vectors
Thetaest    = exp((1-CRest)*NTU);   %
Thetaesti   = exp((1-CRest)*NTUi);  %

dThl(1)     = (1-CRest)/(Thetaest-CRest)*(Th(1)-Tl(N+1));
Tl(1)       = Th(1)-dThl(1);
dTli        = (Tl(1)-Tl(N+1))/N;

%generate summing vectors and initialize the first space.
qlsum(1)    = 0;
qhsum(1)    = 0;
UAsum(1)    = 0;
NTUsum(1)   = 0;


%initialize the temperature profiles with estimates
for i = 1:N
    Tl(i+1)=Tl(i)-dTli;
    dThl(i+1)=dThl(i)*Thetaesti;
    Th(i+1)=Tl(i+1)+dThl(i+1);
end

% main loop of program, handles
while or(abs(Check) > 0.0001,abs(disccheck) > 0.001)

    for i = 1:N+1
        dThl(i)=Th(i)-Tl(i);
        hh(i)= ExcelApp.Run('h_pT',He,ph,Th(i));
        hl(i)= ExcelApp.Run('h_pT',He,pl,Tl(i));
    end

    for i = 1:N
```

```matlab
        dTh(i+1)           = Th(i)-Th(i+1);
        dTl(i+1)           = Tl(i)-Tl(i+1);
        dTlogmean(i+1)  = (dThl(i)-dThl(i+1))/log(dThl(i)/dThl(i+1));
        dql(i+1)           = mfldot*(hl(i)-hl(i+1));
        qlsum(i+1)         = qlsum(i)+dql(i+1);
        dqh(i+1)           = mfhdot*(hh(i)-hh(i+1));
        qhsum(i+1)         = qhsum(i)+dqh(i+1);
        Ch(i+1)            = dqh(i+1)/dTh(i+1);
        Cl(i+1)            = dql(i+1)/dTl(i+1);
        dNTU(i+1)          = NTUi;
        NTUsum(i+1)        = NTUsum(i)+dNTU(i+1);
        dUA(i+1)           = NTUi*min(Ch(i+1),Cl(i+1));
        UAsum(i+1)        = UAsum(i)+dUA(i+1);
        dqlh(i+1)          = abs(dqh(i+1)/dql(i+1) -1);
        CR(i+1)            = Cl(i+1)/Ch(i+1);
        Theta(i+1)         = exp((1-CR(i+1))*NTUi);
    end

    FracUA            = UAsum/UAsum(N+1);
    FracNTU           = NTUsum+NTUsum(N+1);
%%
    AMatrix = zeros(N,N);
    i=1;
    j=1;
    for k = 1:2:N*2
        AMatrix(k,j)= -Theta(i+1);
        AMatrix(k,j+1)= Theta(i+1);
        AMatrix(k,j+2)= 1;
        AMatrix(k,j+3)= -1;
        AMatrix(k+1,j)= -1;
        AMatrix(k+1,j+1)= CR(i+1);
        AMatrix(k+1,j+2)= 1;
        AMatrix(k+1,j+3)= -CR(i+1);
        i=i+1;
        j=j+2;
    end
    AMatrix(:,1)=[];
    AMatrix(:,N*2+1)=[];

    for i = 1:(2*N)
        if i == 1
            BMatrix(i)        = -Theta(i+1)*Th(i);
        else if i ==2
            BMatrix(i)     = -Th(i-1);
        else if i/2 == N
            BMatrix(i-1)       = -Tl(i/2+1);
            BMatrix(i)     = -CR(i/2+1)*Tl(i/2+1);
        else
            BMatrix(i)        = 0 ;
            end
            end
        end
    end

%%
    Tnew= -AMatrix \flip(rot90(BMatrix));


    for i= 1:(2*(N+1))
        if mod(i,2) == 1
            if i == 1
                Thnew(1) = Th(1);
```

```matlab
            else
                Thnew((i+1)/2) = (1-sigma)*Th((i+1)/2)+ Tnew(i-1)*sigma;
            end
        else
            if i > 2*N +1
                break;
            else
                Tlnew(i/2) = (1-sigma)*Tl(i/2) + Tnew(i-1)*sigma;
                if i/2 == N
                    Tlnew(N+1) = Tl(N+1);
                end
            end
        end
    end

    for x=1:N
        Thdisc  = abs(Thnew-Th)/(Th(x));
        Tldisc  = abs(Tlnew-Tl)/(Tl(x));
    end

    disccheck= max([Thdisc,Tldisc]);

    Th = Thnew;
    Tl = Tlnew;

    Check = qlsum(N+1)/qhsum(N+1) - 1 ;

% figure
% plot(NTUsum,Th,NTUsum,Tl,NTUsum,dThl)
% title('HX Temperature Profile');
% xlabel('Thermal Length [NTU] ');
% ylabel('Stream Temperature [K]');
% legend('hp','lp','hp-lp');

end
T_low_out = Tl(1);
T_high_out = Th(N+1);
end
```

9.5. Cold Compressor Subroutine Code

```matlab
function [ Pract,omega,etatt,etapoly ] =
CC_Calc(FID,P_inlet,T_inlet,mdot,Vdot,step,ExcelApp)
%% global data figures
    R = 8314.4598;
    Rhe = R/ExcelApp.Run('MW',FID);
    pconv      = 100000;       % Pa to bar conversion
    sigthres   = 1*10^-4;

%% Evaluate inlet conditions for Pr and angular velocity
%relevant wheel radii for calculations
rwheel_i_hub    = 0.018;         %Radius of impeller wheel to hub [m]
rwheel_i_shroud = 0.044;         %radius of impeller wheel to [m]
rwheel_i    = rwheel_i_hub + (rwheel_i_shroud - rwheel_i_hub)/2;
rwheel_o    = 0.098;          %outer diameter of impeller wheel [m]

%Solve for Pressure ratio as a function of volumetric flow
Pr = (10*Vdot+1);
if or(Pr == 1, Pr<1)
    Pr = 1.1;
end

%Solve for Omega using isentropic and radial bladed angular momentum
%equation
if step <5
    omega=step*50;
else
    omega = angvel(FID,Pr,P_inlet,T_inlet,rwheel_o,ExcelApp);
end

omega = omega*0.75;



%% Call macro template
%ExcelApp.Run('Cp_pT',FID,P0xp,T0xp)

%% INPUT PARAMETERS

%Inlet
L_inlet     = 0.075; %length of inlet duct [m]
d_inlet     = 0.152; %diameter of inlet duct [m]
k_inlet     = 0.0015; %surface roughness of duct [m]


%Impeller
d_outlet3   = 0.005662;         %outlet diameter for the impeller stage
Z_b         = 20;            %number of blades on impeller
h_b          = 0.1;       %impeller blade height [m]
beta_b2     = -60*pi()/180;
beta_b3     = 0*pi()/180;        %impeller blade angle at exit (radially ending blade)
k_impeller  = 0.001;     %roughness of duct
eta         = 0.001;        %tip clearance [m]
% omega       = omega;    %Angular velocity of wheel [rad/s]
rxs         = rwheel_i_shroud;          %Radius of the shroud at inlet of impeller
rxh         = rwheel_i_hub;       %radius of hub at inlet of impeller
L_blade     = 0.136;            %chord length of impeller blade
L_theta     = rwheel_o - rxh;
```

```matlab
%Vaneless Diffuser
r_vx        = rwheel_o; %radius of shroud at inlet to vaneless diffuser duct
r_vy        = 0.3; %radius of shroud at outlet from vaneless diffuser
d_outlet4   = d_outlet3; %diameter of outlet of vaneless diffuser
b_vdh       = d_outlet3; %Passage height at inlet to vaneless diffuser
k_VD        = 0.01; %roughness of duct (Japikse 1982)

%% INLET CALCULATIONS
%duct geometry calculations
A1          = pi() * ((d_inlet/2)^2);

%duct fluid property parameters
P1          = P_inlet;
T1          = T_inlet;
gamma12     = ExcelApp.Run('Cp_pT',FID,P1,T1)/ExcelApp.Run('Cv_pT',FID,P1,T1);
c1          = sqrt(gamma12*T1*Rhe);
mu1         = ExcelApp.Run('visc_pT',FID,P1,T1)*10^-6; %Dynamic visc
rho1        = ExcelApp.Run('rho_pT',FID,P1,T1);
h1          = ExcelApp.Run('h_pT',FID,P1,T1);

%duct flow velocity parameters.
Q1          = mdot/rho1;
C1          = Q1/A1;
W1          = C1;
Mach1       = C1/c1;

%duct geometry calculations.
A2          = pi()*(rwheel_i_shroud^2 - rwheel_i_hub^2);

%MACH SOLVER.
    %{
    NOTE: Stagnation pressure&temp are used for solving across the inlet as there is
no
    relative mach no. in this calculation.
    %}
P01         = P1*(1+((gamma12-1)/2)*Mach1^2)^(gamma12/(gamma12-1));
T01         = T1*(1+(gamma12-1)/2 * Mach1^2);
sigma1      = 1;
sigma1error = 1;
guess = 10^-6;
Mach2 = MachSolve(guess,mdot,Rhe,T01,P01,gamma12,A1,A1,0,0,0,sigma1);
Mach2_Isen  = Mach2;
[T02i,P02i,T2i,P2i] = fluid_prop_calc(T01,P01,Mach2_Isen,gamma12,Rhe,0,0,sigma1);

while (abs(sigma1error) > sigthres)
    Mach2 = MachSolve(guess,mdot,Rhe,T01,P01,gamma12,A1,A1,0,0,0,sigma1);

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %duct fluid property parameters
    [T02,P02,T2,P2] = fluid_prop_calc(T01,P01,Mach2,gamma12,Rhe,0,0,sigma1);
    c2          = sqrt(Rhe*gamma12*T2);
    gamma23     = ExcelApp.Run('Cp_pT',FID,P2,T2)/ExcelApp.Run('Cv_pT',FID,P2,T2);
    mu2         = ExcelApp.Run('visc_pT',FID,P2,T2)*10^-6; %Dynamic visc
    rho2        = ExcelApp.Run('rho_pT',FID,P2,T2);
    h2          = ExcelApp.Run('h_pT',FID,P2,T2);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %duct flow velocity parameters
    C2          = Mach2*c2;
    Mach2_rel   = Mach2/cos(beta_b2);
    W2          = Mach2_rel*c2;
```

```matlab
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Inlet Loss coefficient calculation
    sigmaold = sigma1;
    sigma1 =
inlet_losses(Mach1,Mach2,gamma12,L_inlet,d_inlet,rho1,P01,W1,W2,mu1,k_inlet);
    sigma1error = (sigma1-sigmaold)/sigmaold;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Recalculate mach2 until sigma and mach2 converge
    guess = Mach2;
end

Pr_Inlet = P02/P01;
Tr_Inlet = T02/T01;
% Eff_Inlet_isen =
% Eff_Inlet_poly =

%END INLET CALCULATIONS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%% Impeller Solver Information
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%duct geometry information
A3          = d_outlet3*2*pi()*r_vx;
L_hyd       = 0.5*((1-d_outlet3)/cos(beta_b3));

%Mach no. Solver parameters
guess = 10^-6;
sigma2      = 1;
sigma2error = 1;

%Betay Solver Parameters
Beta_y      = 1*pi()/180;        %initial guess for relative flow angle
slip_coef   = 1-0.63*pi()/(Z_b); %calculate slip coefficient
betaerr = 1;
betathresh = 5*10^-4;
betacheck = 0;

%Relative fluid properties
rho02 = rho2*(1+((gamma12-1)/2)*Mach2^2)^(1/(gamma12-1)); %stagnation density
calculation
P02_rel = P2*(1+((gamma23-1)/2)*Mach2_rel^2)^(gamma23/(gamma23-1));
T02_rel = T2*((1+((gamma23-1)/2)*Mach2_rel^2)^(gamma23/(gamma23-1)))^(1-(1/gamma23));

U2          = rwheel_i*omega; %tangential velocity of blade at inlet
U3          = rwheel_o*omega; %Tangential velocity of blade at outlet (Tip speed)

while abs(sigma2error) > sigthres
    betaerr = 1;
    while (abs(betaerr) > betathresh)

        [Mach3_rel,omega] =
ImpellerMachSolve(guess,mdot,Rhe,T02_rel,P02_rel,gamma23,A2,A3,Beta_y,omega,rwheel_i,r
wheel_o,sigma2);

        U2              = rwheel_i*omega; %tangential velocity of blade at inlet
```

```matlab
        U3              = rwheel_o*omega; %Tangential velocity of blade at outlet (Tip
speed)

        %duct fluid property parameters
        [T03_rel,P03_rel,T3,P3] =
fluid_prop_calc(T02_rel,P02_rel,Mach3_rel,gamma23,Rhe,U2,U3,sigma2);
        c3              = sqrt(Rhe*gamma23*T3);

        %duct flow velocity parameters
        W3 = Mach3_rel * c3;
        Cm3 = W3*cos(Beta_y);
        Ct3 = U3 + Cm3*tan(Beta_y);
        C3  = (Cm3^2 + Ct3^2)^(1/2);
        alpha3 = atan2(Cm3,Ct3);
        Mach3           = C3/c3;

        P03             = P3*(1+((gamma23-1)/2)*Mach3^2)^(gamma23/(gamma23-1));
        T03             = T3*(1+((gamma23-1)/2)*Mach3^2);
        gamma34          = ExcelApp.Run('Cp_pT',FID,P3,T3)/ExcelApp.Run('Cv_pT',FID,P3,T3);
        mu3             = ExcelApp.Run('visc_pT',FID,P3,T3)*10^-6; %Dynamic visc
        rho3            = rho02*((P03*P3*T02*T03)/(P02*P03*T03*T3));
        h3              = ExcelApp.Run('h_pT',FID,P3,T3);

        betaold = Beta_y;
        Beta_y = flow_angle_calc(Beta_y,U3,beta_b3,slip_coef,W3);
        betaerr = (Beta_y - betaold)/betaold;
        betacheck = betacheck + 1;
        guess = Mach3_rel;
    end

    sigmaold = sigma2;
    sigma2 =
impeller_loss(mdot,Rhe,gamma23,T03_rel,omega,W3,C2,C3,Ct3,U3,L_theta,h_b,mu2,rho2,rho3
,k_impeller,rwheel_i,rxs,rxh,Z_b,rwheel_o,d_outlet3,eta,alpha3);
    sigma2error = (sigma2-sigmaold)/sigmaold;

end
%END IMPELLER SOLVER
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Vaneless Diffuser Solver Information
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%duct geometry information
A4 = d_outlet3*2*pi()*r_vy;
alpha_4 = 77*pi()/180;  %initial alpha_4 guess - solved for mach no.
alphacheck = 0;
sigma3 = 1; %initial guess for entropy gain for isentropic mach no. calculation
alphaerr = 1; %initialize the alpha angle error tracking variable
sigma3err = 1; %initialize the entropy gain error tracking variable
guess = 0.8;
Mach4 = guess;
relax = 0.975;
Thresh_VD = 10^-6;

while (abs(alphaerr) > Thresh_VD)||(abs(sigma3err)>Thresh_VD)
    Machold = Mach4;
    Mach4 = MachSolve(guess,mdot,Rhe,T03,P03,gamma34,A3,A4,alpha_4,0,0,sigma3);
    Mach4 = Machold*relax + Mach4*(1-relax);
    if alphacheck == 1
        Mach4_Isen = Mach4;
        [T04i,P04i,T4i,P4i] = fluid_prop_calc(T03,P03,Mach4_Isen,gamma23,Rhe,0,0,1);
```

```matlab
    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %duct fluid property parameters
    [T04,P04,T4,P4] = fluid_prop_calc(T03,P03,Mach4,gamma34,Rhe,0,0,sigma3);
    c4          = sqrt(Rhe*gamma34*T4);
    gamma45     = ExcelApp.Run('Cp_pT',FID,P4,T4)/ExcelApp.Run('Cv_pT',FID,P4,T4);
    mu4         = ExcelApp.Run('visc_pT',FID,P4,T4)*10^-6; %Dynamic visc
    rho4        = ExcelApp.Run('rho_pT',FID,P4,T4);
    h4          = ExcelApp.Run('h_pT',FID,P4,T4);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %duct flow velocity parameters
    sigmaold = sigma3;
    [sigma3,Cf_VD] =
vaneless_diffuser_loss(mdot,Rhe,T04,r_vx,r_vy,C3,U3,b_vdh,alpha3,gamma34,mu3,d_outlet3
,k_VD);
    sigma3err = (sigma3-sigmaold)/sigmaold;

    Ct4 = (Ct3*mdot*r_vx)/(r_vy*(mdot+2*Cf_VD*Ct3*pi()*r_vx*(r_vy-r_vx)*rho3));
    %Ct4 = (Ct3*r_vx*0.85)/r_vy;

    alphaold = alpha_4;
    alpha_4 = asin((Ct4*(1+((gamma34-
1)/2)*Mach4^2)^0.5)/(Mach4*sqrt(gamma34*Rhe*T04)));
    alphaerr = (alpha_4-alphaold)/alphaold;
    alphacheck = alphacheck +1;
    guess = Mach4;
  end

%END VANELESS DIFFUSER SOLVER INFORMATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Print Mach No's
Pract = P04/P01;
Tract = T04/T01;
etatt=(Pract^((gamma12-1)/gamma12)-1)/(Tract-1);
etapoly=(log(Pract)-(gamma12*log(Pract)))/(log(Pract)-gamma12*log((Pract -
Pract^(1/gamma12) +etatt*Pract^(1/gamma12))/etatt));

%% Function declarations
function [loss,Cf] =
vaneless_diffuser_loss(mdot,Rhe,T0y,rx,ry,Cx,UT,bx,alphax,gamma,mu,d,k)
    Re = mdot/(mu*d);
    Cf = k*((1.8*10^5)/Re)^0.2;
    dq = (Cf*rx*(1-((rx/ry)^1.5))*((Cx/UT)^2))/(1.5*bx*cos(alphax));
    sigma = (1-((gamma-1)/(gamma*Rhe*T0y))*UT^2*dq);
    loss=sigma;
end
function loss =
impeller_loss(mdot,Rhe,gamma,T0y,omega,Wy,Cx,Cy,Cty,Uy,L_theta,h_blade,mux,rhox,rhoy,k
,rx,rxs,rxh,Z_b,ry,by,eta,alpha_y)
    D_hyd       = (by + rx*2)/2;
    D           = D_hyd;
    CWthresh    = 1*10^-7;
    CWerror     = 1;
    i           = 1;
    Re          = Uy*ry/(mux/rhox);
    Cf          = 0.01;
    rhobar      = (rhox + rhoy)/2;

    %Colebrook-White equation solver -> Cf
    while abs(CWerror) > CWthresh
    lhs     = 1/sqrt(Cf);
```

```matlab
    rhs      = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
    CWerror  = lhs - rhs;


    if CWerror > 0
        while CWerror > 0
            Cf       = Cf + 0.0001 /i;
            lhs      = 1/sqrt(Cf);
            rhs      = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
            CWerror  = lhs - rhs;
        end
    elseif CWerror < 0
        while CWerror < 0
            Cf       = Cf - 0.0001 /i;
            lhs      = 1/sqrt(Cf);
            rhs      = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
            CWerror  = lhs - rhs;
        end
    end
    i=i+1;
    end


    %INTERNAL LOSSES
    %Incidence loss
    f_inc   = 0.5; %Incidence constant
    Wui     = omega*rxh; %relative tangential velocity at inlet (hub)
    dh_inc  = f_inc*Wui^2 /2; %stagnation enthalpy loss due to blade incidence

    %skin friction loss
    Wxt = omega*rxs;
    Wxh = omega*rxh;
    Ctx = 0;
    Wbar = (Ctx + Cy + Wxt + 2*Wxh + 3*Wy)/8;
    L_b = (h_blade -(by/2))*pi()/2;
    dh_sf = 2*Cf*L_b/D_hyd * Wbar^2;

    %blade loading losses
    dh_euler   = omega*((ry*Cty)-(rx*Ctx));
    Wxt        = sqrt((omega*rxs)^2 + Cx^2);
    Dxt        = rxs*2;
    Dy         = ry*2;
    Wx         = sqrt(Cx^2 + (omega*rx)^2);
    Df         = 1-(Wy/Wxt)+((0.75*dh_euler / Uy^2)/((Wxt/Wy) *((Z_b/pi())*(1-Dxt/Dy)
+ 2*Dxt/Dy)));
    dh_bl      = 0.05*Df^2 * Uy^2;

    %clearance loss
    Cmxm    = Cx;
    dh_cl   = 0.6 * (eta/by)*Cty*((4*pi()/(by*Z_b))*((rxs^2 - rxh^2)/((ry-
rxs)*(1+(rhoy/rhox))))*Cty*Cmxm)^0.5;

    %mixing loss
    eta_wake   = 0.25;
    b_star     = 1;
    dh_mix  = (1/(1+tan(alpha_y)^2))*((1-eta_wake-b_star)/(1-eta_wake))^2 *((Cy^2)/2);

    %disc friction loss
    if Re < 3*10^5
        f_df     = 2.67/(Re^0.5);
    else
        f_df     = 0.0622/(Re^0.2);
    end
    dh_df = f_df * ((rhobar*ry^2 * Uy^2)/(4*mdot));
```

```matlab
    %recirculation loss
    dh_rc = 8*10^-5 * sinh(3.5*alpha_y^3) * Df^2 * Uy^2;

    %leakage loss
    rbar = (rx+ry)/2;

    bx   = rxs-rxh;
    bbar = (bx+by)/2;

    dP_cl = (mdot*(ry*Cty - (rx*Ctx)))/(Z_b*rbar*bbar*L_theta);

    Ucl = 0.816*sqrt(2*dP_cl/rhoy);

    mdotcl      = rhoy*Z_b*eta*L_theta*Ucl;

    dh_leak = (mdotcl*Ucl*Uy)/(2*mdot);

    %sum dh's and solve for sigma
    dh_internal = dh_inc+dh_bl+dh_sf+dh_cl+dh_mix;
    dh_parasitic = dh_df+dh_rc+dh_leak;
    dh_total    = dh_internal + dh_parasitic;
    sigma = (1-((gamma-1)/(gamma*Rhe*T0y))*dh_total)^(gamma/(gamma-1));
    loss = sigma;
end
function Mach = MachSolve(guess,mdot,Rhe,T0x,P0x,gamma,Ax,Ay,angle,Ux,Uy,sigma)
%inputs guess,mdot,Rhe,T0x,P0x,gamma,Ax,Ay,angle,Ux,Uy,sigma
    P0x = P0x*100000;
    threshhold = 1*10^-6; %Threshhold for convergence of equation 5.5
    Mach_y_Prime = guess; %initial guess for Mach number
    Error = 1; %initialize error value to begin loop
    i = 1;
    LHS        = mdot*sqrt(Rhe*T0x/gamma)/(Ax*P0x);
    while abs(Error) > threshhold
        if Error > 0
            while Error > 0
                Mach_y_Prime = Mach_y_Prime + (0.001/i);
                if Mach_y_Prime > 1
                    msgbox('Solution has produced invalid flow: Mach > 1')
                    return
                end
                RHS         = Ay/Ax * cos(angle)*Mach_y_Prime*(1+((gamma-
1)/2)*Mach_y_Prime^2)^(-((gamma+1)/(2*(gamma-1)))) * sigma*(1+((gamma-
1)/(2*gamma*Rhe*T0x))*(Uy^2-Ux^2))^((gamma+1)/(2*(gamma-1)));
                Error = LHS-RHS;
            end
        elseif Error < 0
            while Error < 0
                Mach_y_Prime = Mach_y_Prime - (0.001/i);
                if Mach_y_Prime < 0
                    msgbox('Solution has produced invalid flow: Mach < 1')
                    return
                end
                RHS         = Ay/Ax * cos(angle)*Mach_y_Prime*(1+((gamma-
1)/2)*Mach_y_Prime^2)^(-((gamma+1)/(2*(gamma-1)))) * sigma*(1+((gamma-
1)/(2*gamma*Rhe*T0x))*(Uy^2-Ux^2))^((gamma+1)/(2*(gamma-1)));
                Error = LHS-RHS;
            end
        end
        i=i+1;
        if i>10^5
```

```matlab
                msgbox('Solution not converging. Check for complex numbers.');
                return;
            end
        end
    Mach = Mach_y_Prime;
end
function [Mach,omega] =
ImpellerMachSolve(guess,mdot,Rhe,T0x,P0x,gamma,Ax,Ay,angle,omega,ri,ro,sigma)
%inputs guess,mdot,Rhe,T0x,P0x,gamma,Ax,Ay,angle,Ux,Uy,sigma
    Ux          = ri*omega; %tangential velocity of blade at inlet
    Uy          = ro*omega; %Tangential velocity of blade at outlet (Tip speed)
    P0x = P0x*100000;
    threshhold = 1*10^-6; %Threshhold for convergence of equation 5.5
    Mach_y_Prime = guess; %initial guess for Mach number
    Error = 1; %initialize error value to begin loop
    i = 1;
    LHS         = mdot*sqrt(Rhe*T0x/gamma)/(Ax*P0x);
    while abs(Error) > threshhold
        if Error > 0
            while Error > 0
                Mach_y_Prime = Mach_y_Prime + (0.001/i);
                if Mach_y_Prime > 1
                    omega = omega + 1;
                    Ux          = ri*omega; %tangential velocity of blade at inlet
                    Uy          = ro*omega; %Tangential velocity of blade at outlet
(Tip speed)
                    Mach_y_Prime = 0.8;
                    i=1;
                end
                RHS         = Ay/Ax * cos(angle)*Mach_y_Prime*(1+((gamma-
1)/2)*Mach_y_Prime^2)^(-((gamma+1)/(2*(gamma-1)))) * sigma*(1+((gamma-
1)/(2*gamma*Rhe*T0x))*(Uy^2-Ux^2))^((gamma+1)/(2*(gamma-1)));
                Error = LHS-RHS;
            end
        elseif Error < 0
            while Error < 0
                Mach_y_Prime = Mach_y_Prime - (0.001/i);
                if Mach_y_Prime < 0
                    msgbox('Solution has produced invalid flow: Mach < 1')
                    return
                end
                RHS         = Ay/Ax * cos(angle)*Mach_y_Prime*(1+((gamma-
1)/2)*Mach_y_Prime^2)^(-((gamma+1)/(2*(gamma-1)))) * sigma*(1+((gamma-
1)/(2*gamma*Rhe*T0x))*(Uy^2-Ux^2))^((gamma+1)/(2*(gamma-1)));
                Error = LHS-RHS;
            end
        end
        i=i+1;
        if i>10^5
            msgbox('Solution not converging. Check for complex numbers.');
            return;
        end
    end
    Mach = Mach_y_Prime;
end
function loss = inlet_losses(Machx,Machy,gamma,L,D,rho,P0x,Wx,Wy,mu,k)
P0x = P0x*100000;
CWthresh    = 1*10^-6;
CWerror     = 1;
M           = Machx;
Wbar        = sqrt(Wx^2 + Wy^2);
i           = 1;
Re          = Wx*rho*D/mu;
```

```matlab
Cf            = 0.01;

%Colebrook-White equation solver -> Cf
while abs(CWerror) > CWthresh
lhs     = 1/sqrt(Cf);
rhs     = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
CWerror   = lhs - rhs;

if CWerror > 0
    while CWerror > 0
        Cf      = Cf + 0.0001 /i;
        lhs     = 1/sqrt(Cf);
        rhs     = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
        CWerror   = lhs - rhs;
    end
elseif CWerror < 0
    while CWerror < 0
        Cf      = Cf - 0.0001 /i;
        lhs     = 1/sqrt(Cf);
        rhs     = -2*log(2.51/(Re*sqrt(Cf))) + ((k/D)/3.72);
        CWerror   = lhs - rhs;
    end
end
i=i+1;
end

% if      (M > 0.4) && (abs((Machy - Machx)/Machx)<0.01)
%    dsR = (1/(gamma-1)) * (1/Machx^2 - 1/Machy^2) + (1/(gamma-1))*log(Machx^2 /
Machy^2) - (gamma/(gamma-1))*4*Cf*L/D;
%    loss = exp(-dsR);
% elseif  (M < 0.4) || (abs((Machy - Machx)/Machx)>0.01)
   loss = 1-((4*Cf*L*Wbar^2*rho)/(2*D*P0x));
% end
end
function [T0y_rel,P0y_rel,Ty,Py] =
fluid_prop_calc(T0x_rel,P0x_rel,Machy_rel,gamma,Rhe,Ux,Uy,sigma)
    T0y_rel         = T0x_rel*(1+((gamma-1)/(2*gamma*Rhe*T0x_rel))*(Uy^2-Ux^2));
    P0y_rel         = P0x_rel*(T0y_rel/T0x_rel)^(gamma/(gamma-1))*sigma;
    Py          = P0y_rel/((1 + ((gamma-1)/2) * Machy_rel^2)^(gamma/(gamma-1)));
    Ty          = T0y_rel/(1+(gamma-1)/2 * Machy_rel^2);
end
function beta_y = flow_angle_calc(beta,Uy,beta_by,slip,Wy)
    thresh = 1*10^-6;
    error = 1;
    By = beta;
    i = 1;
    while abs(error)>thresh
        lhs = sin(By);
        rhs = (((-Uy*(1-slip))/Wy) + tan(beta_by)*(1+tan(beta_by)^2 - ((Uy^2 *(1-
slip)^2)/(Wy^2)))^0.5)/(1+tan(beta_by)^2);
        error = lhs-rhs;
        if error > 0
            while error > 0
                By = By -0.001 / i;
                if By < -45*pi()/180
                    msgbox('By is solving below the bounds (-45 to +45deg) for this
model');
                    return;
                end
                lhs = sin(By);
                rhs = (((-Uy*(1-slip))/Wy) + tan(beta_by)*(1+tan(beta_by)^2 - ((Uy^2
*(1-slip)^2)/(Wy^2)))^0.5)/(1+tan(beta_by)^2);
```

```matlab
                        error = lhs-rhs;
                end
        elseif error < 0
                while error < 0
                        By = By +0.001 / i;
                        if By > 45*pi()/180
                                msgbox('By is solving above the bounds (-45 to +45deg) for this
model');
                                return;
                        end
                        lhs = sin(By);
                        rhs = (((-Uy*(1-slip))/Wy) + tan(beta_by)*(1+tan(beta_by)^2 - ((Uy^2
*(1-slip)^2)/(Wy^2)))^0.5)/(1+tan(beta_by)^2);
                        error = lhs-rhs;
                end
        end
        i=i+1;
    end
    beta_y = By;
end
function omega = angvel(FID,Pr,Pi,Ti,r2,ExcelApp)
    i=1;
    gamma = ExcelApp.Run('cp_pt',FID,Pi,Ti)/ExcelApp.Run('cv_pt',FID,Pi,Ti);
    v       = ExcelApp.Run('vol_pt',FID,Pi,Ti);
    omega = 1;

    rhs = (1+((gamma-1)/gamma)*(1/(Pi*10^5*v))*(omega*r2)^2)^(gamma/(gamma-1));
    lhs = Pr;
    w_error = lhs-rhs;

    while abs(w_error) > 10^-5
        if w_error > 0
            while w_error > 0
                omega = omega + 1/i;
                rhs = (1+((gamma-
1)/gamma)*(1/(Pi*10^5*v))*(omega*r2)^2)^(gamma/(gamma-1));
                lhs = Pr;
                w_error = lhs-rhs;
            end
        else
            while w_error < 0
                omega = omega - 1/i;
                rhs = (1+((gamma-
1)/gamma)*(1/(Pi*10^5*v))*(omega*r2)^2)^(gamma/(gamma-1));
                lhs = Pr;
                w_error = lhs-rhs;
            end
        end
        i=i+1;
    end
end


end
```

References

[1] V. Ganni, P. Knudsen, D. Arenius and F. Casagrande, "Application of JLab 12GeV Helium Refrigeration System for the FRIB Accelerator at MSU," in *Advances in Cryogenic Engineering*, 2014.

[2] J. Fydrych, "Chapter 9: Cryogenic Transfer Lines," in *Cryostat Design*, Springer, 2016, pp. 241-272.

[3] A. Whitfield and N. Baines, Design of radial turbomachines, New York: John Wiley & Sons, 1990.

[4] A. Whitfield and N. C. Baines, "A General computer solution for radial and mixed flow turbomachine performance prediction," *International Journal of Mechanical Science,* vol. 18, pp. 179-184, 1976.

[5] D. Eckardt, "Detailed Flow Investigations Within a High Speed Centrifugal Compressor Impeller," *Journal of Fluids Engineering,* pp. 390-399, 1976.

[6] H. Oh, E. Yoon and M. Chung, "An Optimal Set of Loss Models for Performance Prediction of Centrifugal Comrpessors," *Proceedings of the Institution of Mechanical Engineers,* vol. 211, no. A, pp. 331-338, 1997.

[7] R. F. Barron, Cryogenic Heat Transfer, Abingdon, United Kingdom: Taylor and Francis, 1999.

[8] R. C. Pampreen, "A Blockage Model for Centrifugal Compressor Impellers," *Journal of Engineering for Power,* pp. 698-707, 1981.

[9] J. P. Johnston, R. M. Halleen and D. K. Lezius, "Effects of spanwise rotation on the structure of two-dimensional fully developed turbulent channel flow," *Journal of Fluid Mechanics,* pp. 533-557, 1972.

[10] V. Ganni, "Design of Optimal Helium Refrigeration and Liquefaction Systems," Tucson, 2009.