SiPhN v3.0 User Manual

Package description and usage guidelines

SAIKRISHNA NADELLA

November 2020

A word before you begin…

My Masters thesis work at Homi Bhabha National Institute, Mumbai included formulating improved models for Linear stability analysis (LSA) of single phase natural circulation loops (SPNCL) applicable for usual surface heat flux case as well as internal heat generation (distributed) case as encountered in liquid fuelled molten salt reactors. As part of my work, I derived some new formulations for LSA and carried out extensive validation studies. For performing these analyses, various codes have been written. All these codes are compiled into a software package called SiPhN (pronounced as 'siphon'). This package has capabilities to perform steady state, transient and stability (linear and non-linear) analysis of SPNCLs with surface heat flux and internal heat generation. The transient analysis code is able to perform very extensive studies on any polygonal shaped SPNCL with various combinations of local/distributed surface heating, internal heat generation and cooling.

After completion of my Thesis work, I felt that this set of codes may be useful for other researchers also in this field, directly or as a platform for further development. With my personal experience, I found that usage of a code developed by other person is hindered in most cases by the complexity in understanding how to run the code and the code itself. Making changes in the code requires detailed understanding of the code structure. So, I felt it will be useful to attach a GUI for this package for ease of usage, and at the same time keep all the source code well commented for understanding of code structure. I developed full GUI for all features of the code and integrated with SiPhN. A user manual is also developed for this code, for standalone learning of how to use this code. This user manual may be used along with my Masters Thesis for formulation, methods and nomenclature. Currently SiPhN reached version 3.0. Please see the 'Release Notes' for information on recent updates and bug fixes. I sincerely acknowledge the feedback from Shri Abhishek Kumar Srivastava, BARC which led to several improvements in user friendliness of the code.

The code and all associated files are made available for free upon a request mail at saikrishna.nadella9@gmail.com. Any improvements required or features to be added may kindly be intimated to me. Please report bugs, if any. I will try to incorporate them as soon as possible. The package can be redistributed freely as a whole, but not in bits and pieces, and the same may be kindly intimated to me by an email.

Kindly acknowledge this code, if you find it helpful in your research. This code can be cited as: *Saikrishna Nadella, "Development of computer code for stability analysis of molten salt natural circulation loop with and without internal heat generation", M. Tech Thesis, December 2017, Homi Bhabha National Institute, Mumbai.*

Saikrishna Nadella
M. Tech (2015-17), Mech. Engg., HBNI, Mumbai

Table of Contents

SiPhN v3.0

Introduction

SiPhN (pronounced as 'siphon') is a short form of **Si**ngle **Ph**ase **N**atural circulation. This is a software package developed by me as part of my Masters Thesis work on single phase natural circulation loops. With time, this package has seen several improvements. One major change in Version 3.0 is rewriting of entire code in Python3.

SiPhN v2.1 had following modules

1. **LSA-SHF**: Linear Stability Analysis – Surface Heat Flux
2. **LSA-IHG:** Linear Stability Analysis – Internal Heat Generation
3. **SSA – SHF:** Steady State Analysis – Surface Heat Flux
4. **SSA – IHG:** Steady State Analysis – Internal Heat Generation
5. **TA:** Transient Analysis

In current v3.0, the 2nd and 4th modules mentioned above are temporarily removed. For using these modules, the user is directed to v2.1.1 of SiPhN. Based on user requirements, these modules will also be translated to Python3 and included in future version of SiPhN.

In addition to the above modules, a utility called **TPPC** (Thermo Physical Property Calculator) is also included in the SiPhN package for calculating the thermo-physical properties of the working fluid and piping materials as per the user defined correlations.

The formulation behind all these modules can be found in various chapters of my Masters Thesis supplied along with. Detailed description and usage of these modules follow in subsequent sections.

The package structure

The package is supplied as a compressed folder. Unzip it. Read the 'Release notes.txt". You will find following files/folders.

- **src:** All the source code and related files are contained in this folder as follows
  - **.py files:** GUI control files as well as analysis source code
  - **.ui files:** GUI creation files
  - **.jpg:** image files
- **UserDefined:** code templates for user defined functions used in **SSA/LSA/TA** modules.
- **SiPhN.py:** The main program to be executed to start SiPhN.
- **ReleaseNotes.**txt: The detailed list of recent updates/changes in SiPhN.
- **Installation.**txt: Installation methodology of python interpreter.

Installation

Though SiPhN is run directly from source code, ***python3*** interpreter needs to be installed for running SiPhN. We recommend using python interpreter from *python.org.* The tested versions are python 3.6, 3.7 and 3.8. In addition to python interpreter, some dependencies also need to be installed. The best thing is that all dependencies are open source and are directly available on python.org as .whl files. The installation instruction in this method is given in Installation.txt.

Alternately, *Anaconda* package (anaconda.org) can also be used, which comes bundled with all dependencies.

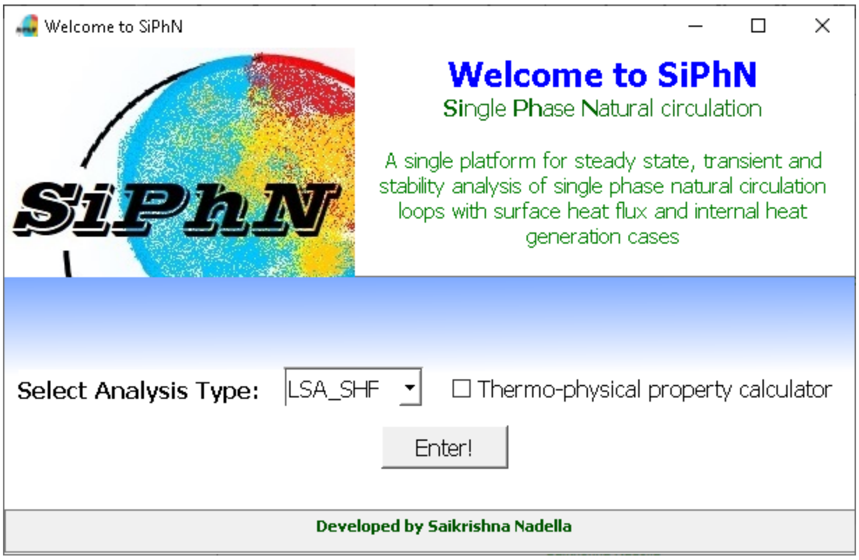
Running SiPhN

Assuming that *Python IDLE* is installed from *python.org,* along with all dependencies, extract the SiPhNv3.0_Release0.zip to suitable folder and open that folder. In the folder, open *CommandPrompt/PowerShell/Terminal* with python access. Run

python ./SiPhN.py

Then the front page of **SiPhN** package (Fig. 1) will be opened. Based on your installation, the script *SiPhN.py* can be run by simply double-clicking it. But, I recommend running the script through *commandPrompt,* although it does not make any difference in code performance.

In the drop down menu select the analysis type (module). Check the box against "Thermo physical property calculator" (optional) to open **TPPC** utility window along with the window of selected module. Then click the **Enter!** button to open selected module GUI. Refer respective sections of the modules for subsequent usage guidelines.



Fig, 1: Front page of SiPhN package

Let us talk about **TPPC** utility here. The window is shown in Fig. 2. User shall enter the temperature in °C in the text box and hit **Calculate.** Then based on the selection among **Fluid**

only/Wall only/Both the properties are calculated and displayed in **Properties** panel. The status message will be displayed in the **Status** box.

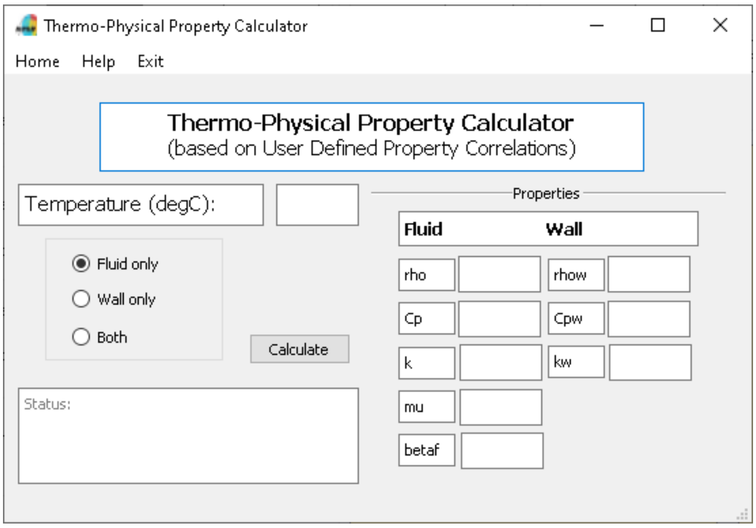


Fig. 2: TPPC module window

The calculation is performed based on user defined thermophysical properties to be entered in the script **/UserDefined/userTPPC.py.** Detailed guidelines for usage of this script is given within the script file. **Just open and read the comments in the beginning of the script!!!**

Mind the units of the properties being entered. SI units are used everywhere in the code, So, the same unit system, as mentioned in script header, should be followed for consistent results. There should not be any problem using any other system of units as far as consistency is maintained. However, no other unit system is verified yet. So, caution should be taken while using with different system of units.

LSA-SHF

Description

- Short form of Linear Stability Analysis – Surface Heat Flux
- This module is used to perform linear stability analysis of single phase rectangular natural circulation loops with surface heating in heater zone.
- LSA is done using two methods *viz.* 1) graphical method and 2) Nyquist plot. Both the methods and associated terminology and nomenclature are elaborated in Master Thesis supplied along with.
- This module has GUI (Fig. 3).
- **Based on user requests, additional options can be added.**

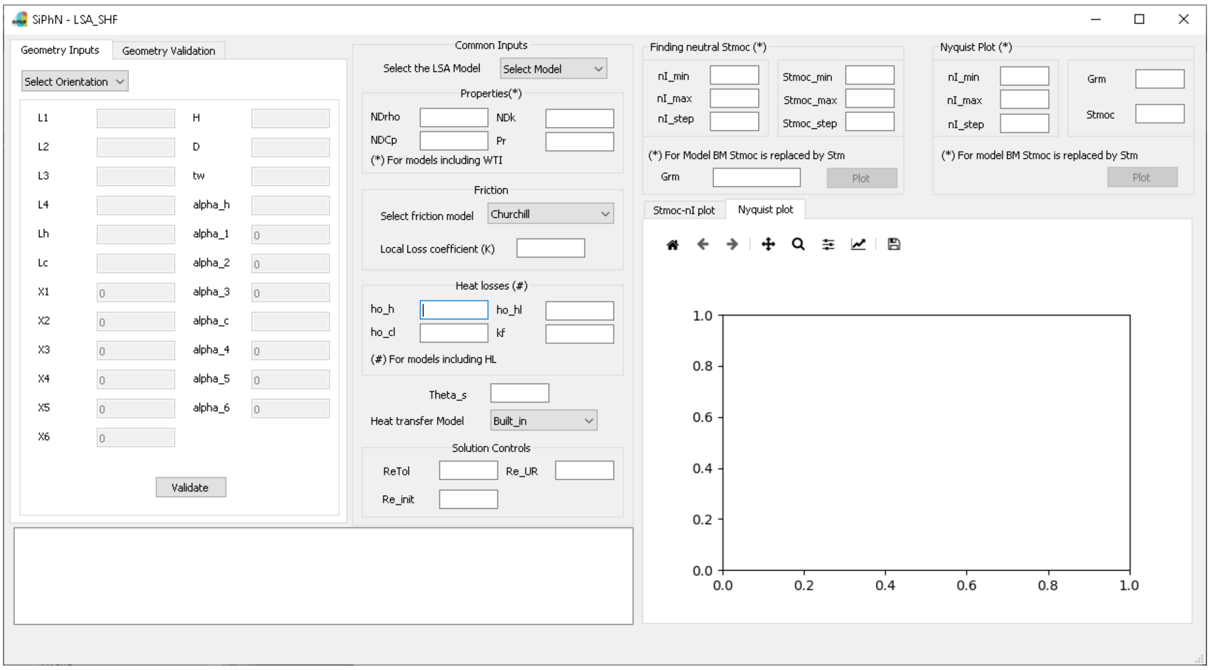


Fig. 3: Window of LSA-SHF module

Usage

Once the module window appears, following is the stepwise procedure to perform Linear Stability Analysis for SHF case.

1. In **Geometry Inputs** panel, Select the orientation.
2. Provide all geometry details (Refer Appendix A for details)
3. Click Validate
4. The loop is drawn in **Geometry Validation** panel and a status message is shown there. If loop is closed satisfactorily, then only the analysis is enabled in code. Note that every time a geometry input is changed, validation need to be performed before proceeding for analysis.
5. Select analysis Model (for details on these models, refer Masters Thesis)

6. Enter the dimensionless properties. Note that these will be used if only selected model includes **WTI** (wall thermal inertia) and/or **HL** (heat loss)
7. Select skin friction correlation.
    a. **Churchill's law:** built-in correlation of Churchill (Eq. 2.7 in Masters Thesis)
    b. **UserDefined:** correlation specified by user in **./UserDefined/user_f.py** will be used
8. Enter total local pressure loss coefficient of the NCL
9. If selected model includes **HL** (heat loss), enter the heat loss coefficients.
10. Select the heat transfer correlation.
    a. **Built-in:** built-in correlation given in Eq. 5.107 in Masters Thesis
    b. **User Defined:** correlation specified by user in **./UserDefined/user_Nu.py** will be used
11. Enter the solution controls
    a. **ReTol:** This is tolerance to be used in Reynolds number (iterative) calculation for estimating steady state conditions.
    b. **Re_UR:** This is the under relaxation factor to be used in Reynolds number (iterative) calculation for estimating steady state conditions. It is recommended to use 0.5 for low Reynolds numbers. If error like 'Max Iterations Over' appears, reduce it to as low as 0.1.
    c. **Re_init:** This is the initial guess of Reynolds number.
12. Enter **theta_s**, which is dimensionless secondary fluid temperature in cooler (refer: Masters Thesis Eq. 5.26)
13. If modified Stanton number is to be calculated for given Grashof number corresponding to stability boundary, use **Finiding Neutral Stmoc** panel.
    a. Enter the range of **nI** (imaginary part of root of characteristic equation) and **Stmoc**. Their step sizes can be selected based on your accuracy requirements.
    b. Enter the **Grm** required and hit **Plot**.
    c. Once the **Successful** message appears in the **TEXT_OUT** box on bottom left corner, check the plot in **Stmoc-nI plot** panel. Also note if any warning messages appear in **TEXT_OUT** box. See for intersection points between black (real part of characteristic function) and red (imaginary part of characteristic function) can be noted. The zooming tools and pan tool in the tool bar (above the plot) can be used for accurate determination of the value of **Stmoc.** User can perform changes in appearance of the plot using available tools in tool bar and can save the same for further use.
    d. If some Error messages appear, change the inputs appropriately to resolve the issues.
    e. Theory behind this method can be found in Masters Thesis in Section 4.3 (Algorithm 3).
14. If stability at given pair of **Grm** and **Stmoc** is sought, use **Nyquist Plot** panel.
    a. Enter the information similar to step 13 and hit **Plot.**

b. Count the number of clockwise encirclements of origin. Zooming and Panning tools in toolbar can be used for easiness.
c. Theory behind this method can be found in Masters Thesis in Section 4.3 (Algorithm 2).

15. You have learnt it now. All the best.

SSA – SHF

Description

- Short form of Steady State Analysis – Surface Heat Flux
- This module is used to determine steady state quantities like mass flow rate etc. of single phase rectangular natural circulation loops with surface heating in heater zone.
- SSA can be performed either with dimensional parameters or non-dimensional parameters. User can give input in either format and output is given in both the formats.
- The methods, associated terminology and nomenclature are elaborated in Master Thesis supplied along with.
- This module has GUI (Fig. 4).
- **Based on user requests, additional options can be added.**

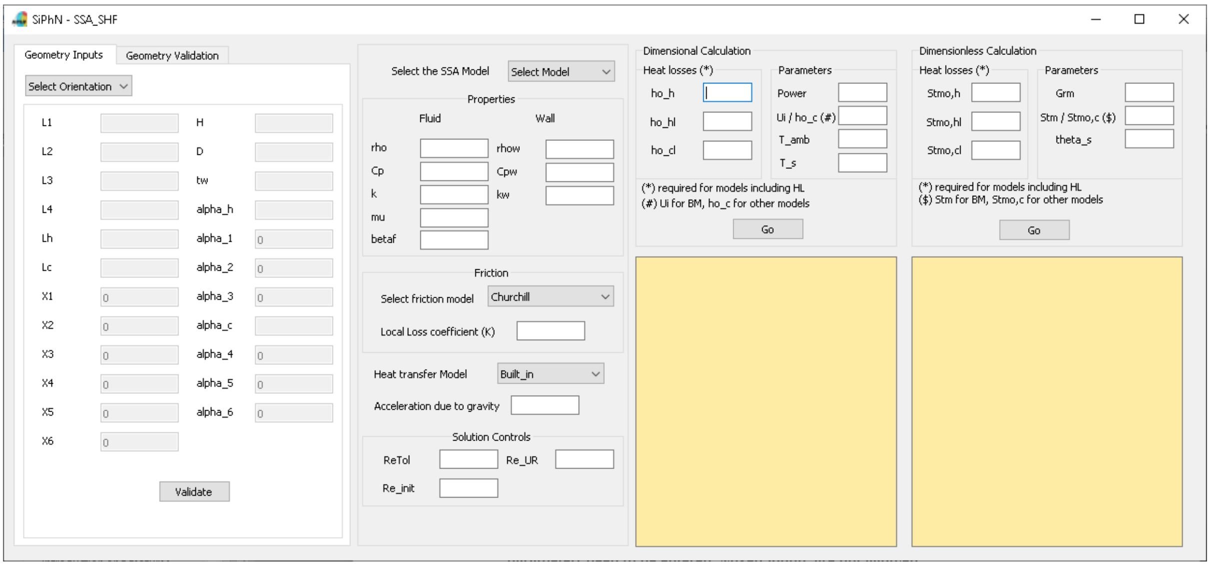


Fig. 4: Window of SSA-SHF module

Usage

All the inputs in this module window have same meaning as module LSA-SHF. If not referred yet, please refer usage section of LSA-SHF module and come back here

1. One additional quantity included here is ‘gravitational acceleration’. Default value is 9.81 m/s^2. Any other value can be given as input to simulate low/high gravity conditions/inclined loop conditions.
2. After giving inputs in the panels of first two columns, user need to decide whether he/she has inputs in dimensional coordinates or non-dimensional. Based on this, the parameters need to be entered. Mixed inputs are not allowed.
3. After entering inputs in chosen panel, hit **Go** button to get outputs in both dimensional and non-dimensional coordinates below the input panels.

4. Note that in model **BM**, the input for **ho_c/Stmo,c** is taken as overall HTC with reference to ID of the loop. For all other models, it is taken as outside HTC based on OD of the loop.
5. Formulation behind this module can be found in Chapter 5 of Masters thesis.
6. You learnt it now. It is that easy!
7. All the best

TA

Introduction and features

- Short form of Transient Analysis
- It is finite difference based solver for polygonal (any number of legs in any arbitrary orientation to make closed loop), single phase, uniform diameter natural circulation loops.
- This module has GUI (Fig. 5)
- Heating can be by surface heat flux or by internal heat generation or by both. Trace heating can be specified. Heat losses can be specified for piping.
- User can specify the thermo-physical property correlations based on temperature, skin friction correlations and heat transfer correlations in a very easy way. The templates are already included and are easy to make changes.
- Constant or temperature dependent thermo-physical properties for working fluid
- Optional feature for determination of approximate steady state mass flow rate for specifying as initial condition, **SS_INIT_.**
- Run time monitoring of mass flow rate within GUI using **Monitor Flow.**
- Update of 'Progress' online in GUI
- Reading inputs and writing outputs from within GUI, using **Load Table from file, Case label, dtSave** and **Output files** options.
- Feature to perform non-linear stability analysis by running the NCL with **SS_INIT_** flow rate for **tf** seconds before enabling the momentum equation. The inaccuracy in mass flow rate prediction acts as perturbation for NCL and the stability of the NCL can be determined by observing the flow rate evolution afterwards.
- Various transients like power raising, power set back, loss of heat sink and many more can be performed by defining two set of conditions in **Mesh inputs and Operating conditions of NCL** and specifying **tShift** for shift of operating conditions.
- The NCL can be divided into any number of sections for convenience, just add as many rows as wanted by clicking **Add Row.**
- The methods, associated terminology and nomenclature are elaborated in Master Thesis supplied along with.
- **Based on user requests, additional options can be added.**

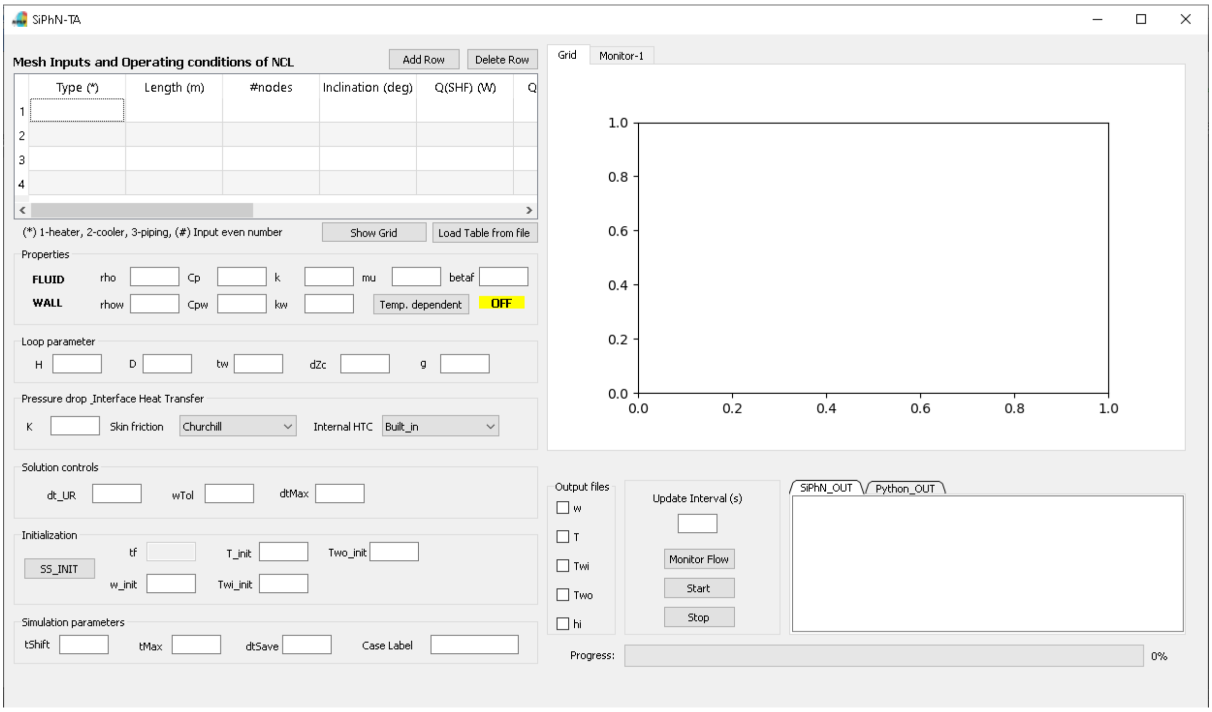


Fig. 5: Window of TA module

UI description and concepts

Many features available in this module are given in above section. Their usage is also indirectly included in this description by using the names of various panels/button/options as in GUI. Let us understand the UI and concepts.

1. Decide how many segments the NCL need to be divided
    a. A segment is not a control volume. It is a section of NCL which has same attributes throughout. The attributes are **inclination, Q (SHF), Q2 (SHF), Q (IHG), Q2 (IHG), ho, ho2** and **Tamb.** Here, **Q** denotes heat input in watts. SHF/IHG distinguishes the route of heat input as SurfaceHeatFlux/InternalHeatGeneration. **ho** is secondary side heat transfer coefficient at cooler or heat loss coefficient at other sections. **Tamb** is secondary side fluid temperature in cooler and ambient temperature for other sections. **Q2** and **ho2** corresponds to 2nd operating condition.
    b. Add as many segments as required by clicking **Add Row** button to create the same number of rows in **Mesh inputs and operating conditions of NCL** table.
    c. If you want to use the same table which you used for an earlier case, just load the **CSV** file from that case folder using **Load Table from file** button. Whenever an analysis is done, the **Mesh inputs and operating conditions of NCL** table is, by default, saved into case folder as CSV file.
2. Type of segment: Distinguish the segment whether it is heater, cooler or connecting pipe
3. No. of nodes: how many number of grid points need to be put in the segment? Choose even number! We forced this for utilizing accuracy of Simpson 1/3rd rule for integration.
4. Inclination:

a. This is inclination of the segment with horizontal (typical positive x axis) in the direction of flow. What do I mean?
b. First of all, assume a direction of flow in the NCL and note it down. For each segment, determine the angle between the flow direction in the segment and standard +ve x axis (horizontal). Repeat for all segments.
c. If any mistake is done in this step, an error will be displayed in **SiPhN_OUT** box stating that the NCL is not closed. It is recommended to check the mesh after filling the **Mesh inputs and operating conditions of NCL** table by clicking **Show Grid** button.

5. Properties panel:
   a. Do you want thermophysical properties of working fluid to be calculated based on temperature?
   b. If **Yes,** just click the button **Temp. dependent** and ensure that it indicates **ON**. Go to **UserDefined** folder and enter the correlations for properties in **userTPPC.py** file. Refer Introduction section on **TPPC** utility for guidelines to modify this script.
   c. If **No,** just make sure that **Temp. dependent** button indicates **OFF** adjacent to it. Enter the values of all the properties in respective boxes in the panel.
6. Loop parameters panel:
   a. D = ID of loop (taken constant everywhere)
   b. tw = wall thickness of piping (taken constant everywhere)
   c. H = height of the loop
   d. dZc = centreline elevation difference between heater and cooler. This concept may be vague for certain cases where distributed heating/cooling takes place. This parameter is used in **SS_INIT_** utility. So, caution is required on **dZc** value while using **SS_INIT_** utility.
7. Pressure drop and interface heat transfer panel:
   a. K = total local pressure loss coefficient in NCL
   b. Skin friction
      i. **Churchill's law:** built-in correlation of Churchill (Eq. 2.7 in Masters Thesis)
      ii. **UserDefined:** Enter required correlation in standard format in **UserDefined/user_f.py** script file. Follow the instructions thereof.
   c. Heat transfer
      i. **Built-in**: built-in correlation given in Eq. 5.107 in Masters Thesis
      ii. **UserDefined**: Enter required correlation in standard format in **UserDefined/user_Nu.py** script file. Various local parameters are provided for calculation of Nusselt number. The built-in function script (**src/Nu_main.py**) may be referred for guidance. Follow the instructions thereof.
8. Solution controls:
   a. The solver in TA modules uses explicit method for energy equations. Limiting time step is calculated at every time step. **dt_UR** is relaxation factor to be applied on limiting time step to decide the time step to be used.

   b. **wTol** is the tolerance to be used in Newton-Raphson based implicit solver of momentum equation to terminate the iterations.
   c. **dtMax** is maximum allowed time step.
9. Initialization panel:
   a. **SS_INIT_** button to be pressed if you want to use the flow rate calculated by Vijayan's correlation (2007) to be used as initial flow rate at t=0.
   b. When **SS_INIT_** is enables, **tf** becomes editable. **tf** is number of seconds from t=0, the momentum equation of natural circulation should **NOT** be solved. This is used to mimic forced circulation.
   c. **T_init, Twi_init** and **Two_init** are initial guess for temperature of working fluid, wall ID and wall OD respectively. Uniform profiles are taken for total loop.
10. Simulation parameters panel:
    a. **tShift** is number of seconds from t=0, after which the operating condition changes from **Q(SHF), Q(IHG) and ho** to **Q2(SHF), Q2(IHG) and ho2.**
    b. **tMax** is the time up to which the simulation need to be performed.
    c. **dtSave** is time interval (in terms of simulated time) in which the solution data to be saved to **outputs/TA/<Case Label>** folder.
    d. **Case Label** is the name of the case folder to be saved in **outputs/TA/**
11. Output files panel
    a. Various solution parameters can be saved to file output. Select the parameters of interest.
12. Start the simulation using **Start** button.
13. Flow monitoring
    a. Enter the plot update interval in terms of simulated time.
    b. The **Monitor Flow** button can be toggled ON/OFF at any time during simulation.
14. If required, the simulation can be stopped by **Stop** button.
15. SiPhN_OUT
    a. The place where the run status of the solver is displayed. Once solution is over, it says "simulation completed successfully".

Usage

Now let us see how to do various type of analysis in this module.

Steady state analysis

Though there are separate module (SSA-SHF) for steady state analysis, they are limited by the assumptions taken in analytical derivation. Attaining analytical solution for complicated cases like combined SHF and IHG etc. is prohibitive. They are also limited in implementation to a rectangular NCL with 4 legs. So, to handle the scenarios which cannot be analysed by SSA-SHF module, TA module can be used.

1. Fill all the input panels as described in previous section.
2. The tricks to do steady state simulations are
   a. Leave Q2(SHF), Q2(IHG) and ho2. Give **tShift** greater than **tMax.**

    b. Give Q2(SHF)=Q(SHF), Q2(IHG)=Q(IHG) and ho2=ho. Then **tShift** can be any value.
3. You may use **SS_INIT_** for getting steady state faster in many cases.
4. **EndData.dat** file in **outputs/TA/<Case Label>** folder contains the parameters and profiles of final time step. If steady state is reached, then this serves as final data.

Start-up transient

Provide negligible value for mass flow rate at **w_init** (avoiding giving Zero for God's sake) and give **tShift** greater than **tMax.** Do the simulation. Remaining procedure is same as in previous section (UI description and concepts).

Note: In perfect HHHC orientation, start-up transient cannot be captured by generic 1D formulation. Caution is to be taken for start-up transients in HHHC orientation. Some attempts have been taken in literature (Naveen Kumar et al.) to avoid this problem. User may refer the literature for details.

Power raise/set-back transients

1. Define first operating powers in Q(SHF) and Q(IHG) columns and second operating powers in Q2(SHF) and Q2(IHG) columns.
2. Define at what time **tShift**, the step change has to occur.
3. Do simulation.

Loss of heat sink transient

1. Define the secondary side heat transfer coefficient of cooler in ho (W/m^2K) column.
2. Define expected secondary side heat transfer coefficient of cooler after loss of heat sink in ho2 (W/m^2K) column. Most conservative value Zero. Based on conduction/natural convection/radiation on secondary side of cooler, appropriate HTC can be given.
3. Define at what time **tShift**, the loss of heat sink has to occur.
4. Do simulation.

Stability analysis

1. Small perturbation
    a. Using **SS_INIT_** and **tf** utilities (with **tf** sufficiently large to establish steady temperature profile) a good initial condition can be set up for natural circulation loop stability analysis. The error in the calculation of mass flow rate by **SS_INIT_** utility itself, acts as initial perturbation to flow. Subsequent development of flow reveals the stability of NCL at that operating condition.
    b. Use **tf** < **tMax** and **tShift** > **tMax**. Define the operating condition in Q(SHF), Q(IHG) and ho columns.
    c. Do simulation.
    d. The advantage of using these utilities is the ability to provide small perturbations which in turn help in validating Linear Stability Analysis codes.
2. Large perturbation: Large perturbations can be inherently given by various ways.

   a. Start-up transient to the given operating condition
   b. Power step change transient from a known stable state
   c. It is always recommended to use **SS_INIT_** and **tf** utilities (as in point 1a above) by taking forced flow rate at required % off from calculated NC flow rate. (this needs temporary change in source code, **TA_Main**). It is recommended due to known level of perturbation in this method compared to the former two methods.
3. Logarithmic decrement or decay ratio may be calculated separately by processing the output file of mass flow rate (**w**).

There can be multiple ways of getting the type of analysis done from TA module.

Appendix A

Giving geometry inputs in LSA-SHF and SSA-SHF modules

The formulation in LSA-SHF and SSA-SHF modules to handle multiple orientations of NCL is based on a generalized loop as shown in Fig. A1. It has 8 legs, with one leg each for heater and cooler. The nomenclature for different parts of the loop is given. The logic behind the usage of such a generalized geometry for multiple orientations is to make lengths of unnecessary legs to zero. This is done automatically for the convenience of user. However, if user wants to select some orientation which is not listed there, he/she can select **FreeForm** from **Select Orientation** drop down menu and enter the details.

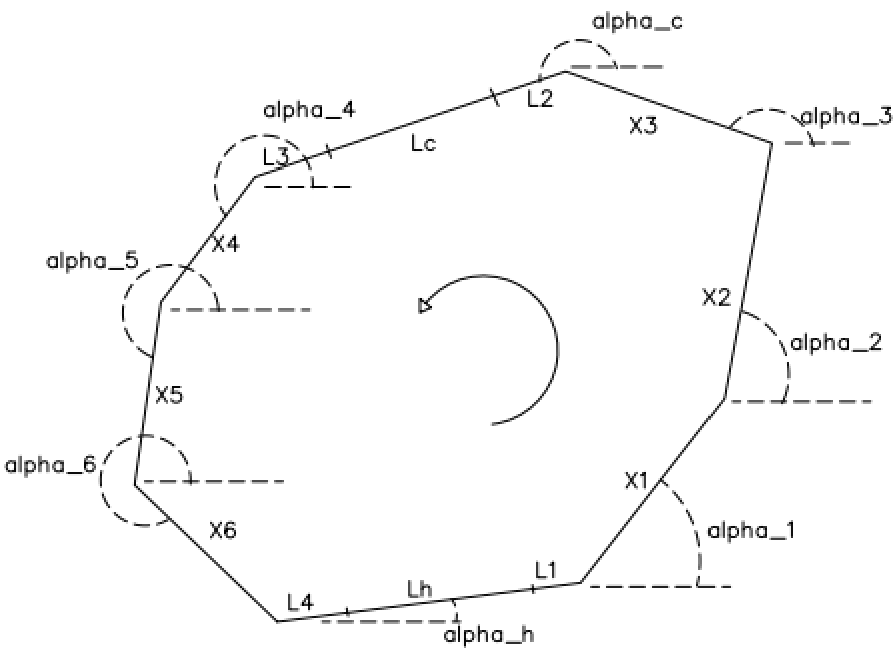


Fig. A1: Generalized geometry of NCL behind the formulation implemented in LSA-SHF and SSA-SHF modules.

The orientations HHHC, HHVC, VHHC and VHVC are typically mentioned everywhere in literature. Masters thesis can be referred for the same. In VHVC2 orientation, cooler is present in the same leg as heater. The subscripts CW/CCW denote clockwise/counter-clockwise flow respectively.