



National Technical University of Athens
School of Mechanical Engineering
Thermal Engineering Section
Laboratory of Applied Thermodynamics

Thesis Title:

**Determining the optimal phase difference of a Superfluid
Stirling Refrigerator using CFD software.**

by

TSAPAS MICHAEL

Supervisor

Prof. Rogdakis Emmanouil

Athens, February 2020

Στους γονείς μου, Νίκο και Έλενα

Abstract

In this thesis, a system consisting of two combined cryocoolers that use ^3He as a working medium is simulated with the use of Computational Fluid Dynamics (CFD) and ANSYS Fluent software. The cryocoolers operate with frequency of 1Hz and temperature below 1K. Each of them employs a recuperator and the two recuperators exchange heat with each other, combining thus the two machines into one system referred to as Superfluid Stirling Refrigerator (SSR). These machines will operate with a phase difference. The phase difference is determined by the mass flow, hot or cold, in the recuperator. The aim is to determine the optimal phase difference that provides the better efficiency when is applied to the machine. Afterwards, we will compare it with the 1D and Schmidt models. In order to determine the optimum phase difference, we have to initially determine the operating range. We will achieve this by setting multiple cases and gradually increase the phase difference that the machine operates in each case. As it will be detailly explained later, the SSR cooling capabilities are eliminated and starts operating out of this range when the parallel flow towards the system cold boxes is appeared. For every case that the machine operates inside the range that provides cooling power, we will also calculate the Coefficient of Performance (COP). Finally, using a polynomial regression we will correlate the COP with the phase difference resulting us to conclude the optimum and worst design points.

Keywords:

«Superfluid Stirling Refrigerator, SSR, Cryocooler, recuperator, Kapton, ^3He , ^4He , Superfluid, Optimum Phase Difference, Coefficient of Performance, COP, ANSYS-Fluent, CFD»

Περίληψη

Μια μηχανή Superfluid Stirling Refrigerator γνωστή και ως SSR αποτελείται από δύο απλούστερες ψυκτικές μηχανές οι οποίες χρησιμοποιούν ως εργαζόμενο μέσο ^3He . Οι δύο αυτές μηχανές, αποτελούνται από 2 έμβολα η καθεμία τα οποία συμπιέζουν και αποτονώνουν ισοθερμοκρασιακά το εργαζόμενο μέσο και συνεργάζονται μεταξύ τους εναλλάσσοντας θερμότητα με την βοήθεια ενός εναλλάκτη. Για να μπορέσει η SSR να είναι αποδοτική και να παρέχει ψυκτική ισχύ, θα πρέπει οι δύο μηχανές να λειτουργούν μεταξύ του με διαφορά φάσης την οποία την μετράμε σε μοίρες στροφάλου. Σκοπός της διπλωματικής αυτής, είναι να προσομοιώσουμε την SSR με την χρήση CFD προγράμματος για να αποφανθούμε αρχικά ποιο είναι το εύρος λειτουργίας σε μοίρες στροφάλου και κατά δεύτερον το βέλτιστο σημείο σχεδιασμού. Για να το πετύχουμε αυτό, με την χρήση του υπολογιστικού προγράμματος ANSYS Fluent θα προσομοιώσουμε διάφορες υποθέσεις στις οποίες κάθε φορά θα αυξάνουμε σταδιακά τις μοίρες εκτός φάσης που λειτουργούν οι δύο μηχανές και θα μελετάμε την συμπεριφορά της SSR. Όπως θα αναλυθεί και εκτενέστερα αργότερα, γνωρίζουμε ότι η μηχανή παύει να έχει ψυκτικές ικανότητες όταν παρατηρείται μέσα στον εναλλάκτη ομορροή προς τα «hot boxes» των δύο μηχανών. Σε κάθε υπόθεση που η SSR παρέχει ψυκτική ισχύ θα υπολογίζουμε τον βαθμό απόδοσης COP (Coefficient of Performance) και χρησιμοποιώντας μια πολυωνυμική προσέγγιση θα τον συσχετίσουμε με τις μοίρες στροφάλου. Τέλος, τα αποτελέσματα θα συγκριθούν με την μελέτη 1D που έχει ήδη διεκπεραιωθεί καθώς επίσης και με το θεωρητικό μοντέλο του Schmidt.

Λέξεις κλειδιά:

«Μηχανή Στέρλινγκ, Εναλλάκτης, Κάπτον, ^3He , ^4He , Υπερρευστό, Βέλτιστη διαφορά φάσης, Βαθμός απόδοσης ψυκτικής μηχανής, ANSYS-Fluent»

Ευχαριστίες

Καταρχάς, στα πλαίσια της συνεργασίας μας θα ήθελα να ευχαριστήσω τον καθηγητή μου, *Εμμανουήλ Ρογδάκη*, που με εμπιστεύτηκε να εκπονήσω την διπλωματική εργασία μου υπό την επίβλεψή του καθώς επίσης και για την αμέριστη κατανόηση του. Θα ήθελα επίσης να ευχαριστήσω τους υποψήφιους διδάκτορες *Ραφαήλ Δομένικο* και *Παναγιώτη Μπιτσικά*, για την συνεχή καθοδήγηση και βοήθεια που μου παρείχαν, βοηθώντας με να επιλύσω οποιοδήποτε κώλυμα προέκυψε κατά την διάρκεια διερεύνησης και συγγραφής της παρούσα διπλωματικής.

Έπειτα, με την περάτωση των σπουδών μου, οφείλω και ένα μεγάλο ευχαριστώ σε όλους όσους βρέθηκαν κοντά μου, για μεγαλύτερο ή μικρότερο χρονικό διάστημα, βοηθώντας και στηρίζοντάς με στην προσπάθειά μου.

Ιδιαίτερα, θα ήθελα να ευχαριστήσω,

Τον πατέρα μου *Νίκο*, ο οποίος όντας και πρότυπό μου, υπήρξε ο θεμελιωτής του πάθους μου για την μηχανολογία, μεταλαμπαδεύοντάς μου απλόχερα τις πολύτιμες γνώσεις και εμπειρίες του. Που χωρίς αυτόν θα ήταν αδύνατο να σπουδάσω, προσφέροντάς μου όλα αυτά τα χρόνια τα μέγιστα.

Την μητέρα μου *Έλενα*, για την οποία ό,τι και να πω θα είναι λίγο, αφού βρίσκεται πάντοτε στο πλάι μου, αρωγός σε κάθε μου προσπάθεια θυσιάζοντας και υπερβάλλοντας εαυτόν, καθώς επίσης υπήρξε και το μεγαλύτερο στήριγμά μου σε πολύ δύσκολες στιγμές. Μαμά ένα ευχαριστώ είναι λίγο.

Τα αδέρφια μου, *Παναγιώτη*, *Αποστόλη* και *Πανωραία*, οι οποίοι στηρίζοντάς με καθημερινά συνέβαλαν στο να διαμορφωθώ ως ο άνθρωπος που είμαι σήμερα.

Την γιαγιά μου, *Μαίρη*, η οποία συνεισφέρει πάντοτε τα μέγιστα.

Τους συμφοιτητές μου, *Νίκο Βασιλάκο* και *Δημήτρη Ντάγκρα* για όλες τις στιγμές που περάσαμε μαζί καθ' όλη την ακαδημαϊκή μας πορεία.

Τον αδελφικό μου φίλο, *Κωνσταντίνο Φέτερη*, με τον οποίο έχουμε ζήσει τόσα.

Και λοιπούς ανθρώπους οι οποίοι με στηρίζαν ο καθένας με τον τρόπο του, τον θείο μου Σπύρο Πετράτο, τον Αλέξανδρο Τσολοβίκο, την Ζωή Μακρή και την Χριστίνα Δημητριάδη.

Table of Contents

Chapter 1. Introduction	21
1.1 What is a SSR and why they are usfull?	22
1.2 Stirling Cycle – How does it work?	22
1.3 History of SSR.	25
Chapter 2. Properties of $^3\text{He} - ^4\text{He}$ mixture.	29
2.1 Liquid $^3\text{He} - ^4\text{He}$	29
2.2 Osmotic Pressure.	31
2.3 Kapton.	32
Chapter 3. Ideal Mathematical Model	33
3.1 Schmidt Model of SSR – The ideal model.	33
3.2 Recuperator design – Practical model.	36
3.2.1 Basic Principles.	36
3.2.2 Cycle Explanation.	37
Chapter 4. Determining the operating range with ANSYS FLUENT.	45
4.1 Geometry.	45
4.1.1 Generate Geometry.	48
4.1.2 Import Selected Geometry.	52
4.2 Meshing techniques.	54
4.2.1 Cylinder Meshing.	56
4.2.2 Recuperator Meshing.	58
4.3 Properties of the materials.	60

4.4 Boundaries.	63
4.4.1 Mesh Interfaces.	63
4.4.2 Wall Conditions.	68
4.5 User - Defined Functions (UDF)	72
4.5.1 What is a UDF?.....	72
4.5.2 Defining UDF with Macros	73
4.5.3 Including the udf.h Header File in the Source File.	74
4.5.4 Interpreting and Compiling UDFs	75
4.5.5 Compiled UDFs	75
4.5.6 Interpreted UDFs	76
4.5.7 Differences Between Interpreted and Compiled UDFs.	77
4.5.8 Creating the UDF	78
4.5.9 Import and Compile the UDF	82
4.6 Dynamic Mesh	83
4.7 Setting up the Solution Parameters.	86
4.7.1 Solver Type – Energy equations.	86
4.7.2 Initialization process.	89
4.7.3 Time step.....	92
4.8 Data Management.	93
Chapter 5. Data Analysis	97
5.1 Operating 120° out of phase.....	99
5.2 Operating 140° out of phase.....	101
5.3 Operating 152° out of phase.....	102

5.4 Operating 160° out of phase.....	104
5.6 Operating 170° out of phase.....	107
5.7 Operating 180° out of phase.....	110
5.8 Operating 200° out of phase.....	113
5.9 Operating 204° out of phase.....	116
Chapter 6. Polynomial Regression – Results Comparison	117
6.1 Polynomial Regression – COP	117
6.2 Comparison with the 1D model.	121
6.3 Comparison with the Schmidt model.....	122
Conclusion	123
Bibliography/ References.....	127

List of Figures

Figure 1. The basic Stirling Cycle.	24
Figure 2. Illustration of the ^4He and $^3\text{He} - ^4\text{He}$ superfluidity.	30
Figure 3. Osmotic pressure	31
Figure 4. Components of a recuperator design SSR.	37
Figure 5. Pistons' motions proportional to time in crank shaft angles for one period.....	38
Figure 6. Pistons' motions proportional to time, operating at the same phase.	44
Figure 7. Components of a SSR.....	46
Figure 8. Dimensions of a SSR.....	47
Figure 9. Positions and Movements of all the pistons.	51
Figure 10. Generated Geometry.....	52
Figure 12. Geometry – Tree Outline view	53
Figure 13. Geometry imported directly from schematic menu.....	54
Figure 14. Using the Face-sizing method.	56
Figure 15. Using the Sweep Method.....	57
Figure 16. Number of divisions in each side.	58
Figure 17. Number of divisions in every direction.	59
Figure 18. Editing materials from ANSYS Fluent Tree menu.	61
Figure 19. Editing fluid material in ANSYS Fluent.	62
Figure 20. Editing Solid material in ANSYS Fluent.	62
Figure 21. Mesh Contact (Compressor – Cooler).	64
Figure 22. Mesh Contact (Cooler – Connection Tube).....	65

Figure 23. Mesh Contact (Connection Tube – Recuperator).....	66
Figure 24. Mesh Contact between the two Recuperators.	67
Figure 25. Coupled Wall.....	67
Figure 26. Reservoir wall’s momentum properties.....	68
Figure 27. Hot reservoir wall’s Thermal properties.....	69
Figure 28. Cold reservoir wall’s Thermal properties.....	71
Figure 28. User-Defined Function for Compressor – SR1.	78
Figure 29. User-Defined Function for Expander – SR1.	79
Figure 30. User-Defined Function for Compressor – SR2.	80
Figure 31. User-Defined Function for Expander – SR2.	81
Figure 32. Compiling the UDFs.....	82
Figure 33. Message in the dialog box after the UDFs has successfully been compiled. ...	83
Figure 34. Setting Dynamic Mesh for the Piston.....	84
Figure 35. Setting Dynamic Mesh for the Cylinders.	85
Figure 36. Pressure Based solver with SIMPLE method.....	87
Figure 37. Pressure Based solver with COUPLED method.	87
Figure 38. Difference between the initial and operating pressure – C1.....	89
Figure 39. Difference between the initial and operating pressure – C2.....	90
Figure 40. Difference between the initial and operating pressure – E1.....	90
Figure 41. Difference between the initial and operating pressure – E2.....	91
Figure 42. Difference between the initial and operating volume.....	91
Figure 43. Selecting the appropriate time step.....	92
Figure 44. Pressure Data.	93

Figure 45. Volume Data.....	94
Figure 46. Heat Flux Data.....	94
Figure 47. Mass Flow Rate.	95
Figure 48. Flows in both regenerators operating 180° out of phase.	98
Figure 49. Mass Flows when operating 120° out of phase.	99
Figure 50. Velocities and Displacements when operating 120° out of phase.	100
Figure 51. Mass Flows when operating 140° out of phase.	101
Figure 52. Velocities and Displacements when operating 140° out of phase.	102
Figure 53. Mass Flows when operating 152° out of phase.	103
Figure 54. Velocities and Displacements when operating 152° out of phase.	104
Figure 55. Mass Flows when operating 160° out of phase.	105
Figure 56. Velocities and Displacements when operating 160° out of phase.	106
Figure 57. Pressure changes when operating 160° out of phase.	106
Figure 58. Volume changes when operating 160° out of phase.....	107
Figure 59. Mass Flows when operating 170° out of phase.	108
Figure 60. Velocities and Displacements when operating 170° out of phase.	108
Figure 61. Pressure changes when operating 170° out of phase.....	109
Figure 62. Volume changes when operating 170° out of phase.....	109
Figure 63. Mass Flows when operating 180° out of phase.	111
Figure 64. Pressure changes when operating 180° out of phase.	111
Figure 65. Volume changes when operating 180° out of phase.....	112
Figure 66. Velocities and Displacements when operating 180° out of phase.	112
Figure 67. Mass Flows when operating 200° out of phase.	114

Figure 68. Pressure changes when operating 200° out of phase.	114
Figure 69. Volume changes when operating 200° out of phase.....	115
Figure 70. Mass Flows when operating 204° out of phase.	116
Figure 71. Correlation between COP and Phase Difference.....	118
Figure 72. Correlation between phase difference and COP.....	120
Figure 73. Correlation between COP and Phase Difference – 1D.....	121

Chapter 1.

Introduction

The purpose of this thesis is initially the understanding of the basic principles of a Superfluid Stirling Refrigerator (SSR), secondly to explain why the recuperative Stirling design is preferred and finally to simulate the SSR in ANSYS FLUENT and search for the optimal design point.

As Ashok B. Patel detaily explained in his PhD thesis in 1992, SSR is a Stirling cycle Refrigerator which uses a liquid $^3\text{He} - ^4\text{He}$ mixture as the working fluid. It belongs in the class of refrigerators which use the standard ideal gas refrigeration cycles to provide cooling below 2 K. The first SSR was demonstrated in 1990 by Kotsubo and Swift and by 1995 Brisson and Swift developed an experimental prototype capable of reaching a low temperature of 296 mK.

The Stirling refrigerators can be designed with a standard regenerator or with a recuperator. Due to the lack of knowledge of the regenerative properties at these temps the recuperative design is used. There are fundamental differences in the two designs due to a mass flow imbalance within the recuperator. This imbalance is defined by the different phase that the two SSRs are compressing and expanding. So, in the recuperator design, the recuperator is actually a coupled wall with very high thermal conductivity and is achieved by having two SSR operating simultaneously with 180° out of phase each other. The compressor piston-1 will compress and have a difference phase from the compressor piston-2. Every number is referring in which SSR they belong. The working fluid of each SSR, after being compressed, it passes through a thin wall before reaching the expander piston. So, the thermal flow can be either hot or cold resulting the two walls to exchange heat considering that there will be a thermal deposit in wall-1 and a thermal withdraw in the wall-2. The working cycle will be explained with more details in the next chapters.

In general, the cooling powers of a SSR are large enough for a number of applications, and it is currently the only closed cycle sub-Kelvin refrigerator capable of

such low temperatures. Also, the cooling powers are enough to allow a SSR to be used as the upper stage of other closed cycle and may be capable of achieving milli-Kelvin temperatures. [1], [2]

1.1 What is a SSR and why they are usfull?

The SSR is a Superfluid Stirling Refrigerator which uses a liquid $^3\text{He} - ^4\text{He}$ mixture as the working medium. Refrigeration below 1 K is required in many field of research including particle physics, surface chemistry, material science and condensed matter physics. Despite the necessity the current non-SSR sub-Kelvin refrigerators have features which preclude their use by most organizations demanding the development of a new design that will better fill the refrigeration needs in these field and allow more researchers access to temperatures below 1 K.

When we refer to non-SSR refrigerators we mean the dilution refrigerator, the ^3He evaporation refrigerator and the adiabatic demagnetization of a paramagnetic salt. SSR has many advantages compared to these with the most important that is more efficient and less expensive because it does not require the expensive sealed pumps and extensive pumping lines of both dilution and ^3He and it does not generate high magnetic field of a demagnetization stage. Also, it delivers microwatts of cooling power for tens of watts of drive power compared to the alternative ways that use kilowatts of drive power to deliver the same cooling power. Finally, the SSR does not require modification of its basic cycle to works in a zero-gravity environment so it's the ideal for sub-Kelvin applications in space.

The main disadvantage of the SSR is that when its technology will be fully developed it will probably achieve an ultimate low temperature between 75 to 100 mK. This lower limit is due to dissipation from the SSR's moving parts at its cold end. [1], [2]

1.2 Stirling Cycle – How does it work?

The basic superfluid Stirling cycle can be understood using Fig. 1, which depicts the four steps in a Stirling Cycle. The liquid $^3\text{He} - ^4\text{He}$ mixture is held between a cold piston and a hot piston, each bypassed by superleaks. The hot piston is in thermal contact

with a high temperature reservoir at T_h while the cold is in thermal contact with a cold temperature reservoir at T_c . The volume behind of each piston is filled with superfluid ^4He . The piston cylinders are connected by a regenerator which can thought of as an array of high heat capacity tubes with poor thermal conductivity in the longitudinal direction and good thermal conductivity in the transverse direction. The regenerator is design so there is no heat interaction between the hot and the cold piston. More specific, in the ideal regenerator the heat capacity of the tubes is infinite, the thermal contact between the fluid and the tubes is perfect and there is no thermal conductance in the longitudinal direction.

The cycle begins with the $^3\text{He} - ^4\text{He}$ mixture is entirely in the compressor piston cylinder. At this point, the mixture is at the same temperature as the high temperature Hot reservoir. The normal ^3He component is isothermally compressed by the hot piston (Fig. 1a). while the superfluid ^4He component of the mixture flows freely through the superleak to the backside of the compressor piston. Because the ^3He component of the mixture behave as an ideal gas in an inert “atmosphere” of ^4He , heat is rejected to the high temperature reservoir. Following this, both pistons are slowly moving in tandem to isochorically displace the ^3He through the regenerator to the cold expander piston cylinder (Fig. 1b). Ideally there is a linear temperature gradient in the regenerator and because of the large heat capacity of the regenerator matrix, the ^3He is reversely cooled from the compressor piston temperature to the temperature of the cold thermal reservoir. The ^3He is now isothermally expanded, absorbing heat from the low temperature reservoir (Fig. 1c). Finally, the expanded ^3He is displaced back through the regenerator, where it reversibly reclaims the heat deposited there earlier to the compressor piston cylinder (Fig. 1d). The refrigerator is now back in the original state and the net effect has been a heat transfer from the cold reservoir to the hot reservoir.

Basic Stirling cycle

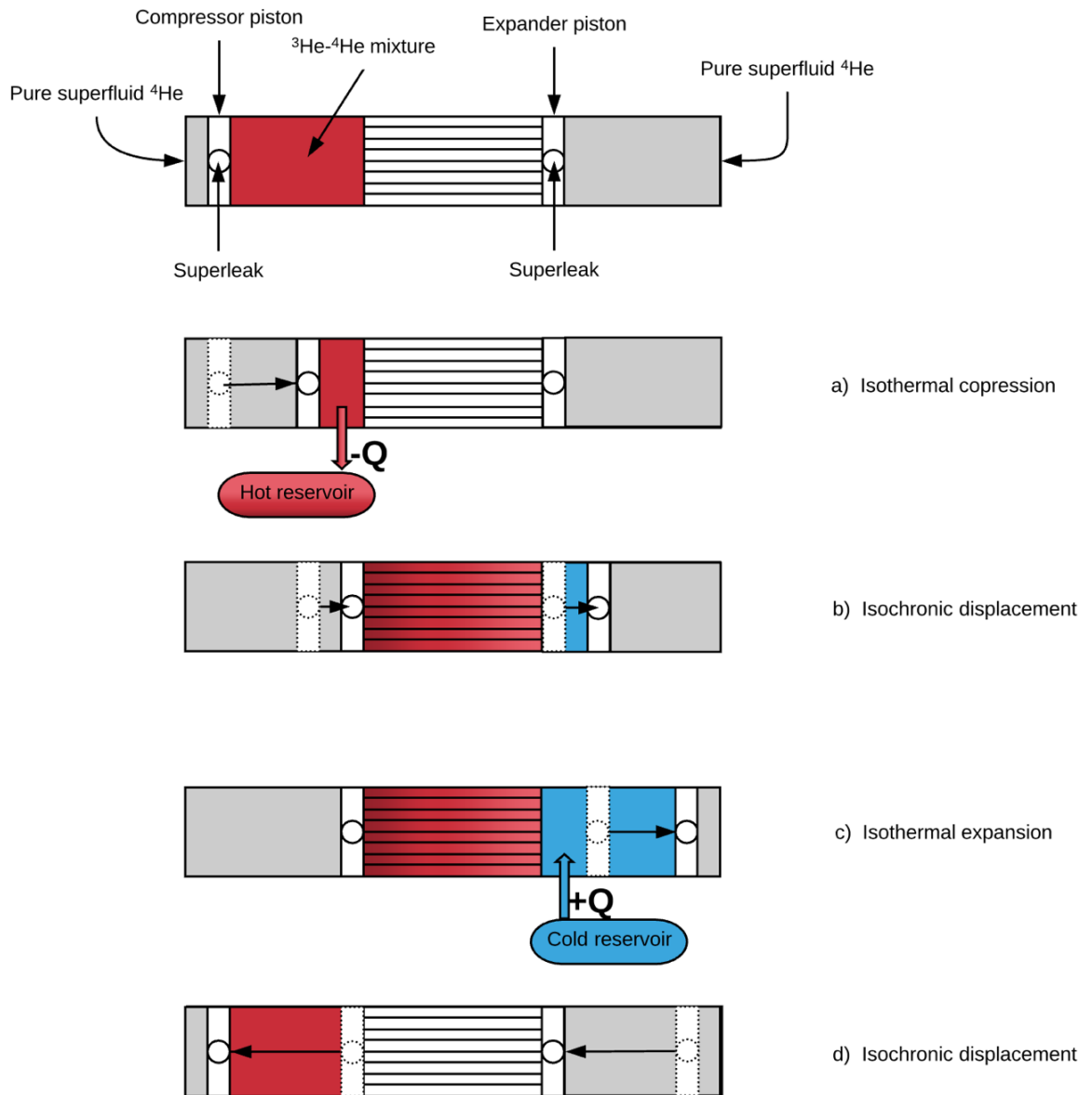


Figure 1. The basic Stirling Cycle.

The basic motions of the pistons and the states of mixture ${}^3\text{He} - {}^4\text{He}$. [1], [2]

But as it was mentioned earlier, due to a lack of proper materials properties at sub-Kelvin temperatures, the principal design configuration of SSR is that with a recuperator and not with a regenerator. So, a more practical SSR is shown in Fig. 2. The refrigerator is actually two Stirling refrigerators operating 180° out of phase with each other. The compressor pistons of each refrigerator are driven sinusoidally, 90° out of phase with the expander pistons. This back-to-back Stirling configuration allows the use of a recuperator (counterflow heat exchanger) instead of a regenerator to provide the thermal isolation between the compressor and the expander piston cylinder. This design allows the removal of the pure ^4He space shown in Fig. 1 since each Stirling refrigerator half acts as a reservoir of ^4He for the other refrigerator.

Real Stirling refrigerators differ significantly from the idealized model described above in several ways. The pistons usually execute harmonic motions rather than discontinuous motions in order to simplify the drive mechanism. The heat transfer within the regenerator is never complete, resulting in fluid entering the piston at temperatures different from T_c and T_h . Gaps in the hot and cold pistons exist, resulting in nonzero clearance volumes. The fluid in the pistons is often not isothermal due to limited heat transfer between the pistons and the surroundings. Finally, the working fluid may not behave as an ideal gas. But despite the differences, the principle of operation is as described above, with the net result being a transfer of heat from the low temperature reservoir to the high temperature reservoir. [1], [2]

1.3 History of SSR.

Kotsubo – Swift

The first prototype of an SSR was built by Kotsubo and Swift. Each piston was made with two nickel bellows which are interconnected by a superleak made of Vycor glass. The regenerator side of both pistons contains the $^3\text{He} - ^4\text{He}$ mixture while the back side of both pistons holds superfluid ^4He . The host piston is pinned at 1.2 K by a ^4He evaporation refrigerator. The regenerator is built out of thirty 200-micron-ID Cu-Ni capillary tubes which are immersed in liquid ^3He . The liquid ^3He acts as the high heat

capacity material in this regenerator. This refrigerator operates at speed of 0.25 rpm with piston volume displacement of 0.9 cm^3 and typical $^3\text{He} - ^4\text{He}$ concentration of 12%. It achieves a temperature of 590 mK and a net cooling power of $5 \text{ }\mu\text{W}$ at 700 mK. This Refrigerator operates at speeds of 0.25 rpm with piston volume displacements of 0.9 cm^3 and typical of 12%. It achieves a temperature of 590 mK and a net cooling power of $5 \text{ }\mu\text{W}$ at 700 mK. However, this design is limited in its performance. The Vycor glass allowed the diffusion of ^3He through superleaks over time. This diffusion decreased performance of the refrigerator and changed the refrigerator properties over time. The speed at which this refrigerator operated was also limited by regenerator design. ^3He has a high heat capacity but low thermal conductivity. Therefore, the thermal penetration depth of the ^3He is small and gets smaller as the frequency of operation is increased. Since the effective heat capacity is proportional to the surface area times thermal penetration depth, the effective heat capacity of the regenerator also decreases with increasing frequency reducing SSR performance.

Brisson – Swift

Brisson and Swift realized the achievable cooling power of the Kotsubo and Swift machine was limited by the regenerator and built a recuperative SSR that consisted of two Stirling refrigerators operating 180 degrees out of phase with each other. These two refrigerators regenerate each other by using a counterflow heat exchanger (the recuperator). This recuperator consist of 238 250 μm ID CuNi capillaries silver soldered in a hexagonally closed pack array with alternating rows corresponding to each half of the SSR. The use of convective heat transfer in a recuperator rather than diffusive heat transfer used in Kotsubo and Swift regenerator design, allows higher operating speed and a corresponding increase in the refrigerator's cooling power because the performance of the recuperator does not depend on the thermal penetration depth of ^3He . The pistons of the SSR are made from stainless steel edge welded bellows with nested convolutions which minimize the clearance volumes within the pistons. Both sides of the pistons are filled $^3\text{He} - ^4\text{He}$ working fluid so that the pistons are double acting. this feature eliminates the Kotsuba and Swift SSR's problem of ^3He diffusing through the superleaks.

The brisson and Swift refrigerator operates at speed of 3 rpm with piston volume displacement of 0.8 cm^3 , $^3\text{He} - ^4\text{He}$ mixture concentration of 6.6% and volume of 7 cm^3 in each half of the refrigerator. Anchoring the hot piston at 1.05 K with a ^4He evaporation refrigerator, this refrigerator achieves a temperature of 296 mK and net cooling power $930 \mu\text{W}$ at 700 mK and $140 \mu\text{W}$ at 500 mK. Operating the same SSR, Watanabe, Swift and Brisson later achieved a temperature of 168 mK while anchoring the hot piston at 387 mK with a ^3He evaporation refrigerator. [2]

Chapter 2.

Properties of $^3\text{He} - ^4\text{He}$ mixture.

2.1 Liquid $^3\text{He} - ^4\text{He}$.

To understand why liquid $^3\text{He} - ^4\text{He}$ mixtures can be used as a Stirling cycle refrigerant requires an explanation of two properties:

- The superfluidity in liquid $^3\text{He} - ^4\text{He}$ mixtures
- The P-V-T relationship of the osmotic pressure of liquid ^3He mixed in liquid ^4He

Above around 2.17 K, depending on pressure liquid ^4He is a normal viscous fluid. However, below 2.17 K ^4He becomes a superfluid.

To better explain this, Figure 2 illustrates two containers. One is filled with pure liquid ^4He and the other with liquid $^3\text{He} - ^4\text{He}$ mixture. Both containers are plugged with a superleak. Superleak is a porous piece of glass which has 0.1 nanometers diameter channels through it. Above 2.17 K, there is no flow out of either container. However below 2.17 K, a portion of the liquid ^4He becomes superfluid and is capable of flowing out of the top container without a pressure drop. Only superfluid ^4He flows out of the bottom container because ^3He becomes a superfluid until milliKelvin. An additional property of superfluid ^4He that needs to be understood is that, at these temperatures, the superfluid portion of liquid ^4He does not participate thermodynamically. Superfluid ^4He is in a ground state and has zero entropy associated with it.

We next consider the P-V-T relation of the osmotic pressure of liquid ^3He in liquid ^4He . The liquid ^3He acts as the high heat capacity material in this regenerator. [1], [2], [6]

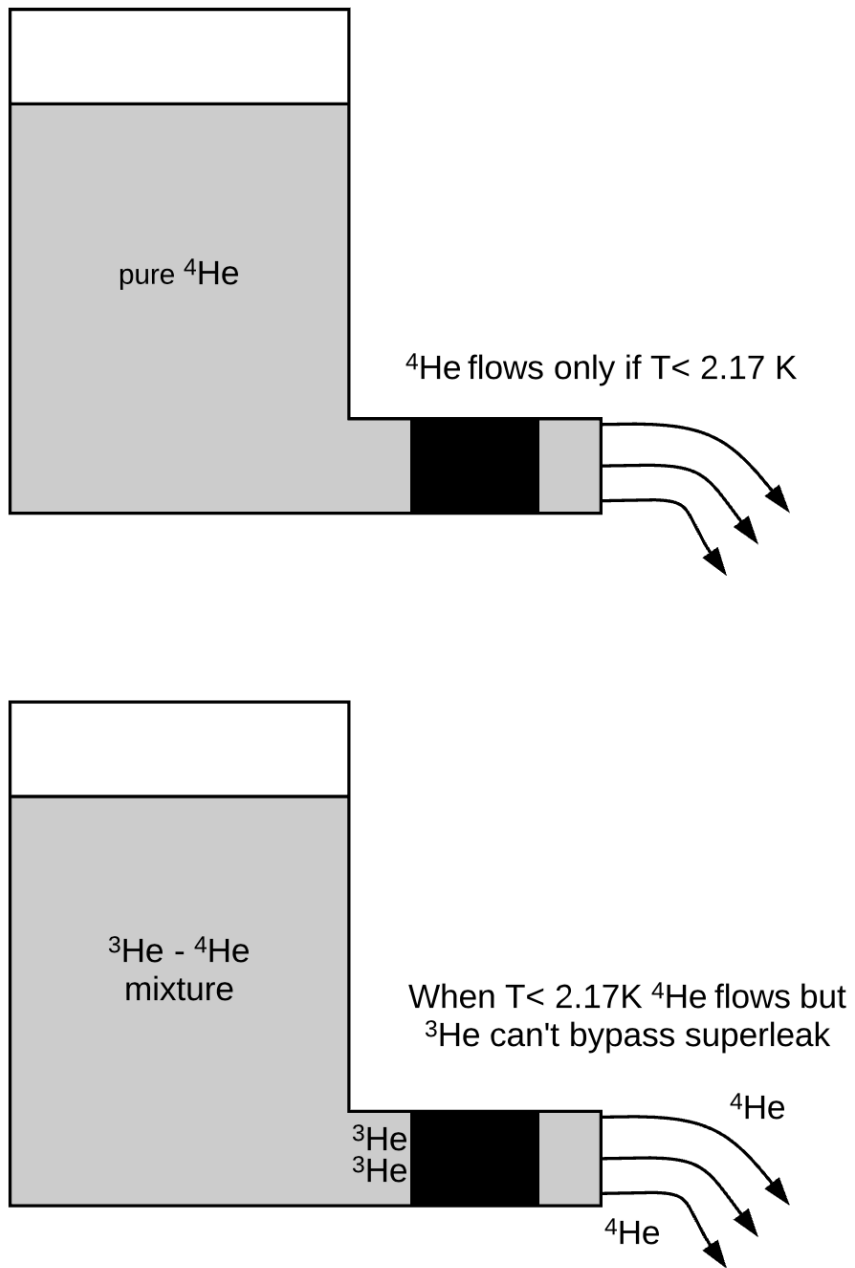


Figure 2. Illustration of the ^4He and $^3\text{He} - ^4\text{He}$ superfluidity.

The liquid ^4He becomes superfluid only when $T < 2.17 \text{ K}$ and starts flowing through the superleaks. In $^3\text{He} - ^4\text{He}$ mixture when $T < 2.17 \text{ K}$, ^4He flows without a pressure drop but ^3He cannot bypass super leak unless. [1], [2], [6]

2.2 Osmotic Pressure.

Starting by defining the osmotic pressure of liquid the osmotic pressure applies only to ^4He . As shown in the Figure 3, there are two containers. One containing liquid $^3\text{He} - ^4\text{He}$ mixture and the other containing pure ^4He at uniform temperature. The containers are connected by a superleak which allows the free passage of only the ^4He between the containers. The figure 3 shows the pressure difference between the two containers. It is the pressure difference that a rigid semipermeable membrane, permeable only to ^4He , must sustain in order to maintain partial mutual stable equilibrium – equal temperatures and chemical potentials – between the $^3\text{He} - ^4\text{He}$ mixture and the pure ^4He . This pressure difference prevents the migration of ^4He through the superleak that would be driven by a chemical potential difference it applies to ^4He so it will always obey BE stats.

The important concepts to understand in this section are that there are two helium liquids mixed together. To first approximation, the ^4He liquid is a superfluid and thermodynamically does not participate while the ^3He liquid behaves as an ideal Boltzmann gas. This combination of properties allows $^3\text{He} - ^4\text{He}$ mixture to be used as a Stirling cycle refrigerant. [6]

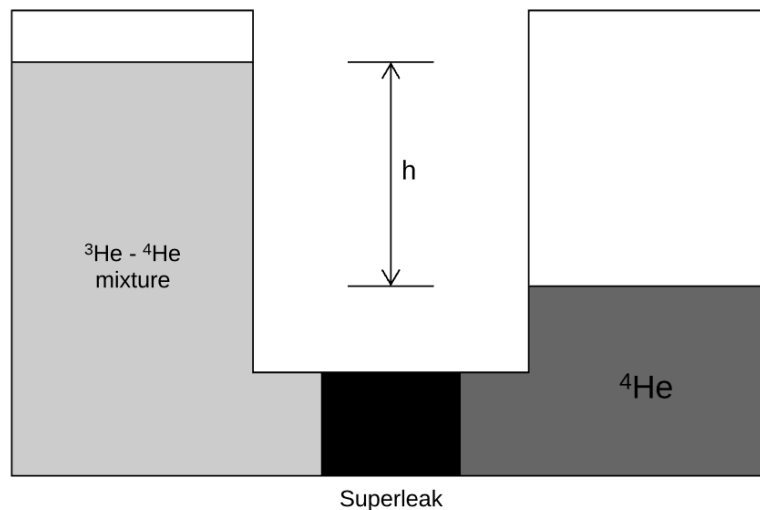


Figure 3. Osmotic pressure

Illustration of osmotic pressure of liquid $^3\text{He} - ^4\text{He}$ mixtures. Both container held at the same temperature. [1], [2]

2.3 Kapton.

As we analysed previously, the model that will be used in the simulations in this thesis will be of a recuperator design. But the main obstacle to low temperature, high efficiency recuperators is the high Kapitza boundary resistance. Kapitza boundary resistances dominate the overall heat transfer coefficient in heat exchangers at low temperatures. Initially, for the recuperator metals were considered most suitable for their high thermal conductivity. But experiments[4] has shown that metals cannot be used due to their extremely large internal volumes. Large internal volumes in a SSR result in a reduced pressure oscillation, consequently and reduced cooling. As Frossati realized, the Kapitza resistance between helium and plastics is about ten times lower than the resistance between helium and metals and thin plastics membranes between two helium flows have better heat transfer characteristics than a corresponding metal heat exchanger at low temperatures. So Kapton's properties will be used to simulate recuperators. [4]

Chapter 3.

Ideal Mathematical Model

We review here the Schmidt model for the superfluid stirling refrigerator. This model is the zeroth order model for the design of any Stirling machine and was used extensively in the original designs of SSR's. The Schmidt model is an ideal model; there are no viscous effects and all heat transfers occur reversibly. Consequently, the efficiency of this cycle is that of Carnot the Schmidt Model uses a regenerator and we outline the model here for completeness in order to compare the results with the practical model – recuperator design that we will use later in the simulations. Also, we are going to review the importance of the angle out of phase the two Stirling refrigerants are operating.

3.1 Schmidt Model of SSR – The ideal model.

Schmidt model is entirely equivalent to modeling the SSR depicted in Figure1. The temperature of the liquid throughout each piston-cylinder is assumed to be at the temperature of the local heat reservoir. We also assume that the ^3He behaves as an ideal Boltzmann gas in an inert back-ground of superfluid ^4He . The volumes of the piston cylinder are assumed to be driven sinusoidally, namely as

$$V_c = V_{clc} + V_{swc} \left(\frac{1 + \cos \theta}{2} \right) \quad (1)$$

$$V_e = V_{cle} + V_{swe} \left(\frac{1 + \cos(\theta + a)}{2} \right) \quad (2)$$

Where V_c , V_{clc} , V_{swc} , are the compressor piston's total volume, clearance volume and swept volume, respectively. Similarly, V_e , V_{cle} , V_{swe} , are the expander piston's total

volume, clearance volume and swept volumes. The crank angle θ , is assumed to vary linearly with time, $\theta = \omega t$, where ω is the angular frequency and t is the time. There is a phase shift between the pistons (compressor and expander) equal to $\alpha = 90^\circ$.

Because of the infinite heat capacity of the ideal regenerator matrix and, by assumption, the regenerator is in steady state, the temperature distribution in the ideal regenerator is independent of time. Since the mass flows through the regenerator is balanced (the net mass flow through the regenerator is zero) and the heat capacity of the ideal gas is a constant, the temperature in the regenerator varies linearly with position from the compressor temperature to the expander temperature.

Since the pressure drops associated with the viscous flow of the ideal gas are assumed negligible, the pressure is spatially uniform throughout the Stirling machine but the time varying. The total number of moles of ^3He , N_3 , in the Stirling refrigerator can be written as:

$$N_3 = \frac{pV_c}{RT_c} + \frac{pV_R}{RT_{LM}} + \frac{pV_e}{RT_e} \quad (3)$$

where p is the time varying pressure, R is the universal gas constant, V_r is the regenerator volume and T_c and T_e are the compressor and expander operating temperatures. T_{LM} is the log mean temperature of the regenerator ($= (T_c - T_e)/\ln(T_c/T_e)$). From Eq. (3), the pressure associated with the ^3He is then

$$p = \frac{N_3 R}{\frac{V_c}{T_c} + \frac{V_r}{T_{LM}} + \frac{V_e}{T_e}} \quad (4)$$

The time dependence of p is buried in the time dependent variation of V_c and V_e . The work per cycle done by the expander and compressor is determined by integrating $p dv$ for each piston over one cycle.

The results are:

$$W_c = \pi V_{swc} \frac{NR}{S_s \sqrt{1 - b_s^2}} \left(\frac{\sqrt{1 - b_s^2} - 1}{b_s} \right) \sin(\beta) \quad (5)$$

$$W_c = \pi V_{swe} \frac{NR}{S_s \sqrt{1 - b_s^2}} \left(\frac{\sqrt{1 - b_s^2} - 1}{b_s} \right) \sin(\beta - a) \quad (6)$$

where

$$S_s = \frac{V_{clc}}{T_c} + \frac{V_{swc}}{2T_c} + \frac{V_r}{T_{LM}} + \frac{V_{cle}}{T_e} + \frac{V_{swe}}{2T_e} \quad (7)$$

$$\beta = \tan^{-1} \left(\frac{V_{swc} \sin\left(\frac{a}{T_e}\right)}{V_{swe} \sin\left(\frac{a}{T_e}\right) + \frac{V_{swc}}{T_c}} \right) \quad (8)$$

$$c = \frac{1}{2} \sqrt{\left(\frac{V_{swe}}{T_e}\right)^2 + \frac{2V_{swc}V_{swe}}{T_c T_e} \cos a + \left(\frac{V_{swc}}{T_c}\right)^2} \quad (9)$$

$$b_s = \frac{c}{S_s} \quad (10)$$

The heat transfers to the gases in the expander can be calculated by applying energy conservation to the expander. For an ideal gas, the energy equation is

$$\frac{dE}{dt} = \dot{Q} - \dot{W} + \dot{N}_3 C_p T \quad (11)$$

where E is the internal energy of the ^3He in the control volume, \dot{Q} and \dot{W} are the rate of heat and work transfer positive \dot{N}_3 is the molar flow rate of ^3He into the expander

volume, C_p is the molar heat capacity at constant pressure ($=5/2R$) of the ^3He and T is the temperature of the ^3He entering and exiting the expander. As we are dealing with ideal components, the temperature of the gas entering and leaving the expander is always T_e . If Eq. (11) is integrated over on cycle, $\dot{N}_3 C_p T$ term reduces to a constant, $C_p T_e$, times a cyclic integral of the molar flow into and out of the control volume. This integral is zero by conservation of mass. Similarly, the internal energy term integrates to zero because the starting and finishing states are the same. We conclude that the heat absorbed by the expander on the compressor result in the conclusion that the heat absorbed by the compressor is equal to the work done by the compressor or $Q_c = W_c$.

The coefficient of performance of the refrigerator, COP, is defined as the heat transferred to the expander divided by the total work absorbed by the refrigerator. Using Eqs. (5) and (6), we find the coefficient of performance of a Schmidt cycle refrigerator is [2], [8]

$$COP = -\frac{Q_e}{W_c + W_e} = \frac{T_e}{T_c + T_e} \quad (12)$$

3.2 Recuperator design – Practical model.

In the practical model, that it will detailly explained in this thesis, there is a heat exchange between the two SSRs so a we will use a recuperator instead of a regenerator.

3.2.1 Basic Principles.

The recuperator design will be consisted of two Sterling Refrigerants operating out of phase but each compressor piston will be 90° out of phase with the respectively expander piston. The working fluid, ^3He , after it will be compressed in the compressor cylinder it will flow through the recuperator before entering the expander cylinder. Each refrigerator will repeat this but in different timestep. The recuperator will be a wall made from Kapton and the two coupled walls will have a very high heat exchange capacity. And this is the

fundamentally difference in this design for the Schmidt model. This out of phase operation will result in a mass flow imbalance within the recuperator so it will operate as a heat exchanger providing the necessary thermal isolation between the compressor and the expander piston cylinder. As shown in Figure 5, the recuperator design can provide cooling power when it works out of phase for a certain range of angles. Operating out of this range, the recuperator does not provide the appropriate heat exchange and thermal isolation in order to achieve the net effect and transfer heat from the cold temperature reservoir to the hot temperature reservoir.

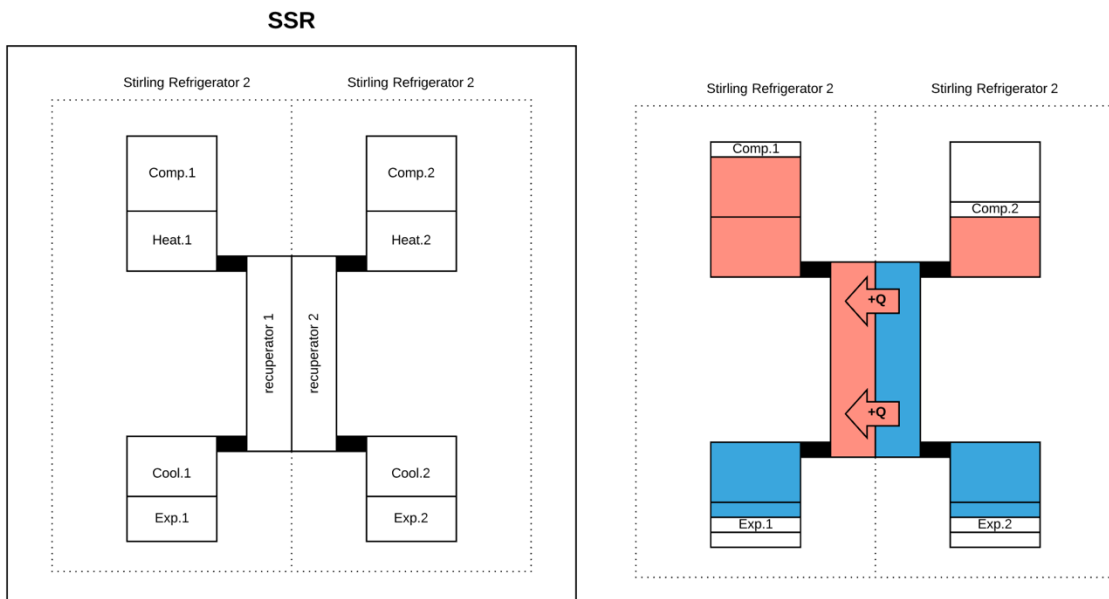


Figure 4. Components of a recuperator design SSR.

The recuperator design consists two Sterling refrigerators, operating out of phase. More specifically of the right side the operating phase is 180° in their initial positions.

3.2.2 Cycle Explanation.

In Figure 5 we simulate the motion of every piston. Every piston executes a harmonic sinusoidal motion and every period is divided to 360° crank shaft angle.

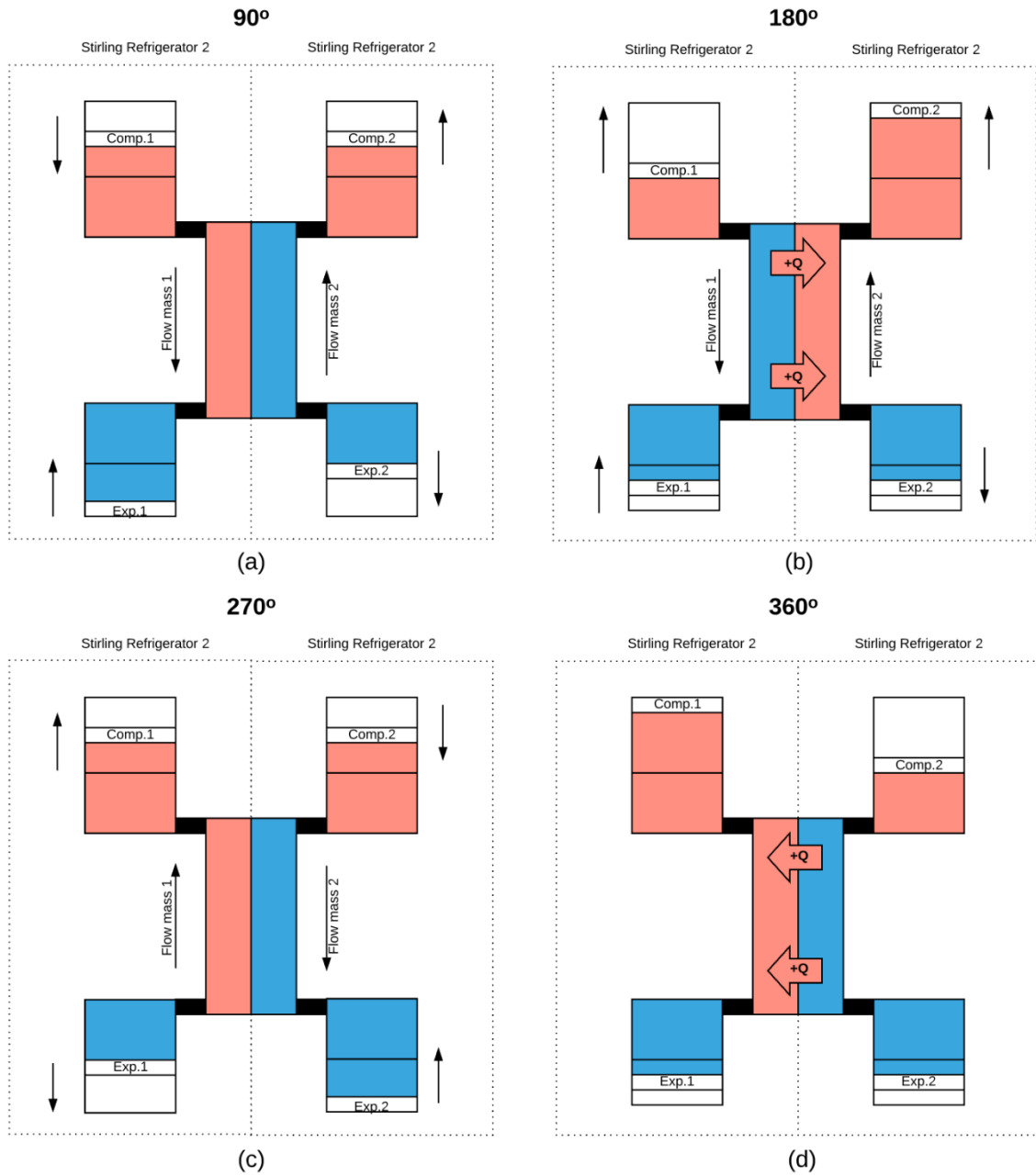


Figure 5. Pistons' motions proportional to time in crank shaft angles for one period. *The two refrigerators operate 180° out of phase. We can observe the initial position for every piston in Fig.4 which is the same as in 360° considering that this a sinusoidal motion. In Fig.5 we can observe the movement for every piston with time step 90°. The black arrows indicate the direction of the velocity that every piston is about to start executing.*

In Fig. 4 the pistons are in their initial positions. One period is divided to 360° , so in Fig. 5 (d) the pistons are again in their initial positions. All the pistons will execute a simple harmonic oscillator and we will explain step by step the phases with a time step 90° .

1. Initial position – 0°

Refrigerator 1

- i) Compressor Piston¹: The compressor piston is in the maximum positive value with the the working fluid complete uncompressed.
- ii) Expander Piston¹: The expander piston is 90° out of phase so is in the equilibrium position.

Refrigerator 2

- i) Compressor Piston²: The compressor piston is in the maximum negative value. This means that the working fluid is fully compressed in the compressor's cylinder and have already flowed through the recuperator in order to expand in the expander's cylinder.
- ii) Expander Piston²: The expander piston is 90° out of phase so is in the equilibrium position.

2. Fig. 5 (a) – 90°

Refrigerator 1

- i) Compressor Piston¹: The compressor piston is in the equilibrium position and is moving to the maximum negative position in order to fully compress the working fluid.

- ii) Expander Piston¹: Simultaneously, the expander piston, after 90°, has reached the maximum negative position. The working fluid has fully expanded, absorbing the heat from the cold reservoir.

Refrigerator 2

- i) Compressor Piston²: The compressor piston is in the equilibrium position and is moving to the maximum positive position in order to isochorically with the expander displace the working fluid through the recuperator.
- ii) Expander Piston²: Simultaneously, the expander piston, after 90°, has reached the maximum positive position. The working fluid has flowed isochorically through the recuperator back to the compressor cylinder.

3. Fig. 5 (b) – 180°

Refrigerator 1

- i) Compressor Piston¹: The compressor piston is in the maximum negative value. This means that the working fluid is fully compressed in the compressor's cylinder and have already flowed through the recuperator in order to expand in the expander's cylinder.
- ii) Expander Piston¹: The expander piston is in the equilibrium position and is about to start isochorically displacing the working fluid through the recuperator back to the compressor cylinder.

Refrigerator 2

- i) Compressor Piston²: The displacement has been finished. The compressor piston is in the maximum positive value with all the working fluid inside the compressor cylinder, complete uncompressed.
- ii) Expander Piston²: The expander piston is in the equilibrium position and is about to start expanding the fluid.

4. Fig. 5 (c) – 270°

Refrigerator 1

- i) Compressor Piston¹: The compressor piston is in the equilibrium position and is moving to the maximum positive position in order to isochorically with the expander displace the working fluid through the recuperator.
- ii) Expander Piston¹: The expander piston has reached the maximum positive position. The working fluid has flowed isochorically through the recuperator back to the compressor cylinder.

Refrigerator 2

- i) Compressor Piston²: The compressor piston is in the equilibrium position and is moving to the maximum negative position in order to fully compress the working fluid.
- ii) Expander Piston²: Simultaneously, the expander piston, after 90°, has reached the maximum negative position. The working fluid has fully expanded, absorbing the heat from the cold reservoir.

5. Fig. (d) – 360° Initial position

Refrigerator 1

- i) Compressor Piston¹: The compressor piston is in the maximum positive value with the the working fluid complete uncompressed.
- ii) Expander Piston¹: The expander piston is 90° out of phase so is in the equilibrium position.

Refrigerator 2

- i) Compressor Piston²: The compressor piston is in the maximum negative value. This means that the working fluid is fully compressed in the compressor's cylinder and have already flowed through the recuperator in order to expand in the expander's cylinder.
- ii) Expander Piston²: The expander piston is 90° out of phase so is in the equilibrium position.

The most important concept to understand is that as shown in Fig. 5 (b), in SR1 the working fluid is in the hot reservoir temperature and is being displaced through the recuperator. At the same time in the SR2, the working fluid is in cold reservoir and is being displaced through the recuperator back to the compressor cylinder. As a result, the kapton made recuperator with the coupled walls with very high heat conductivity, transfers the heat from the hot mass flow (SR1) to the cold mass flow (SR2). After the heat exchange inside the recuperator, the SR1 working fluid inserts the expander cylinder in cold reservoir temperature and the SR2 working fluid enter the cylinder with temperature same as the hot reservoir. This is the fundamental principles of the recuperator design and we can achieve the net effect.

As we can safely assume, the recuperator design despite the many advantages, can only provide cooling power when the SR1 and SR2 are operating out of phase in a specific

range of angles. For example, when the SR1 and SR2 are operating without a phase difference as shown in Fig. 6, the two different flows inside the recuperator have the same temperature – hot reservoir temperature. Both compressor and expander pistons execute the same motion at the same time. At Fig 6 (a) we can clearly observe that two hot flows enter the recuperator simultaneously. This results to no heat exchange between the two flows causing the flows entering the expander cylinder, in both SR, in the temperature of the hot reservoir. This effect is against the fundamental principle and as a result we cannot achieve the net result so the SSR cannot provide cooling power. We can safely assume that the same effect will occur for angles from 0° to 90° or from 270° to 360° approximately unless we can achieve a counterflow inside the recuperator. At the next chapter, we are going to set up a case of a SSR in ANSYS FLUENT and we are going to search and determine the exact amplitude of crank shaft angles, that the two SR can operate in order to achieve cooling power. Also, we will evaluate the results in this range and considering the COP (Coefficient of Performance) as main factor we will search for the optimum design point comparing the results with the ideal – Schmidt model.

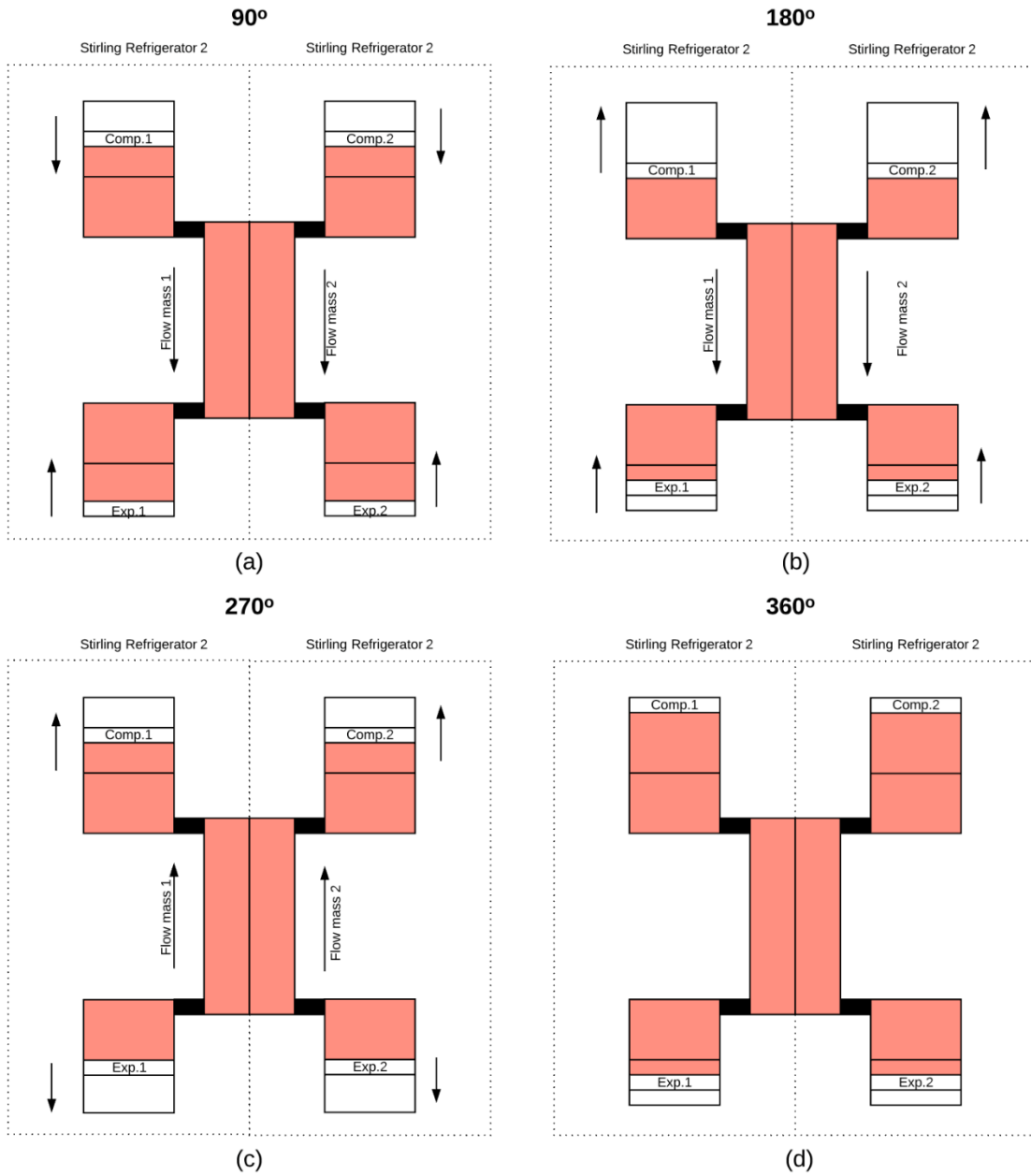


Figure 6. Pistons' motions proportional to time, operating at the same phase.

When the two SR operate at the same phase there is no heat exchange in the recuperator resulting to a SSR with zero cooling power.

Chapter 4.

Determining the operating range with ANSYS FLUENT.

At the next chapter, we are going to set up a case of a SSR in ANSYS FLUENT and we are going to search and determine the exact phase difference in degrees, that the two SR can operate in order to provide cooling power. Initially we will determine the proper geometry. Considering that CFD solution are time-consuming there is a need for a simple design but that it will provide at the same time preciseness and acceptable results. Next step is to determine the best mesh methods for each component of the geometry. Last but not least we will set up the fluent case that requires material properties, equations that will be solved and most important the User Defined Functions that will express the piston motions. [9], [10], [11]

4.1 Geometry.

As described previously a simple geometry is a necessity, so the SSR will consist of two SRs. In more detail, there will be two compressors, each connected through a recuperator to an expander as shown in Fig. 7. The two SRs will share a common face with very high heat exchange capacity. All the external walls will be adiabatic so the exogenous factors won't affect the model. Each compressor's cylinder will be connected to a heat chamber that will operate as a hot reservoir. Respectively, each expander's cylinder will be connected with a cold chamber that will operate as cold reservoir.

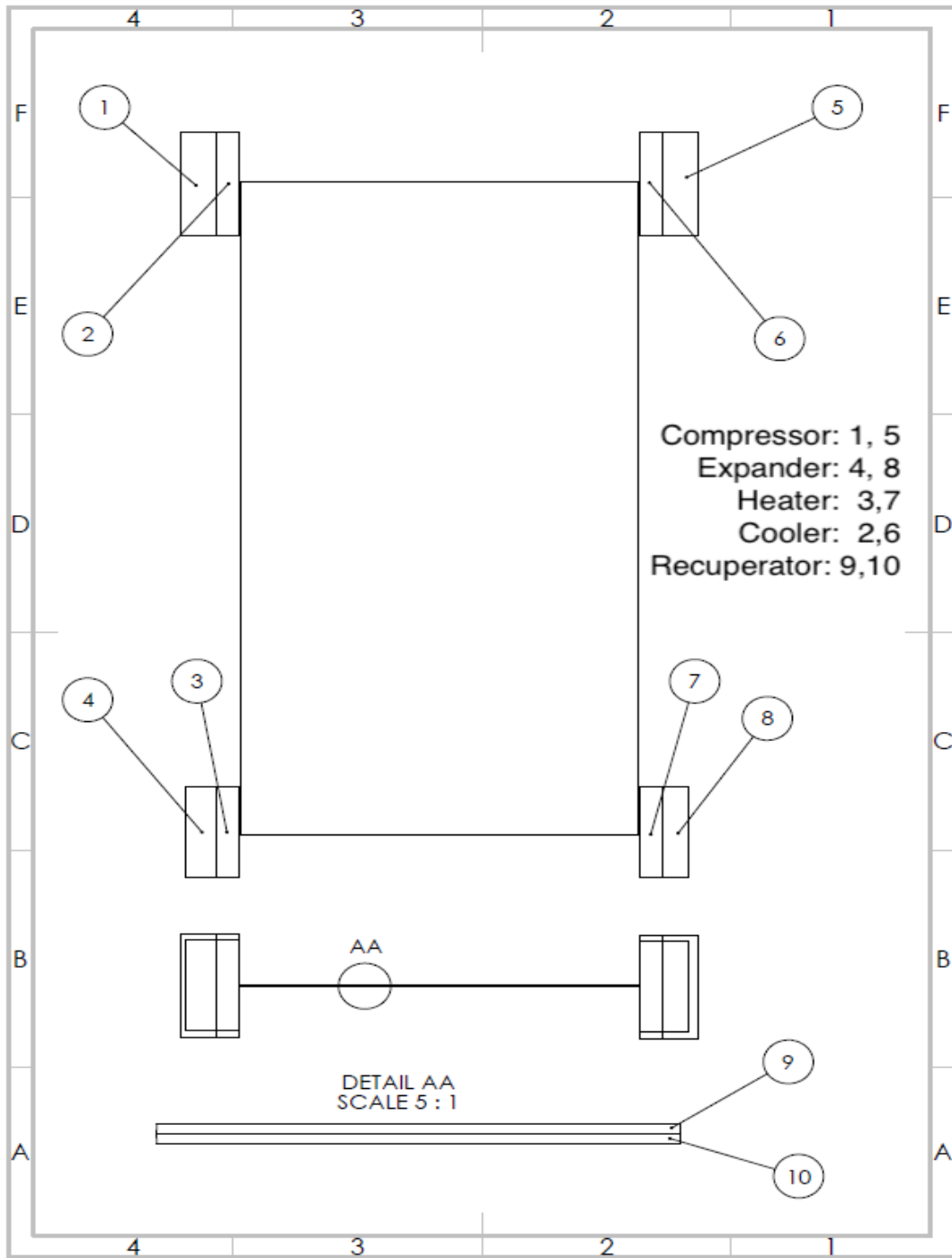


Figure 7. Components of a SSR.

The Sterling refrigerant consist of a compressor which connects to an expander thought the recuperator. Two SRs that their recuperator share a common wall make up a SSR.

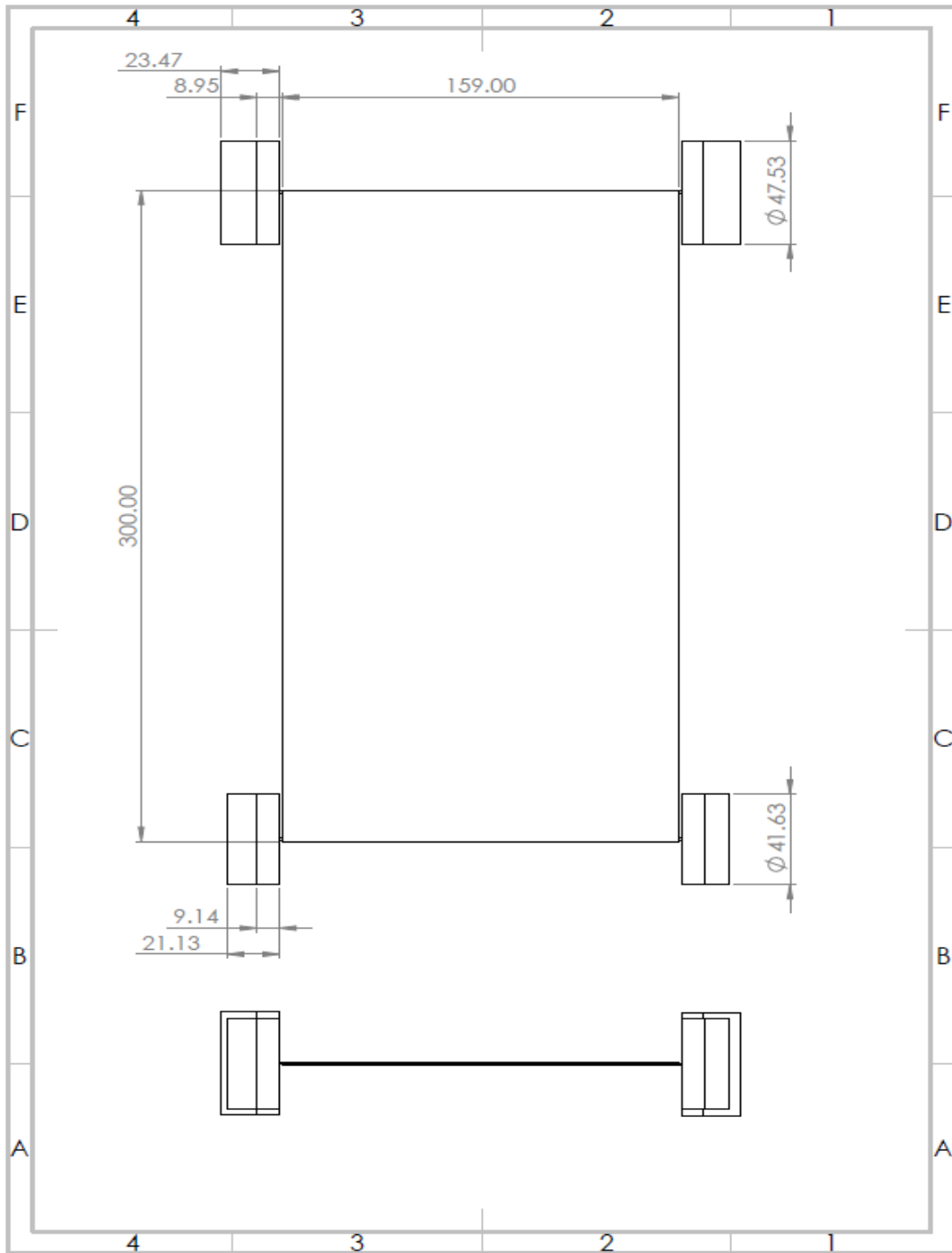


Figure 8. Dimensions of a SSR.

Dimension of SSR when every cylinder has its maximum volume. Clearly the combination of expander – heater result in a smaller volume than the compressor-cooler.

The geometry shown in Fig. 8 every compressors and expanders cylinders volumes have their maximum value. Practically this never happens because the pistons will never synchronize like this. As the SR will work out of phase, in every timestep the volumes will differ as a result of the motion of the piston. So, for every angle out of phase we will simulate we have to determine the exact initial position of the pistons by calculating the initial volume of the pistons' cylinders as shown below.

4.1.1 Generate Geometry.

Initially from the design the ratios of the Cooler are known and the dead volume of the Compressor cylinder too. The Heater and the dead volume Expander and the nominal clearance volumes as well.

Ratios:

$$rV_c = 0.665 \quad (14)$$

$$rV_e = 0.517 \quad (13)$$

Nominal clearance Volumes

$$V_{clc,nom} = 23.887 \text{ cm}^3 \quad (15)$$

$$V_{cle,nom} = 24.062 \text{ cm}^3 \quad (16)$$

Swept volumes

$$V_{swc} = 17.74 \text{ cm}^3 \quad (17)$$

$$V_{swe} = 9.39 \text{ cm}^3 \quad (18)$$

So the volume every timestep is

$$V_c(t) = V_{clc} + \frac{1}{2} V_{swc} [1 + \cos((2 \cdot f \cdot \pi \cdot (t + \Delta t_1)))] \quad (19)$$

$$V_e(t) = V_{cle} + \frac{1}{2} V_{swe} [1 + \cos((2 \cdot f \cdot \pi \cdot (t + \Delta t_2)))] \quad (20)$$

where

$$V_{clc} = (1 - rV_c) V_{clc,nom} \quad (21)$$

$$V_{cle} = (1 - rV_e) V_{cle,nom} \quad (22)$$

The SSR will operate in a specific frequency. In our model, the operating frequency will be as low as 1 Hz so turbulence effects will be avoided and the flow will be laminar. No turbulence is expected also due to the temperatures and the flow concentration of ^3He

The Δt_1 and Δt_2 variables are determine the out of phase operation in the following way

$$\text{Operating Frequency: } f = 1 \text{ Hz} \quad (23)$$

$$\text{Operating Period: } T = \frac{1}{f} = 1 \text{ s} \quad (24)$$

$$T' = \frac{T}{s} \quad (25)$$

$$\text{The out of phase that the two SR are operating: } \Delta\varphi_1 \quad (26)$$

$$\text{The out of phase comp. and exp. pistons are} \quad : \Delta\varphi_2 \quad (27)$$

are operating in every SR

So, the Δt_1 and Δt_2 variables that will be used in the UDF and will determine the initial positions of the pistons are

$$\Delta t_1 = \frac{\Delta\varphi_1}{360} \cdot T' \quad (28a)$$

$$\Delta t_2 = \frac{\Delta\varphi_2}{360} \cdot T' \quad (28b)$$

Finally, with the volume of each cylinder calculated it is easy to calculate the length that every cylinder surface needs to be extruded in order to achieve the specific volume.

In more details

Total area of Compressor Cylidner: 47.53 mm

Total area of Expander Cylidner : 41.63 mm

So,

$$L_{initial_c} = \frac{V_c(0)}{A_{p_c}} \quad (29)$$

$$L_{initial_e} = \frac{V_e(0)}{A_{p_e}} \quad (30)$$

As we mentioned earlier, the equations for the initial lengths will be used only when modeling the SR2. The SR will always operate with the same initial data and only the SR2's data will differ so the desirable out of phase angle is achieved.

For example, when the two SR are operating 180° out of phase the initial and operating conditions are shown in Fig. 9. As it is logical, when operating at 180° the movements and the positions will be equals but in different direction. The SR1-machine1 compressor's initial point will be in the maximum positive value with a velocity directed towards the negative. Similar, the SR1-machine1 expander's initial point will be in the equilibrium position with a velocity directed towards the positive. Something that is correct considering the the C1 and E1 pistons are operating 90° out of phase each other so at the next T/4 the E1 will be at the where the C1 was initially. Equally the SR2-machine2 compressor's initial point will be in the maximum negative value with a velocity directed towards the positive. Also, the SSR2-machine2 expander's piston will be initially at the

equilibrium position directed towards the negative values (same way as explained with the SSR1).

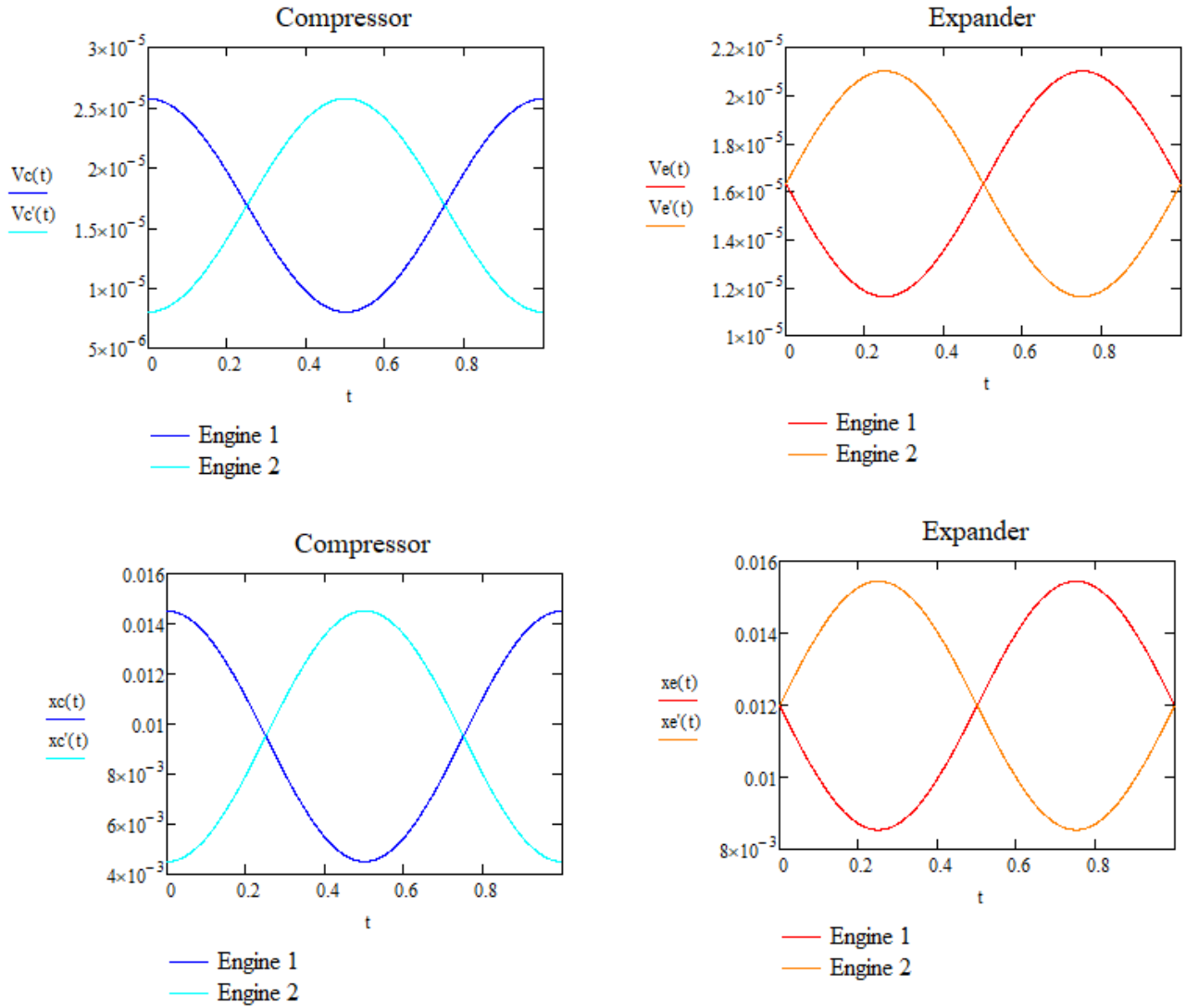


Figure 9. Positions and Movements of all the pistons.

The V_c, V_c' are referred the SR, SR2 compressors' velocities respectively. In the same way the V_e, V_e' are referred to the expanders' velocities. Equally the x and x' are referred to the positions.

4.1.2 Import Selected Geometry.

After each specific geometry for every out of phase initial case is generated, it must be imported in DesignModeler. In the ANSYS Workbench Project schematic, right-click the geometry and from the drop down menu we choose New DesignModeler Geometry as shown in Fig. 10. When ANSYS DesignModeler first appears, we should select desired system of length units to work from. For the purposes of this thesis we set the units to Millimeters. After that we choose File and from drop down menu we choose Import External Geometry File. Then we choose our file and right-click generate. For simplicity purposes the geometry file was created in external CAD design program Solidworks. It can be created inside the design modeler as well. After the geometry is generated as shown in Fig. 11.

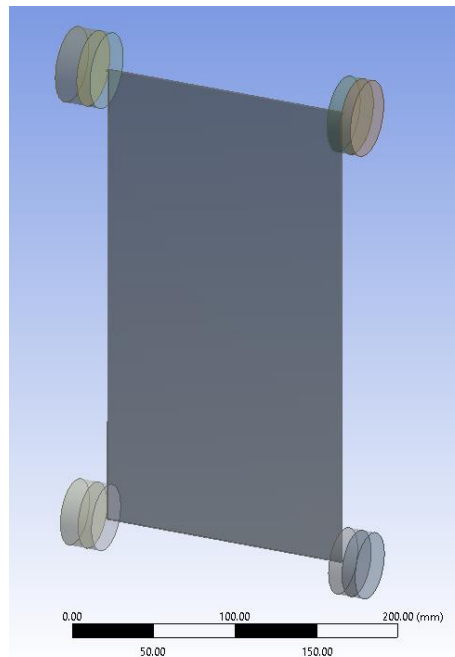


Figure 10. Generated Geometry.

After we import the geometry and choose the desirable length unit.

Also, as shown in Fig. 12 in the Tree Outline view we can from how many parts our geometry consist of as long as every part's name as shown in.

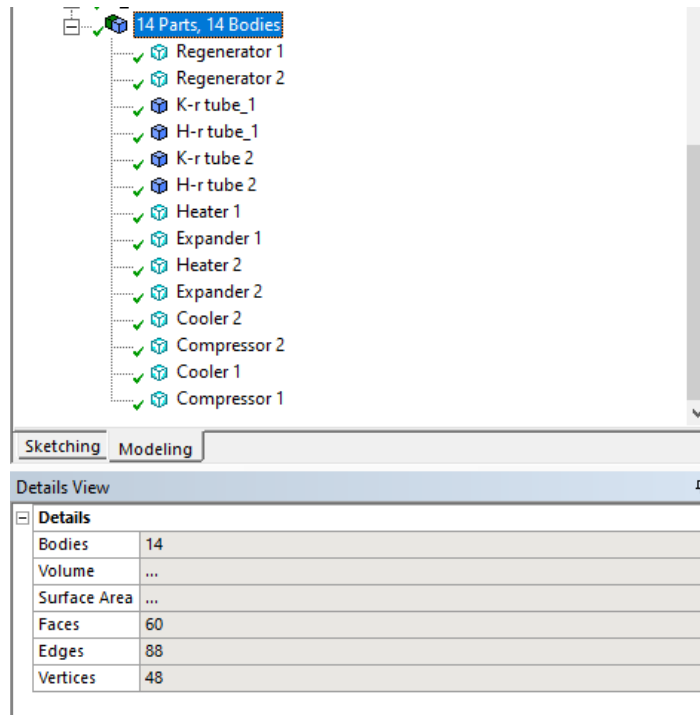


Figure 12. Geometry – Tree Outline view

In the Tree Outline view we can see all the parts that our geometry consists of

Last but not least If you would rather not create the geometry in ANSYS DesignModeler, you can import a pre - existing geometry by right-clicking the Geometry cell and selecting the Import Geometry option from the context menu directly from ANSYS Workbench Schematic as shown in Fig. 13. Finally, after the geometry is either imported or created we hit save project and simply exit the ANSYS DesingModeler.

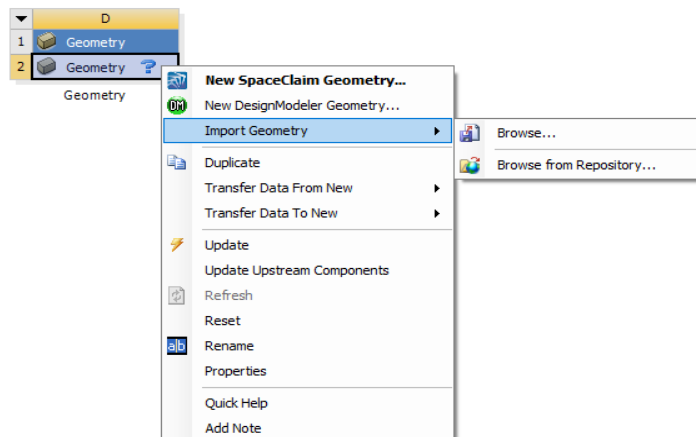


Figure 13. Geometry imported directly from schematic menu.

Instead create the geometry inside the DesignModeler we can import a pre – existing.

4.2 Meshing techniques.

The application of CFD is rapidly expanding with the growth in affordability of computational resources. It is becoming essential for CFD solvers to provide validation. Mesh related issues play a very important role on accuracy and convergence. This means that an incorrect meshing method could result to a very time-consuming solution or even worse to non-providing a solution due to errors such as floating-point error. As a result, a proper meshing way for each domain it's necessary.

Initially, a pre-processing step for the computational field simulation is the discretization of the domain of interest and is called mesh generation. The process of mesh generation can be broadly classified into two categories based on the topology of the elements that fill the domain. These two basic categories are known as structured and unstructured meshes. The different types of meshes have their advantages and disadvantages in terms of both solution accuracy and the complexity of the mesh generation process. A structured mesh is defined as a set of hexahedral elements with an implicit connectivity of the points in the mesh. The structured mesh generation for complex geometries is a time-consuming task due to the possible need of breaking the domain

manually into several blocks depending on the nature of the geometry. An unstructured mesh is defined as a set of elements, commonly tetrahedrons, with an explicitly defined connectivity. The unstructured mesh generation process involves two basic steps: point creation and definition of connectivity between these points. Flexibility and automation make the unstructured mesh a favourable choice although solution accuracy may be relatively unfavourable compared to the structured mesh due to the presence of skewed elements in sensitive regions like boundary layers. In an attempt to combine the advantages of both structured and unstructured meshes, another approach in practice is hybrid mesh generation. In a hybrid mesh, the viscous region is filled with prismatic or hexahedral cells while the rest of the domain is filled with tetrahedral cells. It has been observed that a hybrid mesh in viscous regions creates a lesser number of elements than a completely unstructured mesh with a similar resolution. This type of mesh has no restrictions on the number of edges or faces on a cell, which makes it extremely flexible for topological adaptation. It is given that unstructured mesh has an advantage over the structured mesh in handling complex geometries, mesh adaptation using local refinement and de-refinements, moving mesh capability by locally repairing the bad quality elements, and load balancing using appropriate graph partitioning algorithms. In the case of a non-matched block-to-block boundary, interpolation issues have to be handled properly to satisfy the conservation principles. However, the structured mesh has a better accuracy for viscous calculations due to the fact that it can handle cells with very high aspect ratio cells in the boundary layer.

In this specific simulation in order to choose the appropriate meshing technique for each domain we have to consider two critical factors. First, due to the fact that the SSR operates in very low Hz and low temperatures the flow could be considered laminar, so there will be no turbulence effects. Secondly, that all the compressors and expanders' cylinders will be a moving – dynamic mesh in cylindrical volume. As a result, the techniques will be used is Face – Sizing for all the domains and Sweep Method for all the cylinders due to dynamic mesh.

4.2.1 Cylinder Meshing.

After the discretization of the domain of interest was made, all the cylindrical volumes were grouped and meshed together. Initially as shown in Fig. 14 a Face-sizing technique was chosen. After some research, it was decided that the acceptable and most efficient element size for these domains was $7.5e-004$ meters.

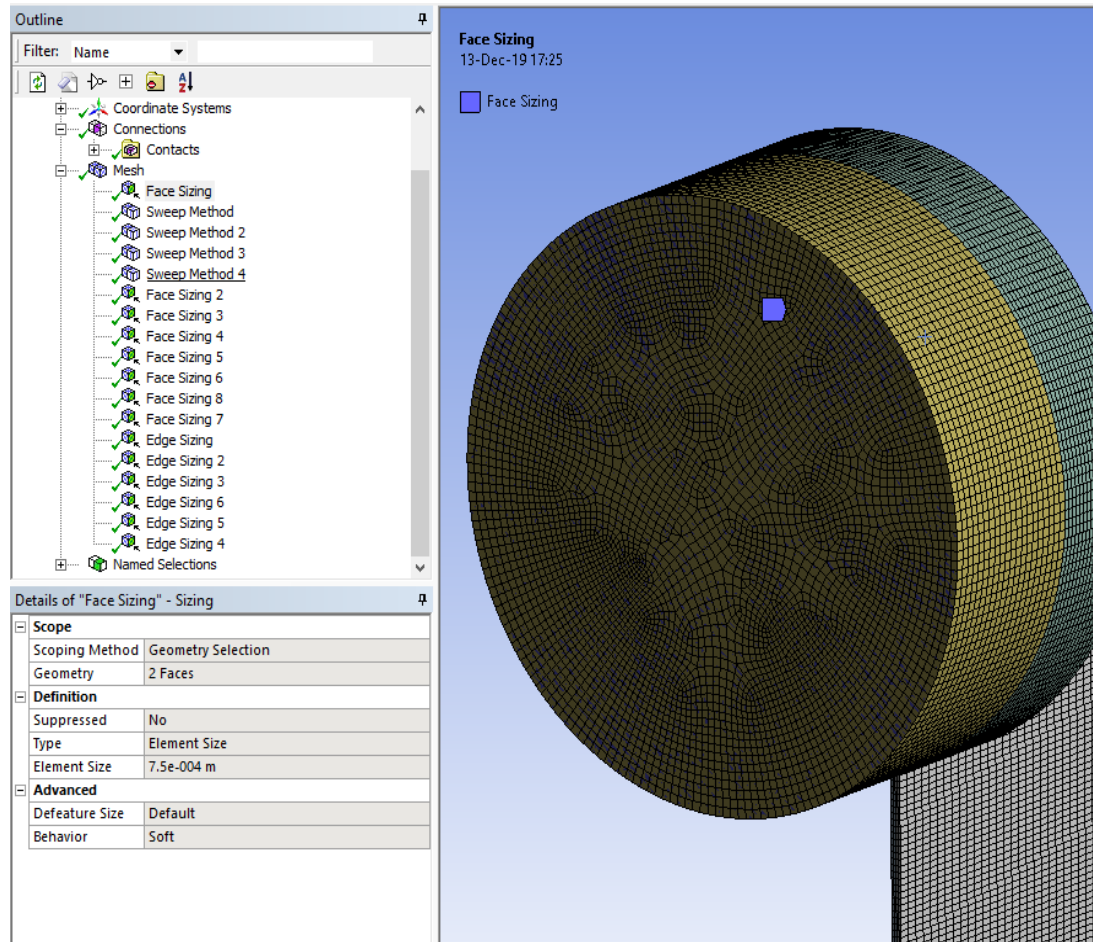


Figure 14. Using the Face-sizing method.

In all domains Face-sizing method was used to set the maximum acceptable and efficient element-size. In the cylindrical faces it was set to $7.5e-004$ meters.

After setting the appropriate element size the Sweep-method was used for a better mesh of the cylindrical volumes as shown in Fig. 15.

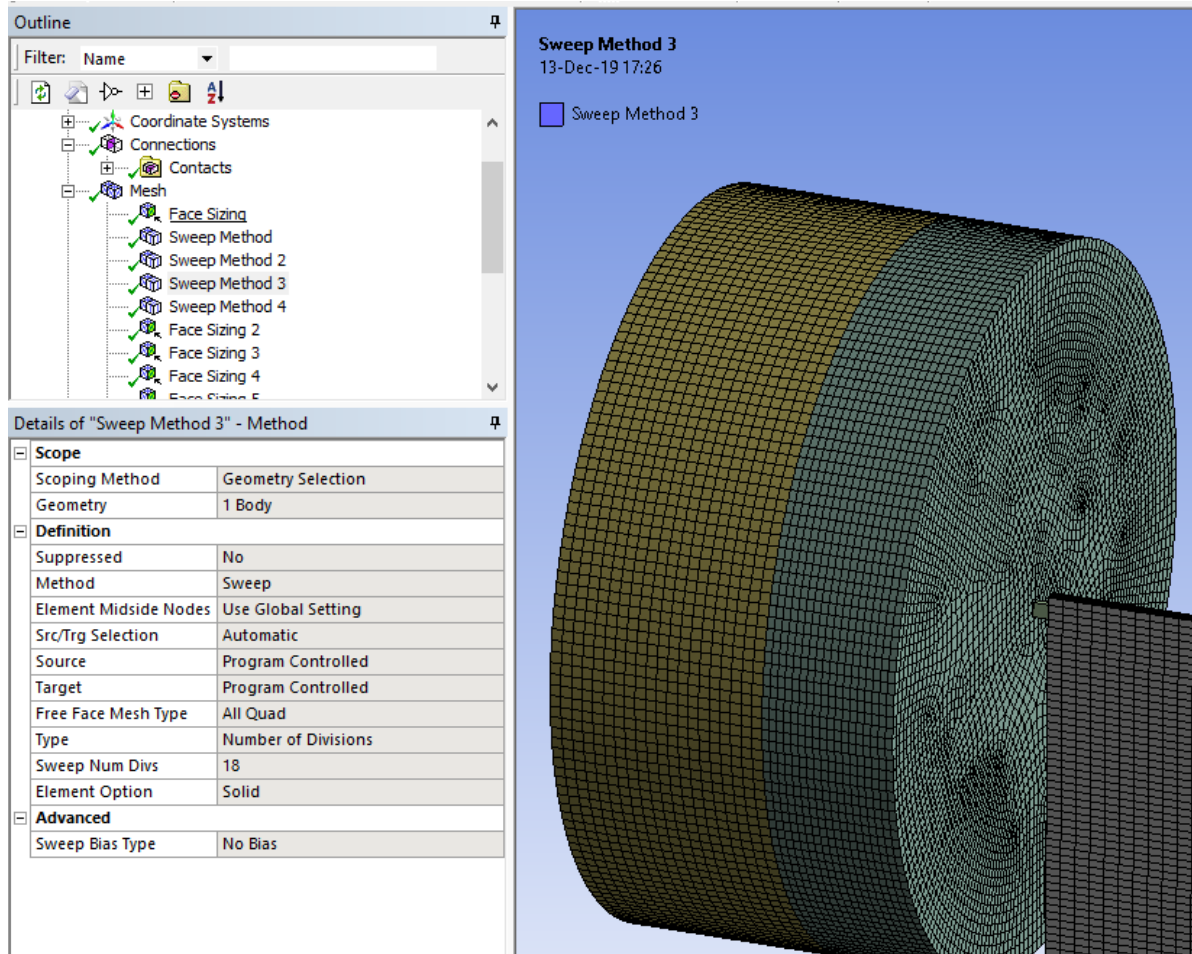


Figure 15. Using the Sweep Method.

Sweep-method was used in all the cylindrical volumes for a better meshing.

The Free Face Mesh type was set to All Quad and the number of divisions was set 18 for the Heaters and Coolers and 20 for the Compressors and Expanders. Despite that the length of the heater/cooler is enough smaller than the length of the Compressor/Expander it was almost used the same number of divisions. Because as explained previously, the heater will operate as a hot reservoir dissipating the heat, resulting to a greater need for more detailed mesh in order to calculate the Coefficient Of Performance in every case (COP).

4.2.2 Recuperator Meshing.

In the same way, the Recuperator was meshed using the Face-sizing technique. The basic difference this time is that instead of Element – size the Number of Divisions method was used. The number of elements in the longitudinal direction was set to 600, in the transverse direction was set to 160 and the thickness was modeled with 2 elements as shown in Figure 16.

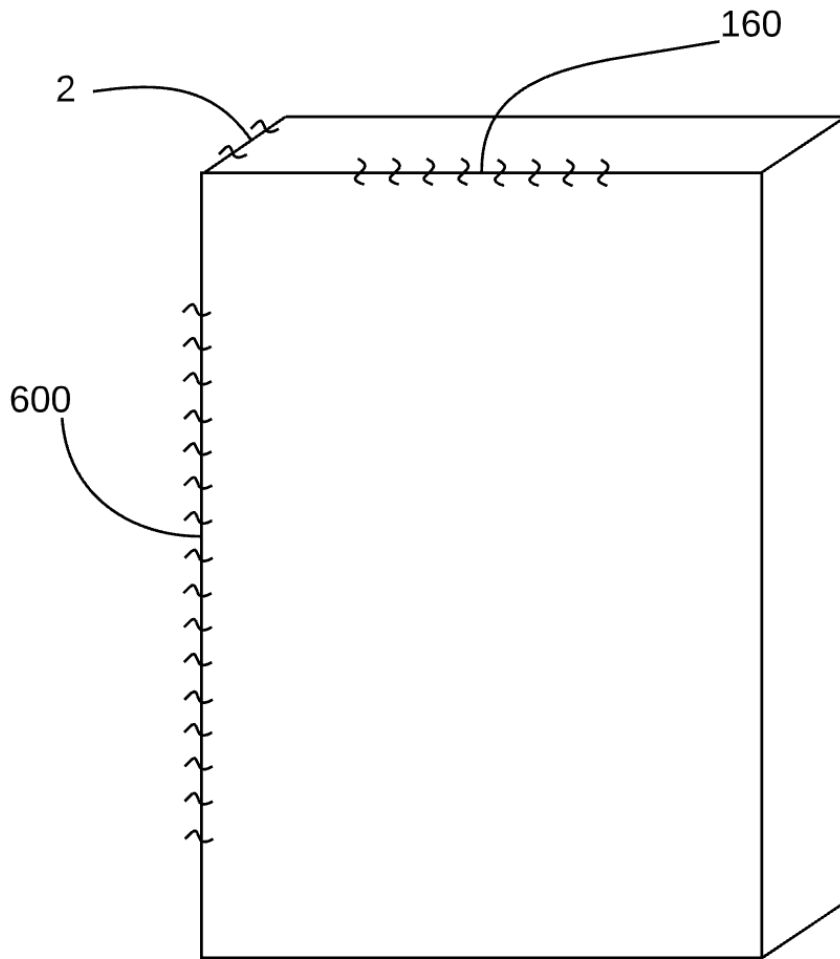


Figure 16. Number of divisions in each side.

Each side of the recuperator was modeled with a different number of divisions. In the longitudinal direction 600 was used, in the transverse direction 160 and the thickness was modeled with 2 elements.

In Fig. 17 shows how the actual elements look like when we meshing our model. Each yellow dot represents an element.

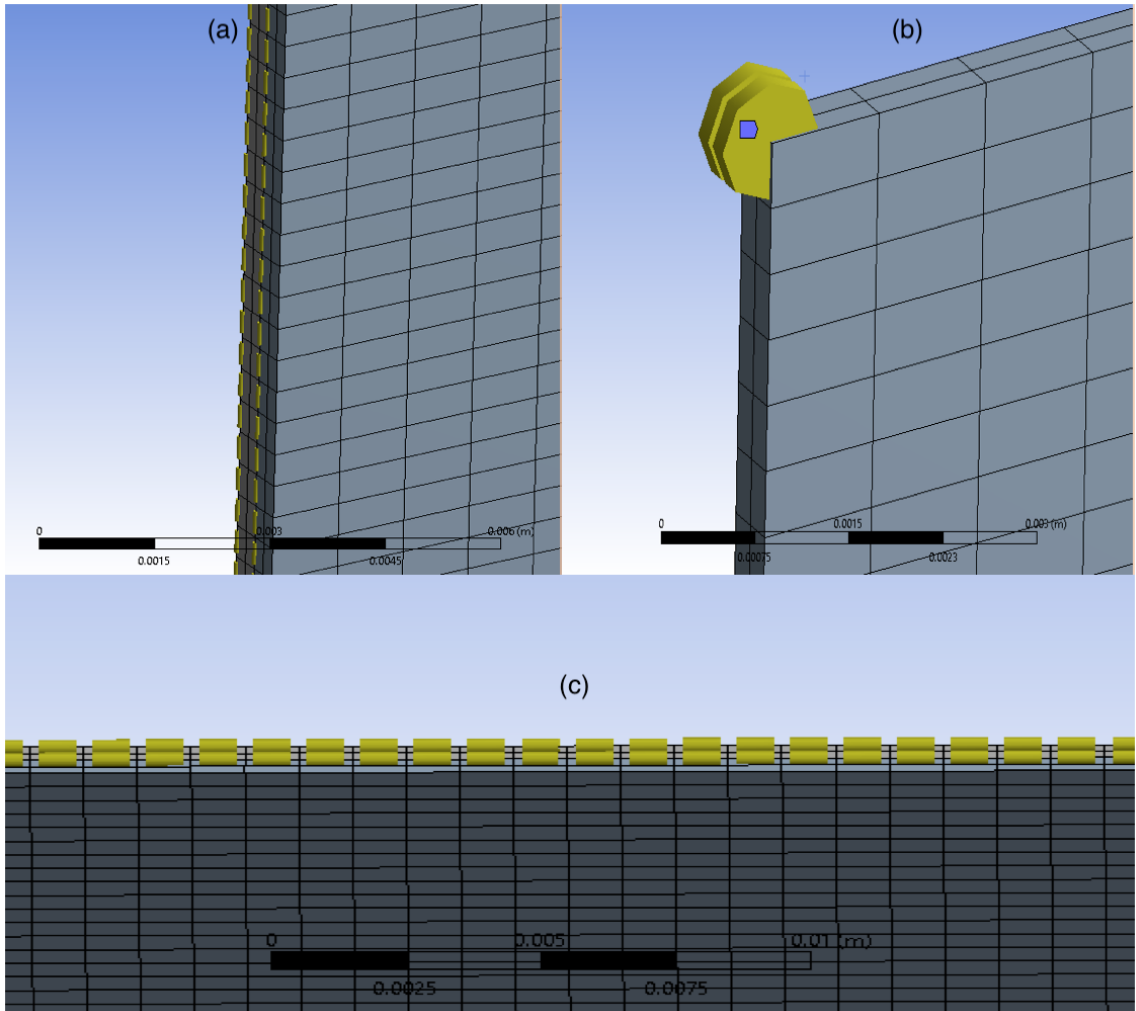


Figure 17. Number of divisions in every direction.

(a) *Longitudinal direction* (b) *Thickness* (c) *Transverse direction*

4.3 Properties of the materials.

As it is explained in chapter 2, the working fluid of a Superfluid Stirling Refrigerant is mixture liquid $^3\text{He} - ^4\text{He}$. This mixture is suitable due its superfluidity in these low temperatures (below 2 Kelvin) and the P-V-T relationship of the osmotic pressure. In this thesis, due the complexity of this mixture, the λ -point of ^3He being 0,001 and considering this is the initial case we will use as a working fluid ideal-gas ^3He . This consideration will simplify our model in many ways but still provide validation in our result. First of all, the geometry will be simplified once there will be no SuperLeaks in the pistons surfaces. Secondly and most importantly the materials properties will be simplified ending up in less time-consuming solution that with the current resources is much valuable. The material for the Solid Surfaces will be Kapton as described in chapter 2 too. Kapton's properties in low temperatures make it suitable for this simulation.

In later and more complex cases, the working fluid will be modared along with other variables for comparison the accuracy with the current results. At the last chapter of this thesis, information will be given about research that has been done in the field of material – mixture and ways of modeling pistons' superleaks.

In ANSYS Workbench – Fluent window, from the information Tree on the left side of the screen we can modify the working materials. The materials will be:

Material Type: Fluid

Density (kg/m³): Ideal-gas (Calculated in each timestep)

Specific Heat (j/kg-k): 5193

Thermal Conductivity (w/m-k): 0.152

Viscosity (kg/m-s): 1.99e-05

Molecular Weight (kg/kmol): 4.0026

All the properties will remain constant during the simulation except the Density that will be calculated in every time-step from the following equation.

Ideal Gas Law for compressible Flows

$$\text{Density: } \rho = \frac{p_{op} + p}{\frac{R}{M\omega}T} \quad (31)$$

p : local relative (or gauge) pressure predicted by ANSYS Fluent.

p_{op} : operating pressure

The input of the operating pressure is of great importance when you are computing density with the ideal gas law. Equation 25 notes that the operating pressure is added to the relative pressure field computed by the solver, yielding the absolute static pressure. By default, Operating pressure is set to 101325 Pa.

The properties for the material used in Solid domains can easily be accessed in the web and there are:

Material Type: Solid

Density (kg/m³): 1420

Specific Heat (j/kg-k): 1090

Thermal Conductivity (w/m-k): 0.12

The Fig. 18, Fig. 19 and Fig. 20 shows illustrates how to access and edit the materials in ANSYS Fluent interface.

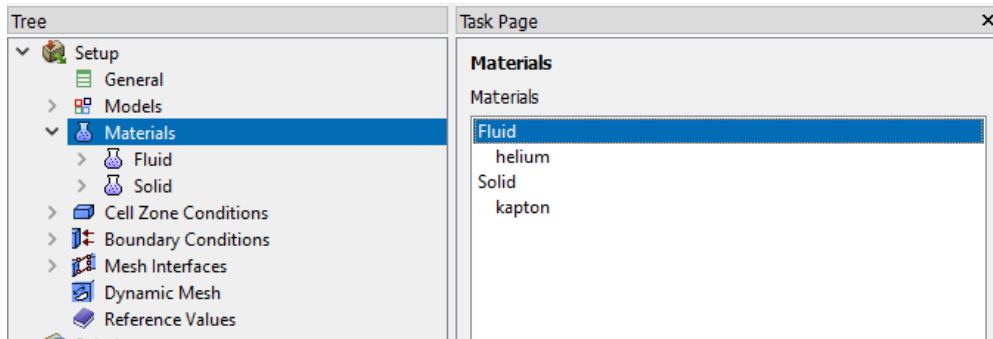


Figure 18. Editing materials from ANSYS Fluent Tree menu.

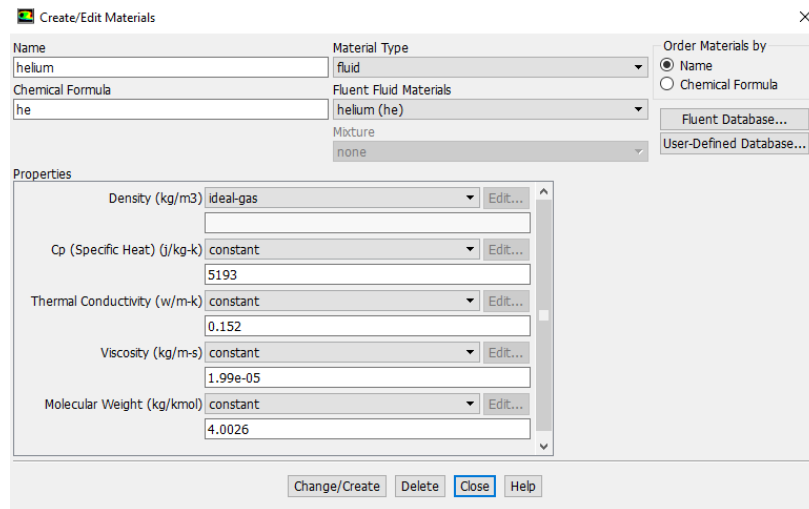


Figure 19. Editing fluid material in ANSYS Fluent.

He³ properties were simulated as an ideal-gas for simplicity but accuracy results at the same time.

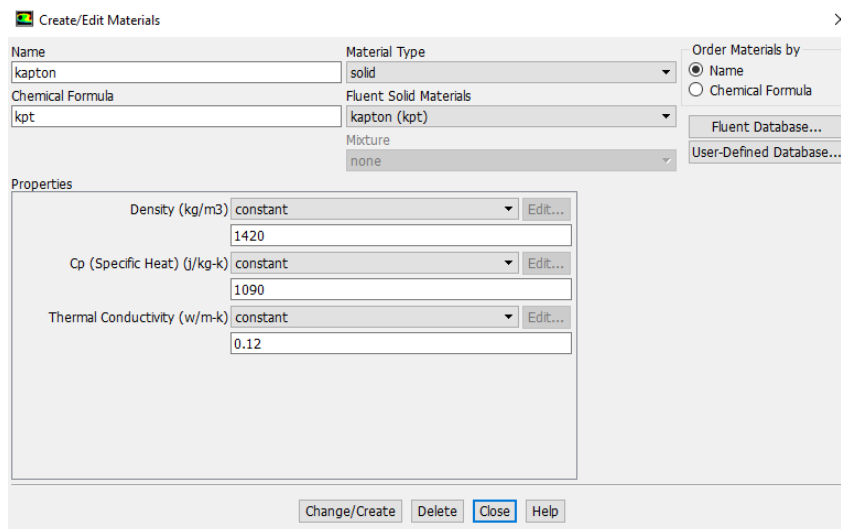


Figure 20. Editing Solid material in ANSYS Fluent.

Kapton is suitable due to its heat - properties at low temperatures.

4.4 Boundaries.

For the solution of any computational fluid dynamics problem, initial and boundary conditions need to be specified. Boundary conditions are essential component of a mathematical model. They direct the motion of flow which leads to a unique solution. A boundary value problem is actually a differential equation (or system of differential equations) to be solved in a domain on whose boundary a set of conditions is known. It is opposed to the “initial value problem”, in which only the conditions on one extreme of the interval are known. Boundary value problems are extremely important as they model a vast amount of phenomena and applications, from solid mechanics to heat transfer, from fluid mechanics to acoustic diffusion. They rise naturally in every problem based on a differential equation to be solved in space, while initial value problems usually refer to problems to be solved in time. These types of boundary conditions are frequently used. In CFD softwares, we are familiar with standard boundary conditions such as inlet, outlet, wall, symmetry etc. But there are certain applications like the certain application which require custom boundary conditions which are bit different from these standard boundary conditions due that the working fluid is recirculated in the domains of interest as a result domain of inlet or outlet won't be specified and the solution will accrue from the initial conditions. This is also the reason why the initial positions of the pistons must be completely specified in every run.

In this chapter, the boundaries of all the surfaces will be detailed explained as long with the Mesh Interfaces and Contacts tool that will determine the flow of the working fluid.

4.4.1 Mesh Interfaces.

As explained previously, in the Superfluid Sterling Refrigerant the working fluid is recirculated inside the domains of interest. The domains of interest have already been named and set through the geometry and mesh properties. In this section, with the contact tool the flow of the working fluid will be determined. First of all and most importantly, in order the fluid to be able to flow through the domains, the domains must remain in contact.

We will analyze and present in figures the flow from the compressor's cylinder until the recuperator for one piston considering that it is completely the same for the remain three pistons. Initially, the working fluid starts from the Compressor's cylinder as it is pushed to the cooler's chamber. As shown in Fig. 21 the Compressor's cylinder's face (RED) is in contact with the Cooler's face (BLUE). These two separate faces will be set as Mesh Interface with the Contact tool and the working fluid will freely flow from the one domain to another frictionless.

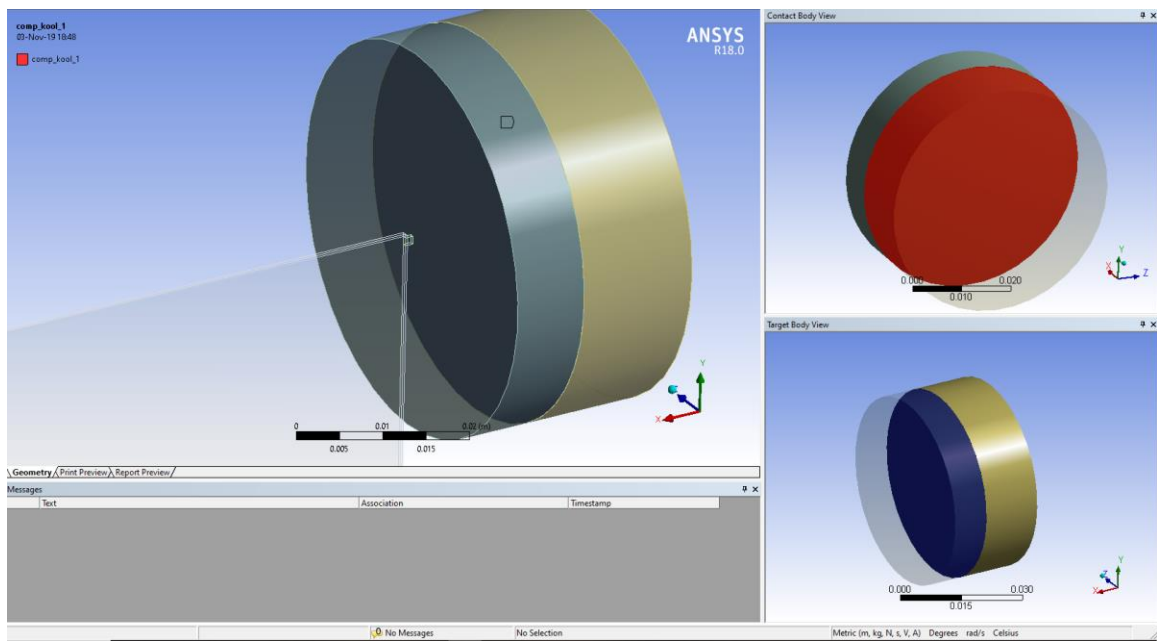


Figure 21. Mesh Contact (Compressor – Cooler).

Every time the working fluid in flowing through different domains, the two surfaces that are in contact will set as Interfaces.

After the first Mesh Interface is set, the working fluid will be directed from the cooler chamber to the recuperator. As a result, two Interfaces will be needed. The first one is from the cooler to the connection tube and the second from the connection tube to the recuperator. As shown in Fig. 22 the Cooler surface (BLUE) that is contact with the connection tube's surface (RED) will be set as interface with the same way as previously.

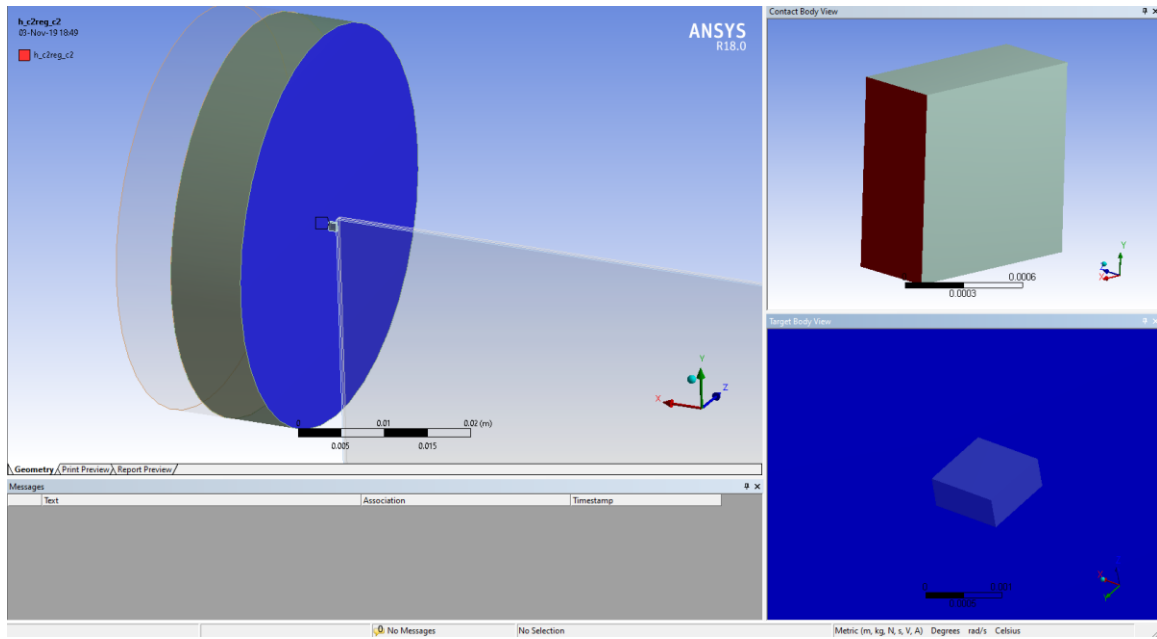


Figure 22. Mesh Contact (Cooler – Connection Tube)

A second Interface is needed here because after the flow reaches the Cooler, it will have to flow through the Connection Tube in order to enter the Recuperator.

Finally, as shown in Fig. 23 another Interface will be set between the Connection Tube and the Recuperator's face as the flow will finish the procedure and enter the domain. After the flow enters, it will exit and follow the same flow motif but in inverse. So for all the cylinders the following same three Mesh Contacts will be used.

- Compressor with Cooler.
- Cooler with Connection Tube.
- Connection Tube with Recuperator.

It's the same for the expander's piston but instead of Compressor it will be Expander and instead of Cooler it will be Heater resulting to the creation of twelve Mesh Interfaces.

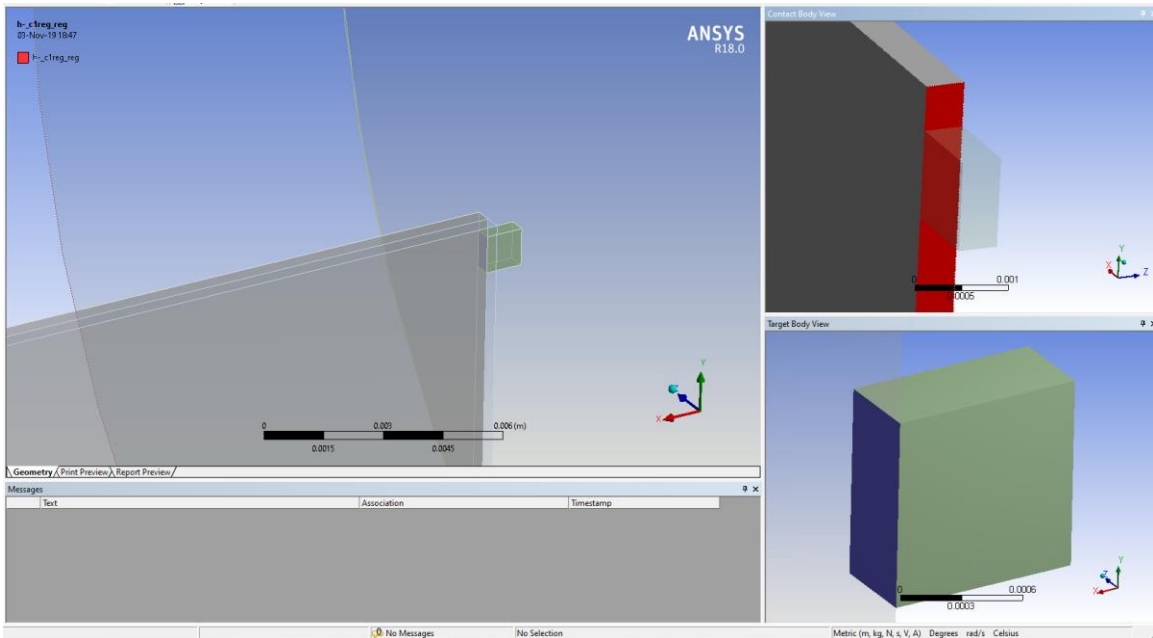


Figure 23. Mesh Contact (Connection Tube – Recuperator)

Finally, the flow enter the recuperator from the Mesh Interface between the Connection Tube and the Recuperator.

The last but most important interface that will be used, is between the two recuperators. As it was explained in detail, the two recuperators of the two Sterling Refrigerants exchange heat, as the working fluid flows inside, that depends from the out of phase operation. The two recuperators are in contact as shown in Fig. 24 and the purpose of this Interface is that every time the cold surface with completely absorb the heat without any energy losses. So, the heat will freely flow throught and the recuperator cold recuperator each time will operate as a regenerator. To achieve this with the Contact Tool we will set an interface surface and later in the ANSYS Fluent interface we will set the two surfaces as Coupled walls.

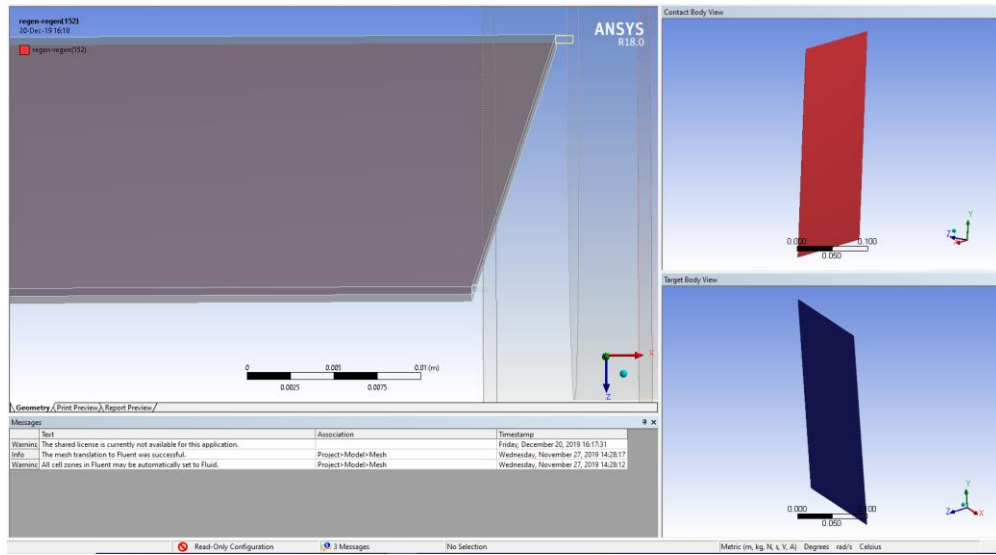


Figure 24. Mesh Contact between the two Recuperators.

A Mesh Interface was set between the two recuperators so it can be used as coupled walls resulting to zero loss Heat absorption between them.

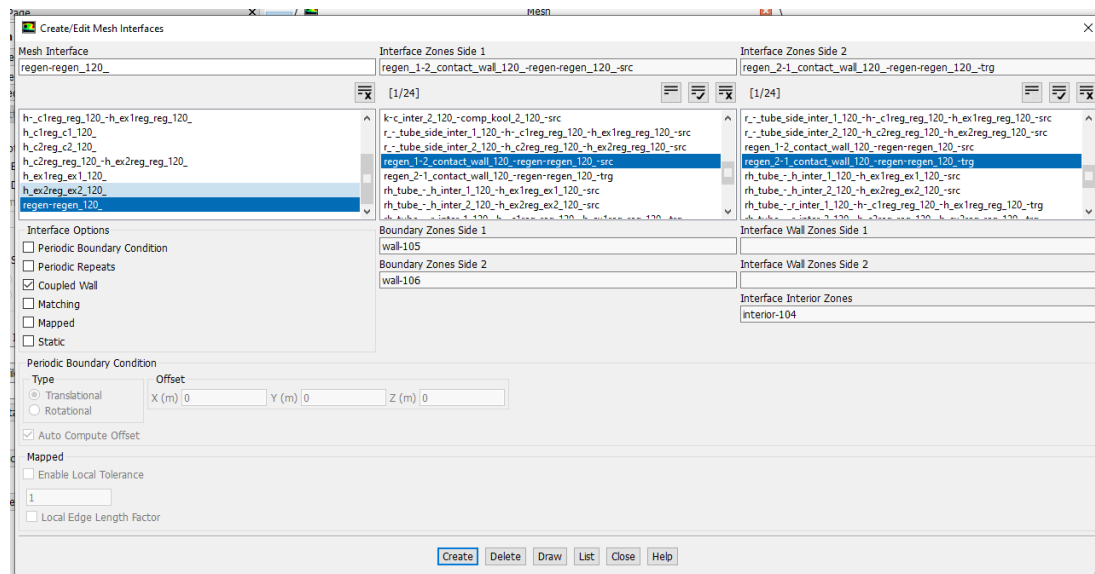


Figure 25. Coupled Wall.

The two recuperators walls was set as Coupled Wall under the Thermal Conditions in ANSYS Fluent solver.

When the option Coupled Wall is selected under the Thermal Conditions box in Fig. 25 a "shadow" zone will automatically be created so that each side of the wall is a distinct wall zone. In the Wall dialog box, the shadow zone's name will be shown in the Shadow Face Zone field. You can choose to specify different thermal conditions on each zone.

4.4.2 Wall Conditions.

In the boundaries conditions, all walls are set to be Adiabatic Walls so thermal isolation from the surrounding will be achieved.

The only walls that differ are the recuperator walls that described previously and the hot and cold reservoir. As was analyzed during the cycle explanation in chapter 3, the heat will be absorbed from the cold reservoir and rejected at the hot reservoir. So, for these two interactions cold and hot reservoir wall's properties will be edited. Initially, as shown in Fig. 26, stationary wall and no slip options are selected.

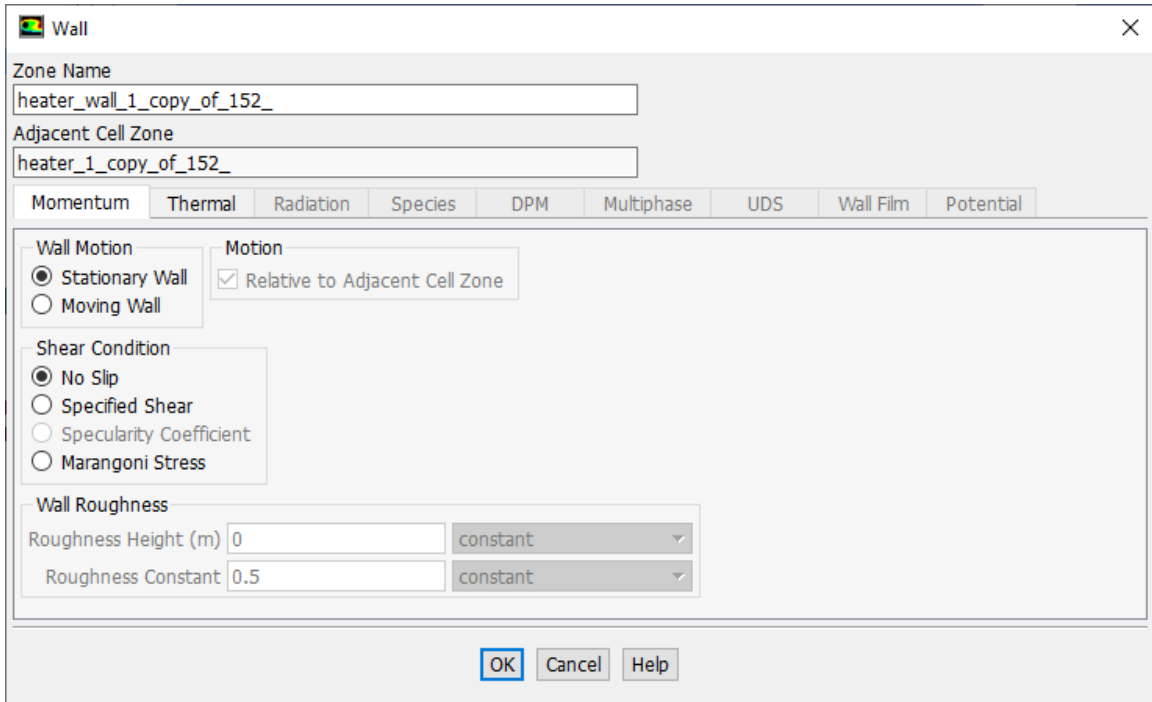


Figure 26. Reservoir wall's momentum properties.

There will be no movement so stationary wall and no slip options are selected. For both hot and cold reservoir momentum options are the same.

The stationary wall indicates that there will be no wall movement. The no-slip condition is the default, and it indicates that the fluid sticks to the wall and moves with the same velocity as the wall, if it is moving. The specified shear and Marangoni stress boundary conditions are useful in modeling situations in which the shear stress (rather than the motion of the fluid) is known. Examples of such situations are applied shear stress, slip wall (zero shear stress), and free surface conditions (zero shear stress or shear stress dependent on surface tension gradient). The specified shear boundary condition allows you to specify the x, y, and z components of the shear stress as constant values or profiles.

Secondly and most important, as shown in Fig. 27 for heat transfer Convection option will be selected but all the other basic options will be explained too.

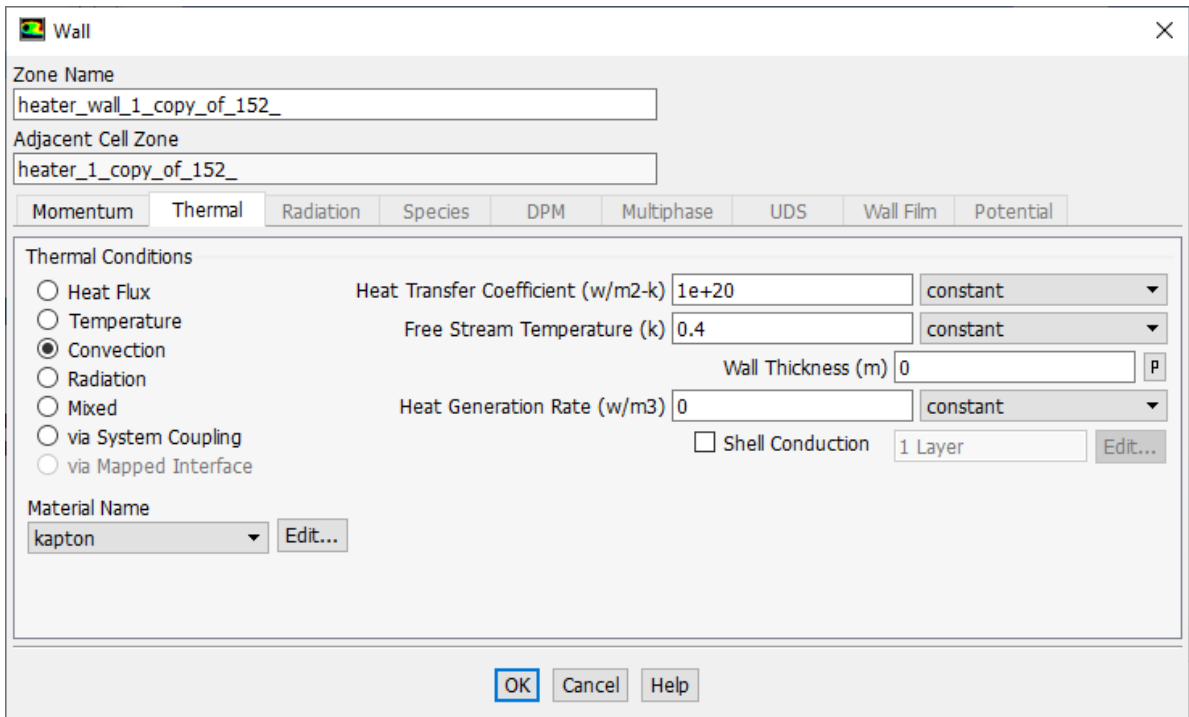


Figure 27. Hot reservoir wall's Thermal properties.

Convection option is suitable for these types of applications.

Heat Flux boundary condition option is suitable for a fixed heat flux condition. Then a value for the heat flux at the wall surface is needed to be set in the Heat Flux field. An adiabatic wall can be defined by setting a zero heat flux conditions. This is the default condition for all walls.

Temperature boundary condition option is suitable for a fixed temperature condition in the wall. After the wall's temperature is specified in the dialog box, the heat transfer to the wall is computed using Eq. 32 or Eq. 33.

$$\text{Heat: } q = h_f(T_w - T_f) + q_{rad} \quad (32)$$

where

$h_f = \text{fluid - side local heat transfer coefficient}$

$T_w = \text{wall surface temperature}$

$T_f = \text{local fluid temperature}$

$T_f = \text{local fluid temperature}$

$q_{rad} = \text{radiative heat flux}$

Convective Heat Transfer Boundary Condition options is suitable when a heat transfer coefficient boundary condition at a wall is specified. ANSYS FLUENT uses the inputs of the external heat transfer coefficient and external heat sink temperatures to compute the heat flux to the wall as shown in Eq. 32 and Eq. 33

$$\begin{aligned} \text{Heat: } q &= h_f(T_w - T_f) + q_{rad} \\ q &= h_{ext}(T_{ext} - T_w) \end{aligned} \quad (33)$$

where

$h_{ext} = \text{external heat coefficient}$

$T_{ext} = \text{external heat - sink temperature}$

$q_{rad} = \text{radiative heat flux}$

External Radiation Boundary Conditions option is suitable when you are interested in radiation heat transfer from the exterior of your model.

Combined Convection and External Radiation Boundary Conditions options is thermal condition that combines the convection and radiation boundary conditions.

As a result, the most suitable method is the Convection Heat Transfer Boundary option by setting a very high number for the Heat Transfer Coefficient ($\text{w/m}^2\text{-k}$) and the Free Stream Temperature (k) will be set equal to the lower and maximum temperature value for the hot and cold reservoir respectively, as shown in Fig. 32 and Fig. 33.

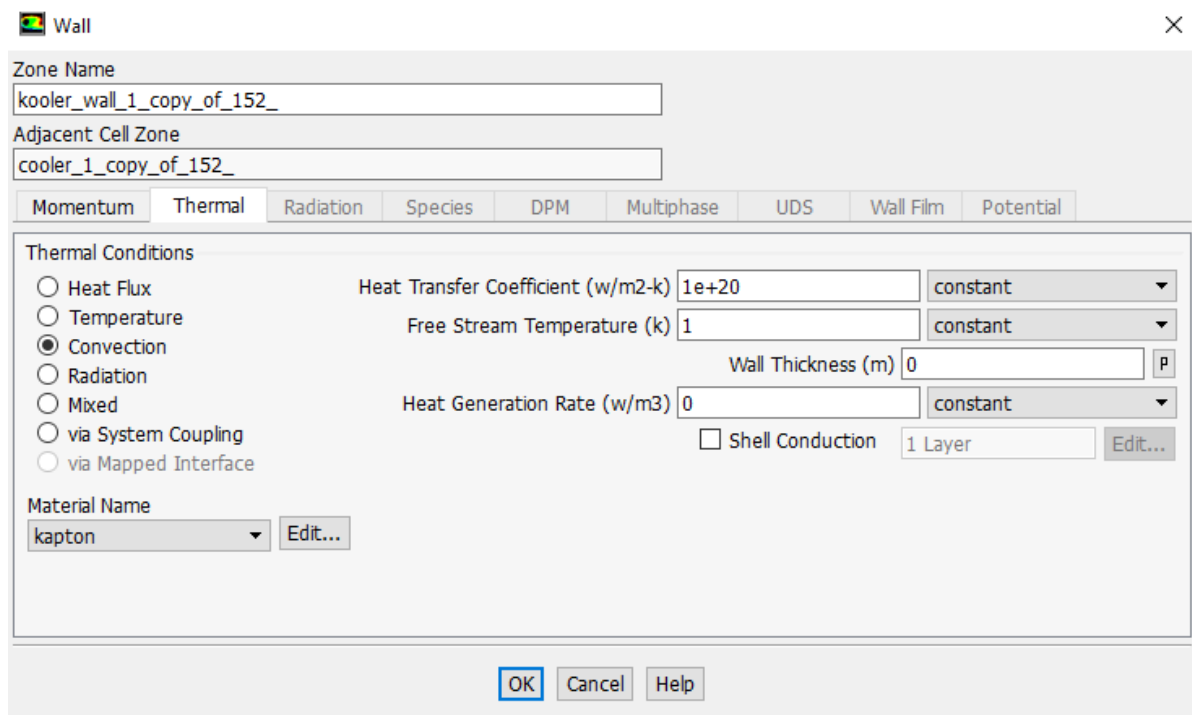


Figure 28. Cold reservoir wall's Thermal properties.

Convection option is suitable for these types of applications.

4.5 User - Defined Functions (UDF)

User defined functions (UDF) allows user to modify the behavior of ANSYS Fluent to satisfy his particular modeling needs. For example, these modifications may be used to impose desired initial and boundary, material models as well as various physical and chemical transformations such as heat transfer, chemical reactions or phase change. UDFs are flexible and powerful, and allow user to use ANSYS Fluent as a framework to implement new models. In later section UDFs will explained in more details as explained in the ANSYS FLUENT UDF manual release 15. [7]

4.5.1 What is a UDF?

A user-defined function, or UDF, is a C function that can be dynamically loaded with the ANSYS Fluent solver to enhance its standard features. For example, user can use a UDF to:

- Customize boundary conditions, material property definitions, surface and volume reaction rates, source terms in ANSYS Fluent transport equations, source terms in user-defined scalar (UDS) transport equations, diffusivity functions, and so on.
- Adjust computed values on a once-per-iteration basis.
- Initialize a solution.
- Perform asynchronous (on demand) execution of a UDF.
- Execute at the end of an iteration, upon exit from ANSYS Fluent, or upon loading of a compiled UDF library.
- Enhance postprocessing.
- Enhance existing ANSYS Fluent models (such as discrete phase model, multiphase mixture model, discrete ordinates radiation model).

UDFs are identified by a.c extension (for example, myudf.c). One source file can contain a single UDF or multiple UDFs, and you can define multiple source files. UDFs are defined using DEFINE macros provided by ANSYS Fluent. They are coded using additional macros and functions (also supplied by ANSYS Fluent) that access ANSYS Fluent solver

data and perform other tasks. Every UDF must contain the `udf.h` file inclusion directive (`#include "udf.h"`) at the beginning of the source code file, which enables both the definition of `DEFINE` macros and other ANSYS Fluent- provided macros and functions, and their inclusion in the compilation process.

Source files containing UDFs can be either interpreted or compiled in ANSYS Fluent.

- For interpreted UDFs, source files are interpreted and loaded directly at runtime in a single-step process.
- For compiled UDFs, the process involves two separate steps. A shared object code library is first built, then loaded, into ANSYS Fluent.

After being interpreted or compiled, UDFs will become visible and selectable in ANSYS Fluent dialog boxes, and can be hooked to a solver by choosing the function name in the appropriate dialog box.

In summary, UDFs:

- Are written in the C programming language.
- Must be defined using `DEFINE` macros supplied by ANSYS Fluent.
- Must have an include statement for the `udf.h` file.
- Use predefined macros and functions to access ANSYS Fluent solver data and to perform other tasks.
- Are executed as interpreted or compiled functions.
- Are hooked to an ANSYS Fluent solver using a graphical user interface dialog boxes.

4.5.2 Defining UDF with Macros

All UDFs are defined using ANSYS Fluent-supplied function declarations. These function declarations are implemented in the code as macros, and are referred to in this

document as DEFINE (all capitals) macros. Definitions for DEFINE macros are contained in the udf.h header file.

The general format of a DEFINE macro is:

DEFINE_MACRONAME (udf_name, passed-in variables)

where the first argument in the parentheses is the name of the UDF that user supplies. Name arguments are case-sensitive and must be specified in lowercase. After the function has been interpreted or compiled, the name that you choose for your UDF will become visible and selectable in drop-down lists in ANSYS Fluent. The second set of input arguments to the DEFINE macro are variables that are passed into your function from the ANSYS Fluent solver.

For example, the macro:

DEFINE_PROFILE(inlet_x_velocity, thread, index)

defines a boundary profile function named inlet x velocity with two variables, thread and index, that are passed into the function from ANSYS Fluent. These passed-in variables are the boundary condition zone ID (as a pointer to the thread) and the index identifying the variable that is to be stored. After the UDF has been interpreted or compiled, its name (inlet_x_velocity) will become visible and selectable in drop-down lists in the appropriate boundary condition dialog box (for example, Velocity Inlet) in ANSYS Fluent.

4.5.3 Including the udf.h Header File in the Source File.

The udf.h header file contains:

- Definitions for DEFINE macros
- #include compiler directives for C library function header files
- Header files (for example, mem.h) for other ANSYS Fluent-supplied macros and functions.

You must, therefore, include the `udf.h` file at the beginning of every UDF source code file using the `#include` compiler directive:

```
#include "udf.h"
```

For example, when `udf.h` is included in the source file containing the `DEFINE` statement from the previous section,

```
#include "udf.h"
DEFINE_PROFILE(inlet_x_velocity, thread, index)
```

upon compilation, the macro will expand to

```
void inlet_x_velocity (Thread *thread, int index)
```

4.5.4 Interpreting and Compiling UDFs

Source code files containing UDFs can be either interpreted or compiled in ANSYS Fluent. In both cases the functions are compiled, but the way in which the source code is compiled and the code that results from the compilation process is different for the two methods. These differences are explained in the following sections.

4.5.5 Compiled UDFs

Compiled UDFs are built in the same way that the ANSYS Fluent executable itself is built: a Makefile script is used to invoke the system C compiler to build an object code library. User initiates this action in the Compiled UDFs dialog box by clicking Build. The object code library contains the native machine language translation of your higher-level C source code. The shared library must then be loaded into ANSYS Fluent at runtime by a process called "dynamic loading". User initiates this action in the Compiled UDFs dialog box by clicking Load. The object libraries are specific to the computer architecture being used, as well as to the particular version of the ANSYS Fluent executable being run. The libraries must, therefore, be rebuilt any time ANSYS Fluent is upgraded, when the computer's operating system level changes, or when the job is run on a different type of computer.

In summary, compiled UDFs are compiled from source files using the graphical user interface, in a two- step process. The process involves the Compiled UDFs dialog box, where you first build a shared library object file from a source file, and then load the shared library that was just built into ANSYS Fluent.

4.5.6 Interpreted UDFs

Interpreted UDFs are interpreted from source files using the graphical user interface, but in a single- step process. The process, which occurs at *runtime*, involves using the Interpreted UDFs dialog box, where you Interpret a source file.

Inside ANSYS Fluent, the source code is compiled into an intermediate, architecture-independent machine code using a C preprocessor. This machine code then executes on an internal emulator, or interpreter, when the UDF is invoked. This extra layer of code incurs a performance penalty, but enables an inter- preted UDF to be shared effortlessly between different architectures, operating systems, and ANSYS Fluent versions. If execution speed does become an issue, an interpreted UDF can always be run in compiled mode without modification.

The interpreter that is used for interpreted UDFs does not have all of the capabilities of a standard C compiler (which is used for compiled UDFs). Specifically, interpreted UDFs *cannot* contain any of the following C programming language elements:

- gotostatements
- Non-ANSI-C prototypes for syntax
- Direct data structure references
- Declarations of local structures
- Unions
- Pointers to functions
- Arrays of functions
- Multi-dimensional arrays

4.5.7 Differences Between Interpreted and Compiled UDFs.

The major difference between interpreted and compiled UDFs is that interpreted UDFs cannot access ANSYS Fluent solver data using direct structure references; they can only indirectly access data through the use of ANSYS Fluent-supplied macros. This can be significant if, for example, you want to introduce new data structures in your UDF.

Here is a summary of the differences between interpreted and compiled UDFs:

Interpreted UDFs:

- Are portable to other platforms
- Can all be run as compiled UDFs
- Do not require a C compiler
- Are slower than compiled UDFs
- Are restricted in the use of the C programming language
- Cannot be linked to compiled system or user libraries
- Can access data stored in an ANSYS Fluent structure only using a predefined macro.

Compiled UDFs:

- Execute faster than interpreted UDFs
- Are not restricted in their use of the C programming language
- Can call functions written in other languages (specifics are system- and compiler-dependent)
- Cannot necessarily be run as interpreted UDFs if they contain certain elements of the C language that the interpreter cannot handle.

Thus, when deciding which type of UDF to use for your ANSYS Fluent model:

- Use interpreted UDFs for small, straightforward functions
- Use compiled UDFs for complex functions that:

Have a significant CPU requirement (for example, a property UDF that is called on a per-cell basis every iteration) and require access to a shared library

4.5.8 Creating the UDF

Considering the above information, the UDFs for the pistons' movements were created as shown in Fig. 28. Initially the compiler directives for the libraries were include with the command 'include' and after the macros were defined. The specific UDF will return the velocity of the pistons in every time step. The passed in variables will be the time and from the generated equations for the volume the velocity will be calculated as shown in Fig. 28, when operating 180° out of phase. Note that dynamic mesh UDFs are defined by using DEFINE_CG_MOTION.

```
#include "udf.h"
#include "math.h"
DEFINE_CG_MOTION(piston_comp_1_new,dt,vel,omega,time,dttime)
{
    real pi,w,hz;
    pi=3.141592653589793238462643383279;
    hz=1;
    w=2*pi*hz;

    real Vswc,Apc;
    Vswc=17.74*pow(10,-6);
    Apc=17.74*pow(10,-4);

    real Ve;
    Ve=(Vswc*w/(2*Apc))*sin(w*time);
;

    vel[0]=Ve;
    vel[1]=0;
    vel[2]=0;
}
```

Figure 28. User-Defined Function for Compressor – SR1.

This UDF has no initial phase so it will be used for the SR1 compressor's piston movement.

With the same way, the UDFs for the SR2 compressor's and both expanders' pistons were created with the only difference that the final velocity equation will have an initial phase as shown in Fig. 29, Fig. 30 and Fig. 31.

```
#include "udf.h"
#include "math.h"
DEFINE_CG_MOTION(piston_exp_1_new,dt,vel,omega,time,dtime)
{
    real pi,w,hz,T;
    pi=3.141592653589793238462643383279;
    hz=1;
    T=1/hz;
    w=2*pi*hz;

    real Vswe,Ape;
    Vswe=9.39*pow(10,-6);
    Ape=9.39*pow(10,-4);

    real Ve;
    Ve=(Vswe*w/(2*Ape))*sin(w*(time+T/4));
;

    vel[0]=Ve;
    vel[1]=0;
    vel[2]=0;

}

```

Figure 29. User-Defined Function for Expander – SR1.

Velocity equation has an initial phase of $T/4$ as described in Ch. 4.2.2 “Cycle explanation”. Considering also that the period is 1 Hz that equals to an operation $\pi/4$ out of phase from the compressor's piston

At the end of every UDF the vel[..] command indicated in which direction the velocity will be given. Every vel[1], vel[2], vel[3] indicates the x,y,z direction respectively. Based on the geometry design in this particular model, the suitable direction is among the x axis.

```
#include "udf.h"
#include "math.h"
DEFINE_CG_MOTION(piston_comp_2_new,dt,vel,omega,time,dtime)
{
    real pi,w,hz,T;
    pi=3.141592653589793238462643383279;
    hz=1;
    w=2*pi*hz;
    T=1/hz;

    real Vswc,Apc;
    Vswc=17.74*pow(10,-6);
    Apc=17.74*pow(10,-4);

    real Ve;
    Ve=-(Vswc*w/(2*Apc))*sin(w*(time+2*T/4));
;

    vel[0]=Ve;
    vel[1]=0;
    vel[2]=0;

}
```

Figure 30. User-Defined Function for Compressor – SR2.

The vel[..] indicates the direction that the calculated velocity will be given after each time-step.

The UDFs that used to calculate the velocity for the SR2 have an (-) in the velocity equations in order to specify the correct initial direction. When the SSR will operate in a different phase, only the UDFs of the SR2 will be modified. The SR1's UDFs will remain the same because only the second machine will change its initial condition so the specific out of phase time will be achieved.

```
#include "udf.h"
#include "math.h"
DEFINE_CG_MOTION(piston_exp_2_new,dt,vel,omega,time,dtime)
{
    real pi,w,hz,T;
    pi=3.141592653589793238462643383279;
    hz=1;
    T=1/hz;
    w=2*pi*hz;

    real Vswe,Ape;
    Vswe=9.39*pow(10,-6);
    Ape=9.39*pow(10,-4);

    real Ve;
    Ve=-(Vswe*w/(2*Ape))*sin(w*(time+3*T/4));
;

    vel[0]=Ve;
    vel[1]=0;
    vel[2]=0;

}
}
```

Figure 31. User-Defined Function for Expander – SR2.

Every time the SSR will operate in different out of phase, only the SR2 initial conditions and UDFs will be modified.

4.5.9 Import and Compile the UDF

After the UDF is created, it must be imported and compiled in ANSYS FLUENT. The general procedure for compiling a UDF source file and building a shared library for the resulting objects, and loading the compiled UDF library into ANSYS FLUENT using the graphical user interface (GUI) is as shown in Fig. 32. Note that the UDF source file you want to compile is in the same folder that contains your case and data files.

Initially, we start ANSYS FLUENT by using FLUENT Launcher and open the Compiled UDFs dialog box. In the Compiled UDFs dialog box we click on Add... under Source Files to select the UDF source file (or files) you want to compile as shown in Fig 32. This will open the Select File dialog box. In the Select File dialog box, we click the names of all of the desired files (e.g., udfexample.c), so that the complete paths to the source files are displayed under Source File(s). We can remove a selection by clicking the path in Source File(s) list and then clicking the Remove button. Click OK when your selections are complete. In the Compiled UDFs dialog box, we enter the name of the shared library we want to build in the Library Name field (or we leave the default name libudf), and click Build. All of the UDFs that are contained within each C source file we selected will be compiled and the build files will be stored in the shared library you specified.

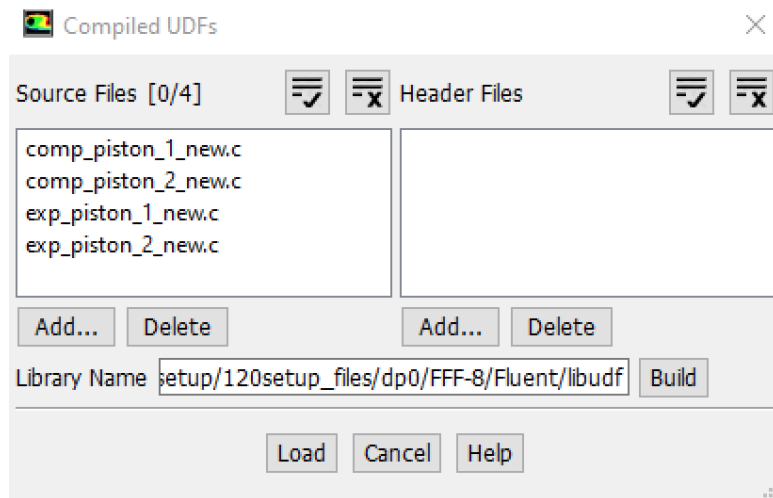


Figure 32. Compiling the UDFs.

After the creation of the UDFs, they must be imported compiled and build.

When we compile the UDFs it is important to check in the error dialog box that the UDFs were built and the path was successfully created as shown in Fig 33.

```
Creating library libudf.lib and object libudf.exp
Creating user_nt.udf file for 3ddp_node ...
(system "copy "C:\PROGRAMS\ANSYS\11\fluent18.0\src\udf\makefile_nt.udf "C:\Users\
1 file(s) copied.
(chdir "C:\Users\...dp0\FFF-10\Fluent\libudf") (chdir "win64\3ddp_node")# Generating
comp_piston_1_new.c
comp_piston_2_new.c
exp_piston_1_new.c
exp_piston_2_new.c
Generating Code...
# Generating udf_names.c because of makefile comp_piston_1_new.obj comp_piston_2_new.obj exp_piston_1_new.obj exp_piston_2_new.obj
udf_names.c
# Linking libudf.dll because of makefile user_nt.udf udf_names.obj comp_piston_1_new.obj comp_piston_2_new.obj exp_piston_1_new.obj e
Microsoft (R) Incremental Linker Version 14.00.23026.0
Copyright (C) Microsoft Corporation. All rights reserved.

Creating library libudf.lib and object libudf.exp

Done.
```

Figure 33. Message in the dialog box after the UDFs has successfully been compiled.

The UDFs are compiled and Libudf folder was successfully created inside the Fluent folder with no error messages.

4.6 Dynamic Mesh

After UDFs have been compiled, they are ready to be used in order to control the behavior of a dynamic mesh. In more details, the compiled UDFs will be used to define the motion of all the pistons. But despite they will be moving towards the direction that will be specified from the UDFs in each time step, they will be under no deformation. As a result, the option Rigid Body will be used for all the pistons. On the other hand, the pistons' cylinders will be deformed accordingly to the motion of the piston. So, the suitable option will be Deforming Body. Both options are shown in the Fig. 33 and Fig. 34 that represents the Dynamic Mesh Panel. In summary, the pistons will be directed by the UDFs and stay undeformed and the cylinders will be deformed in each time-step accordingly the position and the motion of the pistons.

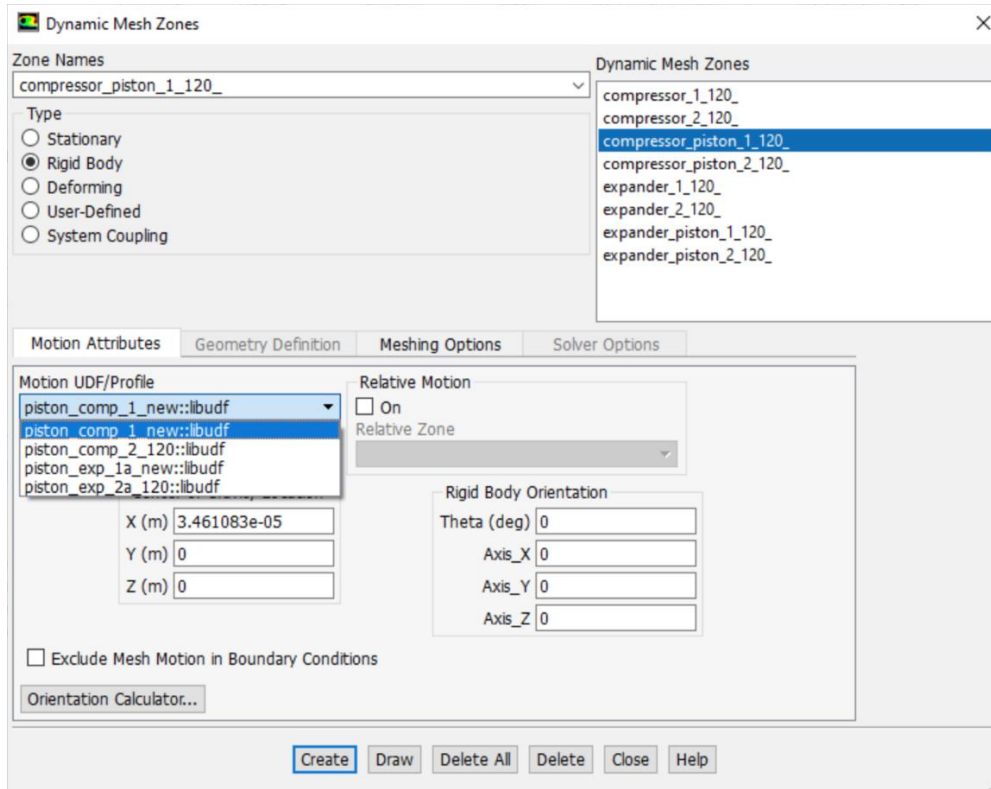


Figure 34. Setting Dynamic Mesh for the Piston.

The same options will be used for all the pistons with the only difference that every piston will be connected with the appropriate UDF.

Considering how time consuming is the calculations, a mesh motion preview it's necessary because a wrong direction for example it could end up to hundreds of time-steps until it will be visible. So, when the mesh update methods and their associated parameters have been specified, and the motion of dynamic zones have been defined, we can preview the motion of the mesh or the zone as it changes with time before starting the simulation. The same dynamic zone or mesh motion will be executed when we start your simulation. The zone motion preview only updates the graphical representation (in the graphics window) of the zones that we have selected using the Mesh Display dialog box. The zone motion preview will only update those zones that have solid body motion specified. So only the Pistons motion will be visible. On the other hand, the mesh motion preview is different from the zone motion described above in that the mesh connectivity is changed

in mesh motion. Note that the mesh motion will actually update the node locations as well as the connectivity of the mesh, so we must be sure to save our case file before doing the dynamic mesh motion. Once we have advanced the mesh by a certain number of time steps, we will not be able to recover the previous status of the mesh, other than by reloading the appropriate ANSYS FLUENT case file.

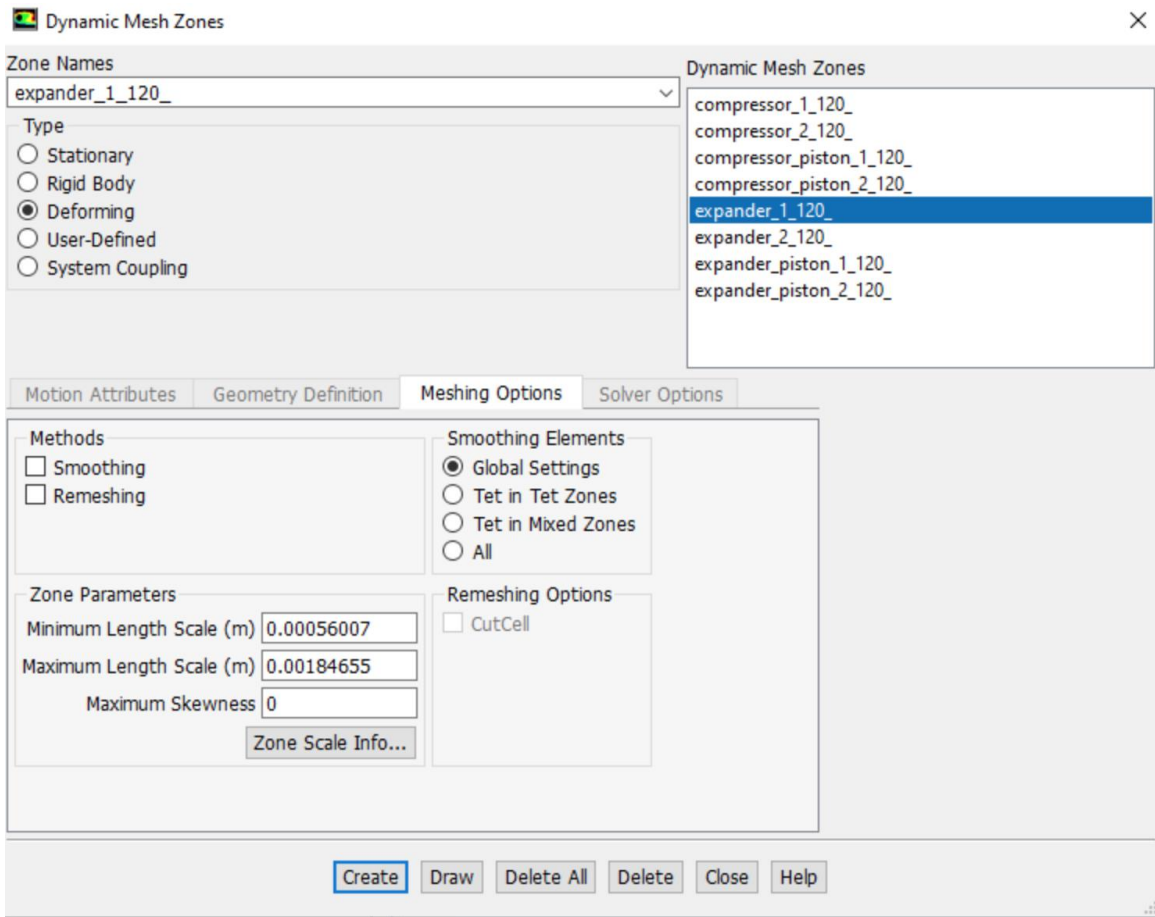


Figure 35. Setting Dynamic Mesh for the Cylinders.

Both compressors' and expanders' cylinders will be deformed accordingly to each piston's motion.

4.7 Setting up the Solution Parameters.

In this chapter, we will analyze the options that used to set up the solution. First of all, the Solver Type that was used and what method. After this, the initializations process will be explained as long with which time-step and how many iterations were selected. Last but not least, how the monitors were set up and which data we exported for later calculations.

4.7.1 Solver Type – Energy equations.

When setting up a solution, one of the most important options is to select the most suitable solver between Pressure – Based solver and Density – Based Solver. Both solvers are able to solve flow problems but both have their advantages and disadvantages. Usually the Pressure – Based solver is used when we simulate subsonic flows while the Density – Based solver is used when we simulate supersonic flows. Considering that the SSR will operate in approximately 1 Hz, the pressure based solver is more suitable.

In the Pressure – Based solver, the governing equations are solved sequentially. In other words, governing equation for the variables such as u, v, w, p, T are solved one after the another as shown in Fig. 35. This method is called SIMPLE method. But unlike the sequential – SIMPLE method, there is another method which is called COUPLED. In COUPLED Pressure – Based solver the momentum and the continuity equations are solved simultaneously. This means that the governing equations are solved in coupled manner and other scalar equations are solved in segregated manner. The solution convergence rate is greatly enhanced when compared to sequentially pressure algorithm but in terms of memory, the coupled need more memory requirement than sequential. The algorithm for coupled pressure based is represented in Fig. 36. Also in this simulation, we are interested of course in heat transfer. The way that ANSYS calculates the heat transfer is by solving the Energy equations as presented in page 79.

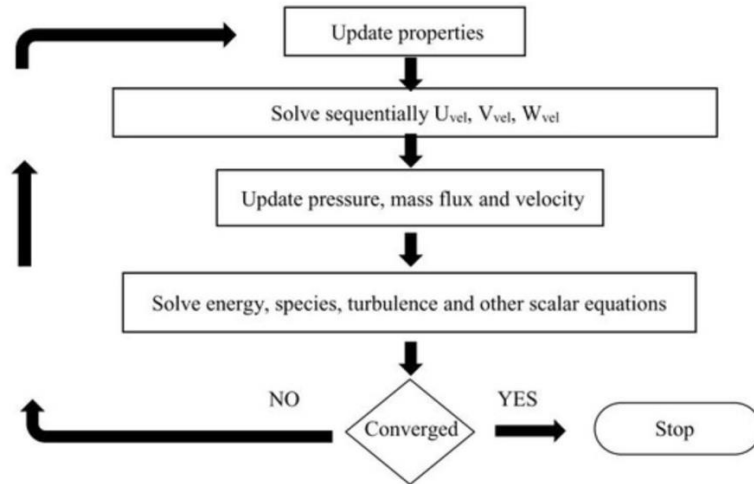


Figure 36. Pressure Based solver with SIMPLE method.

When using the SIMPLE method in Pressure – Based solver, all the governing equations are solved sequentially. [7]

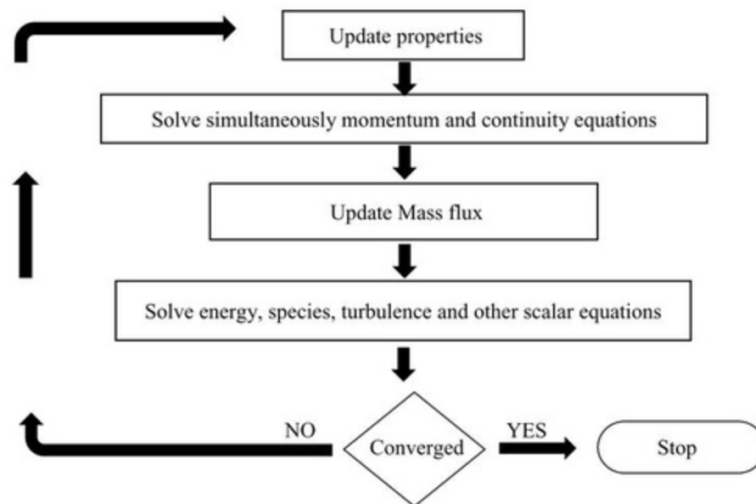


Figure 37. Pressure Based solver with COUPLED method.

When using the COUPLED method, the governing equations are solved simultaneously resulting to a greatly enhanced solution convergence rate but also the need of much more memory requirement is generated. [7]

In ANSYS, the Energy equation is solved in the following form:

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\vec{v}(\rho E + p)) = \nabla \cdot \left(k_{eff} \nabla T - \sum_j h_j \vec{J}_j + (\bar{\tau}_{eff} \cdot \vec{v}) \right) + S_h \quad (34)$$

where k_{eff} is the effective conductivity and \vec{J}_j is the diffusion flux of species j . The first three terms on the right-hand side of Eq. 34 represent the energy transfer due to conduction, species diffusion and viscous dissipation respectively. The S_h term includes the heat of chemical reaction and any other volumetric heat sources that have been defines. In Eq. 34,

$$E = h - \frac{p}{\rho} + \frac{v^2}{2} \quad (35)$$

where the sensible enthalpy h is defined for ideal gases as

$$h = \sum_j Y_j h_j \quad (36)$$

To sum up and considering the above, the chosen method was Pressure – Based solver that will use the SIMPLE method. The movement of the pistons it will result to a transient problem and due to the low frequencies, there will be no turbulence and the flow will be considered laminar. Also due to heat transfer the Energy Equation will be solved too.

4.7.2 Initialization process.

Before we start solving the equations, the solutions must be first initialized. This means that initial values will be set for all the variables. There are two ways to proceed to initialization. We can either compute the initial values from information in a specified zone and enter them manually or have the solver compute average values based on all zones. Also, we can indicate whether the specified values for velocities are absolute or relative to the velocity in each cell zone.

In our case, we set an initial average pressure at approximately 1 bar and an initial average temperature at 1 K. But of course, in every cylinder the operational pressure and temperature differs from the initial values but after some cycles the values start to stabilize and follow a sinusoidal pattern. The following Fig. 38, Fig. 39, Fig. and Fig. 40 show the difference between the initial and operational pressure through the cycle in all the cylinders when operating 160° out of phase.

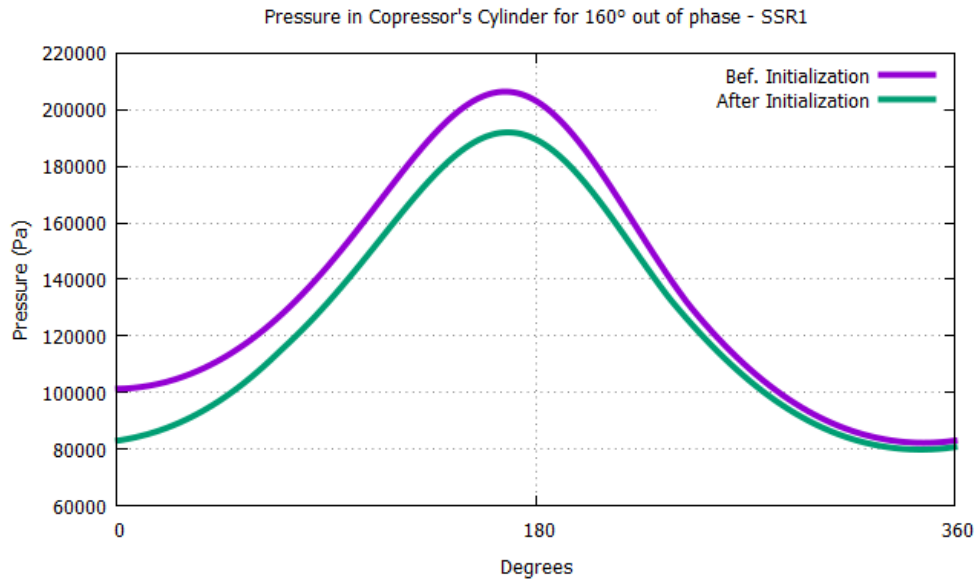


Figure 38. Difference between the initial and operating pressure – C1.

The initial value of 1 bar in the expander – SR2 was finally stabilized with a value at the start of cycle at approximately 0,81 bars after several cycles were executed.

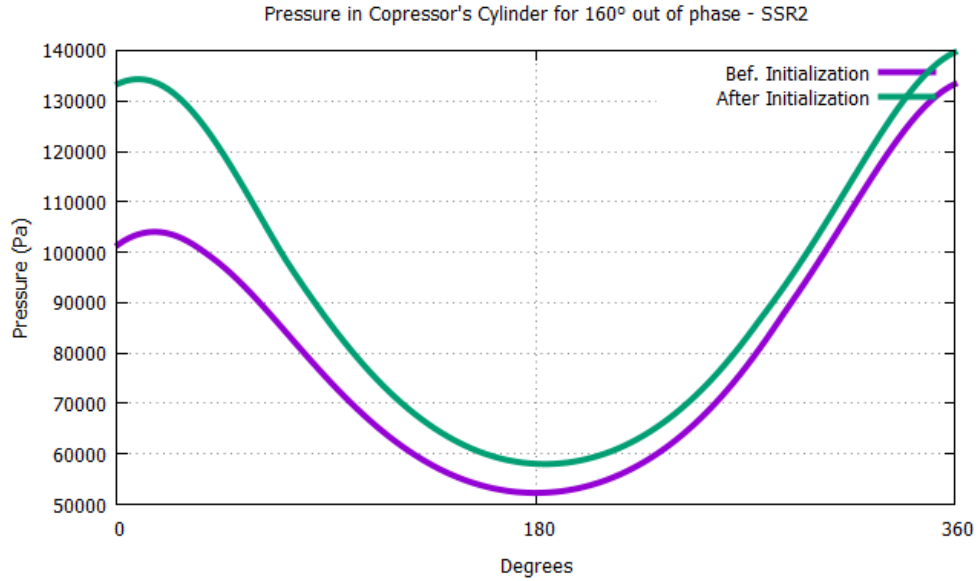


Figure 39. Difference between the initial and operating pressure – C2.

The initial value of 1 bar in the expander – SR2 was finally stabilized with a value at the start of cycle at approximately 1,33 bars after several cycles were executed

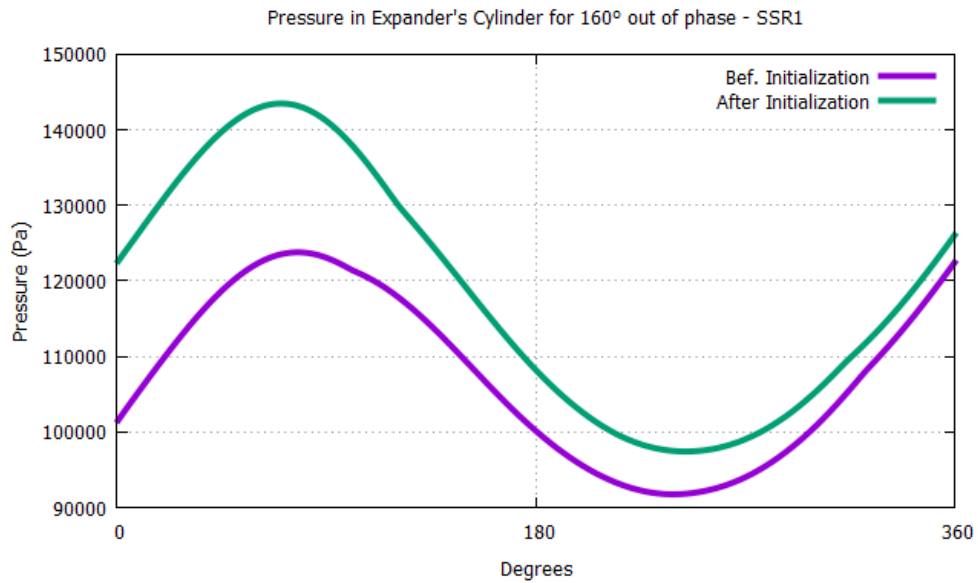


Figure 40. Difference between the initial and operating pressure – E1.

The initial value of 1 bar in the expander – SR2 was finally stabilized with a value at the start of cycle at approximately 1,23 bars after several cycles were executed

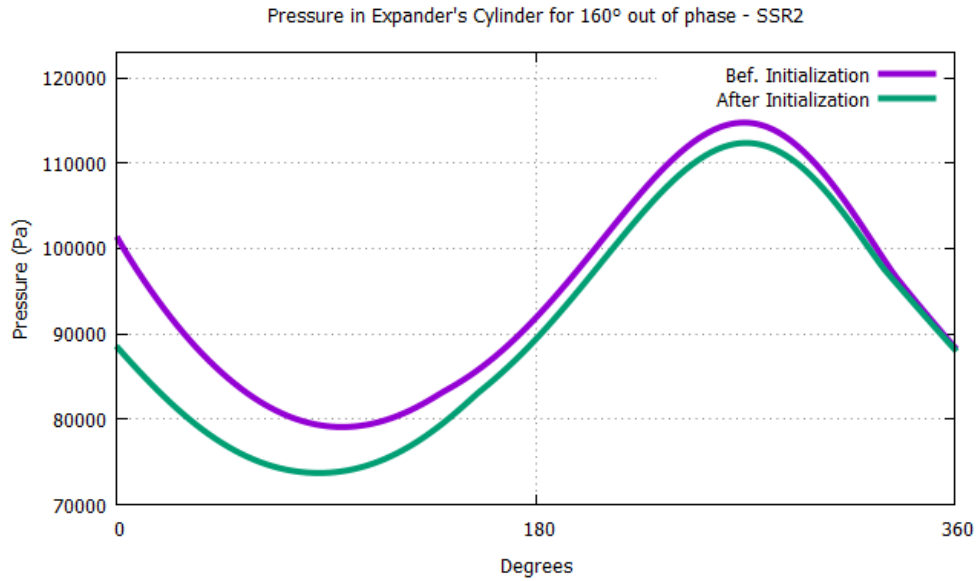


Figure 41. Difference between the initial and operating pressure – E2.

The initial value of 1 bar in the expander – SR2 was finally stabilized with a value at the start of cycle at approximately 0,89 bars after several cycles were executed.

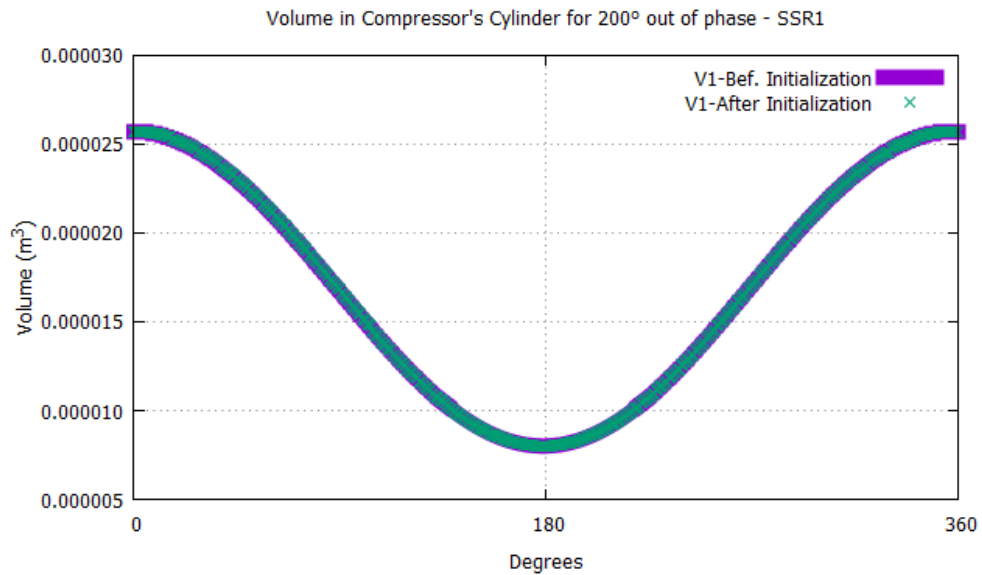


Figure 42. Difference between the initial and operating volume.

As we can see there is no difference in the volumes among the cycles.

4.7.3 Time step.

When we have to choose an appropriate time step size in a transient simulation, especially when performing a sensitivity study, we must consider the tradeoff between accuracy and computational cost. Smaller time steps will typically yield more accuracy stability at the expense of longer run-times and increased computational cost.

In this specific simulation, we considered necessary that in each time step the piston's movement would be less than 1 crank shaft angle and the total time steps needed for one cycle will be equally divided to the 360 degrees. As a result, the chosen time step was 0.0001736111 seconds and the total number of time steps for every cycle was set to 5760. Last but not least the max iteration per time step were set to 50 as shown in Fig 43.

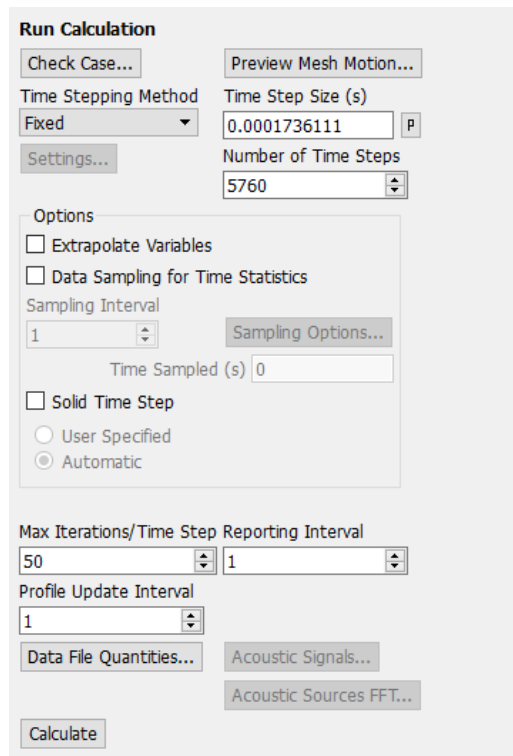


Figure 43. Selecting the appropriate time step.

Every degree was separated to 16 time steps resulting to 5760 total time steps for a complete cycle and a time step size 0.0001736111 seconds.

4.8 Data Management.

The final step when setting up a simulation, is to set which variables we will be monitored and later exported. These variables will be used to calculate the Coefficient of Performance. In this case, we will monitor the press and volume change in every cylinder, the heat fluxes of the heater walls and finally the mass flow in the regenerators to ensure that there won't be a parallel flow towards the heaters, as it will be explained in more details in later chapters.

Initially as illustrated in Fig. 44, the pressure of all the pistons will be exported by selecting Report type Volume – Average and then Absolute Pressure. The same procedure is followed for all the cylinders.

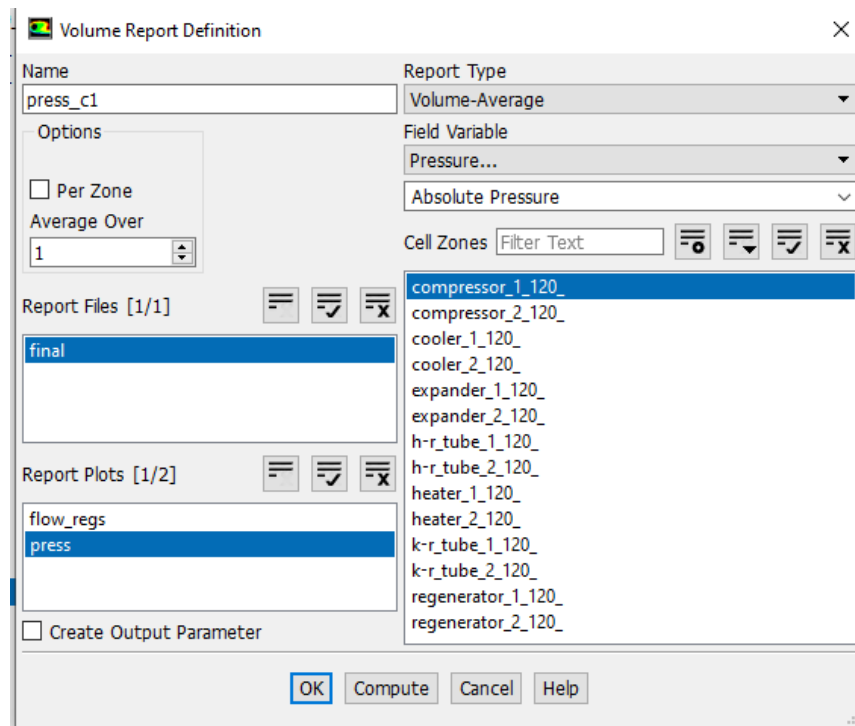


Figure 44. Pressure Data.

Pressure changes will be exported because they are correlated with the work.

Similarly, the volume changes will be exported by selecting Report Type Volume as illustrated in Fig. 45 and repeat the procedure again for all the cylinders.

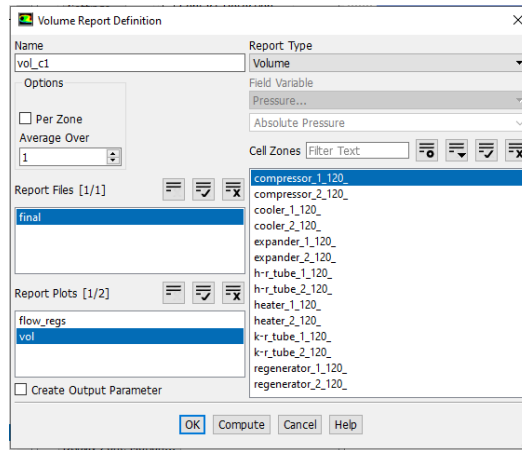


Figure 45. Volume Data.

Volume changes are also correlated with the work and will be used later to calculate the COP.

Then the Heat Fluxes were exported by selecting Report Type Integral then Wall Fluxes and Total Surface Heat Flux as shown in Fig. 46

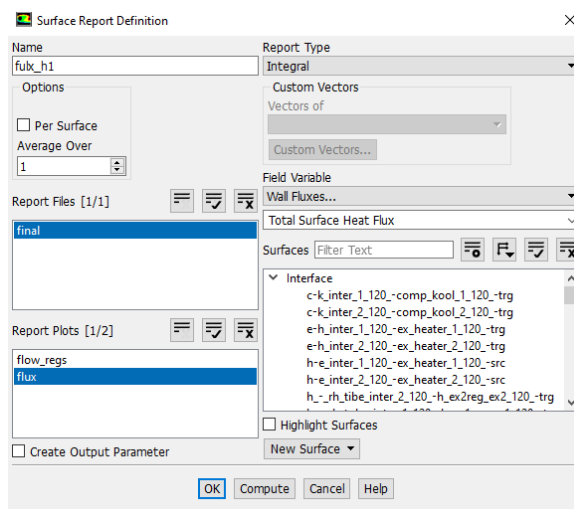


Figure 46. Heat Flux Data.

Heat Flux will be monitored for both Heaters

The final variable that it will be exported is the mass flow rate inside the recuperator for both machine 1 and machine 2. To achieve this, we will monitor in every time step the flow from the Heater to the recuperator and from the recuperator the Cooler at the same time as shown in Fig. 47.

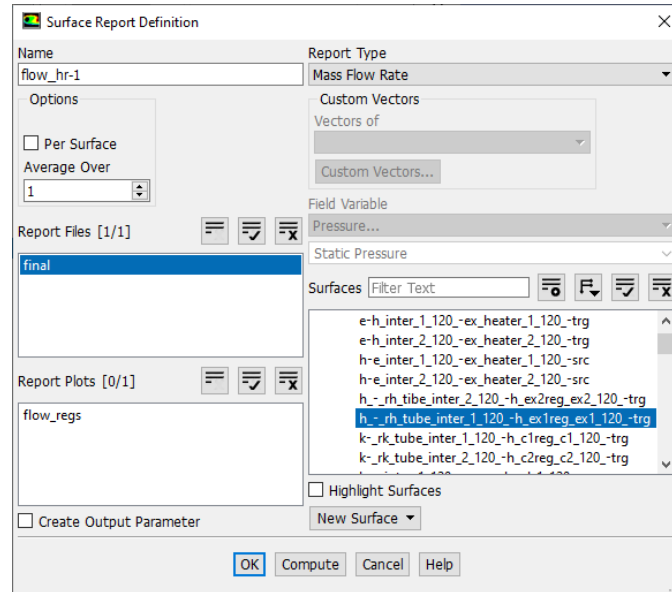


Figure 47. Mass Flow Rate.

The above figure, illustrates the set up to export the mass flow rate data that flow from the heater to the recuperator.

Chapter 5.

Data Analysis

In this chapter, we will analyze the data that we obtained in the previous simulations and also visualize the flows in specific diagrams so we will determine if the SSR can provide cooling power. There were made several simulations for a variate of degrees when operating out of phase. In more details, we examine the conditions when we operate with a phase difference of 152° , 160° , 170° , 180° , 200° and 203° . Each simulation will be presented and explained separately.

In this part, is very important to explain the different types of flows and the connection between them and the Coefficient of Performance. So, there are three kinds of flows in the recuperator of the machine system. During counterflow, gas in the two channels is flowing towards opposite directions. In that case, the recuperator is operating in a similar way to a Stirling cryocooler regenerator, with the fluid in one of the channels operating as the regenerator matrix. According to the Fig. 48, positive values of flow-rate indicate that gas is flowing from the cooler of the machine towards the machine's cooler. This means that counterflow exists in almost the whole duration of the cycle except of two small time intervals around 170° and 350° of engine's cycle. At those intervals, parallel flow towards the system coolers is noticed. In that case, the recuperator does not act as a regenerator. However, this situation is not troubling as both flows can be heated by the environment. The troubling case is that of parallel flow towards the heaters. This case is not present in the simulation, as there is no time interval in which both flow rates have a positive value. This case is disruptive as a significant amount of heat may end up to the heaters of the machine and reduce the cooling potential of the machine to zero. [9], [10]

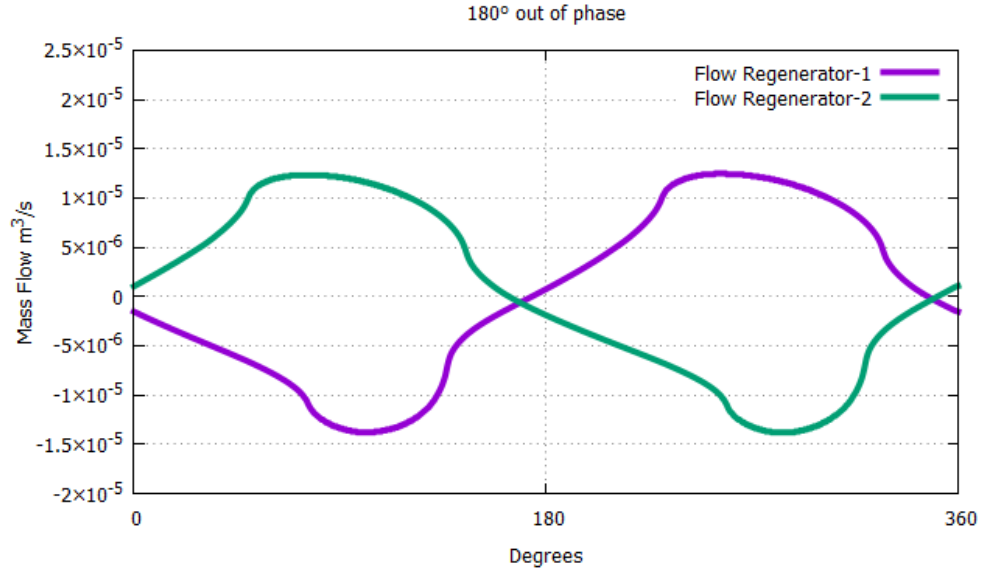


Figure 48. Flows in both regenerators operating 180° out of phase.

Counterflow exists in almost the whole duration of the cycle except of two small time intervals around 170° and 350° of engine's cycle.

Considering the above, for every simulation initially we will examine the Mass Flow illustrations to determine if there is parallel flow towards the heaters. If this is true, COP will be calculated just to verify it. If no parallel flow towards the heaters is detected, then the COP will be calculated to compare the cooling power that the machine will provide when operating in these degrees and we will illustrate the diagrams of pressure and volume changes inside the cylinders.

In both cases the coefficient of performance (COP) is calculated as shown in Eq. 37. COP is defined by the cooling heat and the total work input where

$$COP = \frac{Q_h}{W_t} \quad (37)$$

and

$$W_c = p_c \cdot dV_c \quad (38)$$

$$W_e = (p_c - dp) \cdot dV_e \quad (39)$$

5.1 Operating 120° out of phase.

When the SSR operates at 120° out of phase the Fig.50 illustrates the velocity and the displacement of the pistons on both compressors and expanders. As we explained earlier initially we will examine the flows on the regenerators to detect if there are parallel flows towards the heaters. As shown in Fig. 49, counterflow exists in almost the whole duration of the cycle except of two points of interest at around 80° and 260°. At those intervals, we observe parallel flows. At 80° the flows have negative values so they are heading towards the system coolers. In that case, the recuperator does not act as a regenerator. However, this situation is not troubling as both flows can be heated by the environment. On the other hand, at around 260° we detect that the intersection point is on the positive values resulting to a parallel flow towards the system heaters. This is a disruptive case as a significant amount of heat may end up to the heaters resulting to an SSR with zero cooling power.

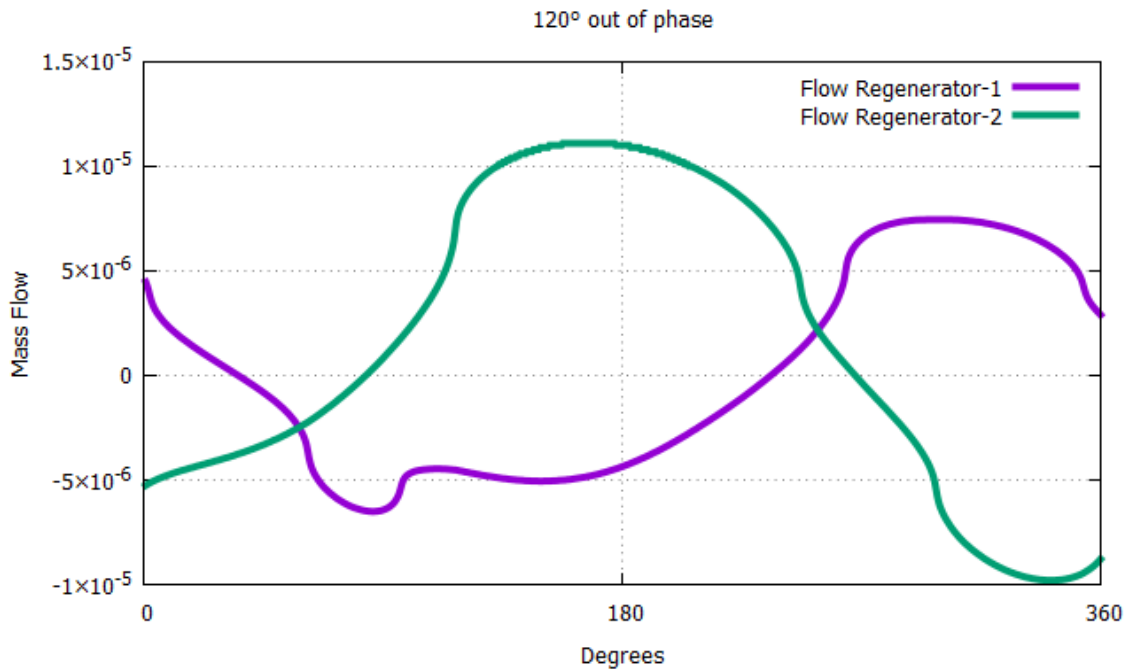


Figure 49. Mass Flows when operating 120° out of phase.

At 80° we observe a parallel flow towards the coolers and at 260° a parallel flow towards the heater, something that can potentially eliminate the SSR's cooling power.

To confirm the above statements, we will calculate the coefficient of performance (COP), if the results lead us to a COP greater than 100% or with negative values obviously the statements are correct. So, from the data that we exported during the simulation at 120° we calculated,

$$W_{total} = -0,0136 \text{ J/c}$$

$$Q_{total} = -0,0286 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 210 \%$$

Obviously, the SSR cannot deliver this COP and our statements are confirmed.

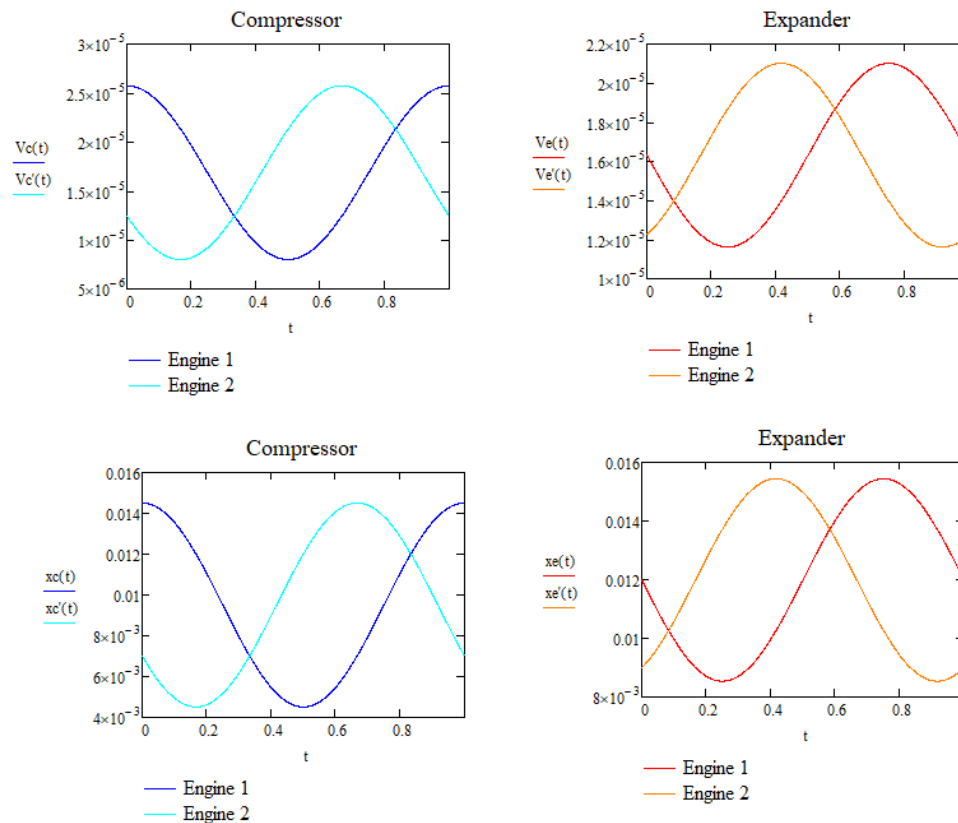


Figure 50. Velocities and Displacements when operating 120° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle.

5.2 Operating 140° out of phase.

Similarly, with the 120°, when operating 140° out of phase the Fig. 52 illustrates the the velocity and the displacement of the pistons on both compressors and expanders. Still in this simulation, counterflow exist in almost the whole cycle but now the points of interest are at approximately 60° and 240° as shown in Fig. 51. For the same reason that we explained previously, the troubling situation is again is the intersection point on the positive values at around 240° resulting to an that SSR cannot provide cooling power. For confirmation, we will calculate the coefficient of performance (COP) at this case too. From the expoting data we calculated,

$$W_{total} = -0,013 \text{ J/c}$$

$$Q_{total} = -0,028 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 210 \%$$

Obviously, the SSR cannot deliver this COP and our statements are again confirmed.

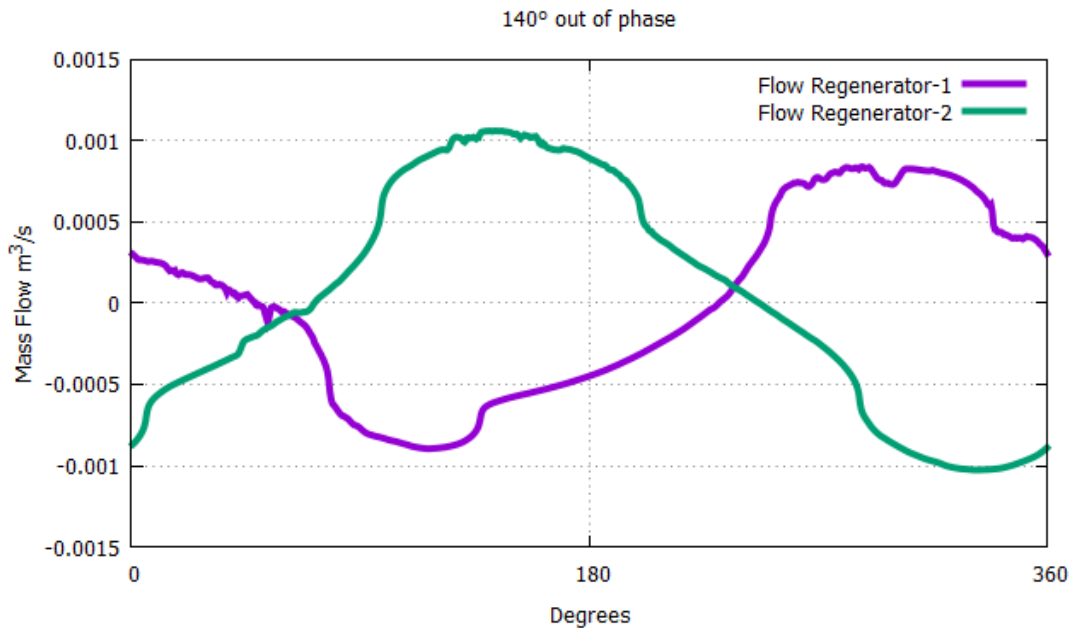


Figure 51. Mass Flows when operating 140° out of phase.

At 80° we observe a parallel flow towards the coolers and at 260° towards the heaters.

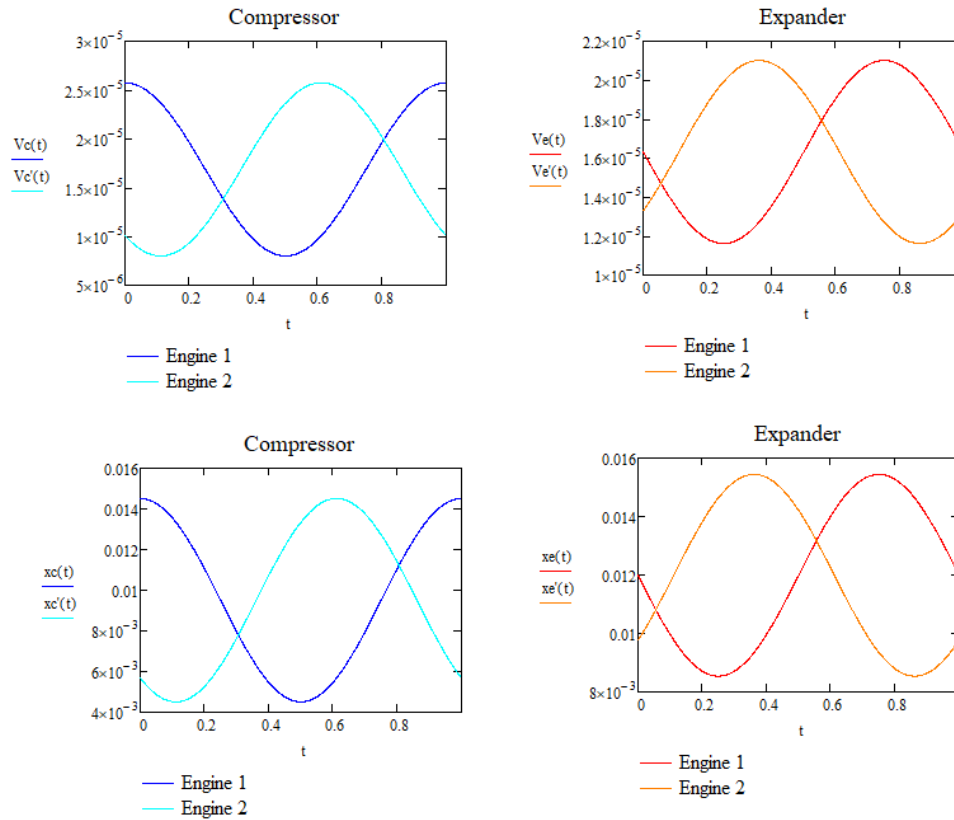


Figure 52. Velocities and Displacements when operating 140° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle.

5.3 Operating 152° out of phase.

As we noticed from the previous simulations, as we increase the phase difference, when starting from 120° , the intersection point is moving closer to the x axis and to the negative values. This means that we are approaching the lower bound where the machine is starting to provide cooling power. This also why in this simulation we increased the degrees by 12 and not by 20 like the previous. Again, the flows and velocities with displacements are represented in Fig. 53 and Fig. 54 respectively.

As we observe at Fig. 53, at approximately 200° the interval point is still above the x axis. Significantly closer but there is still a parallel flow towards the heaters. We calculated again the Coefficient of performance (COP) but as shown below it was about

32.59%. This means that with the specific data the machine could have delivered a cooling power if we could somehow overcome the parallel flow problem.

From the exporting data when operating 152° out of phase we calculated,

$$W_{total} = -1,812 \text{ J/c}$$

$$Q_{total} = -0,590 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 32,59 \%$$

This time, the data indicate that cooling power is achievable but we cannot overcome the parallel flow problem.

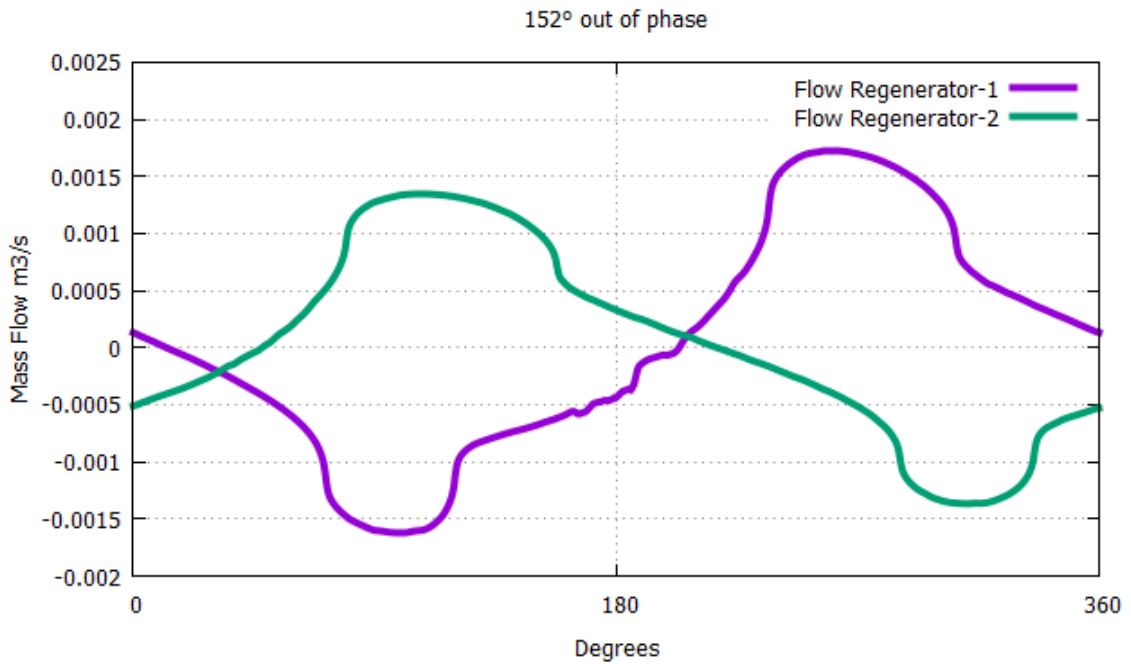


Figure 53. Mass Flows when operating 152° out of phase.

At 20° we observe a parallel flow towards the coolers and at 200° towards the heaters.

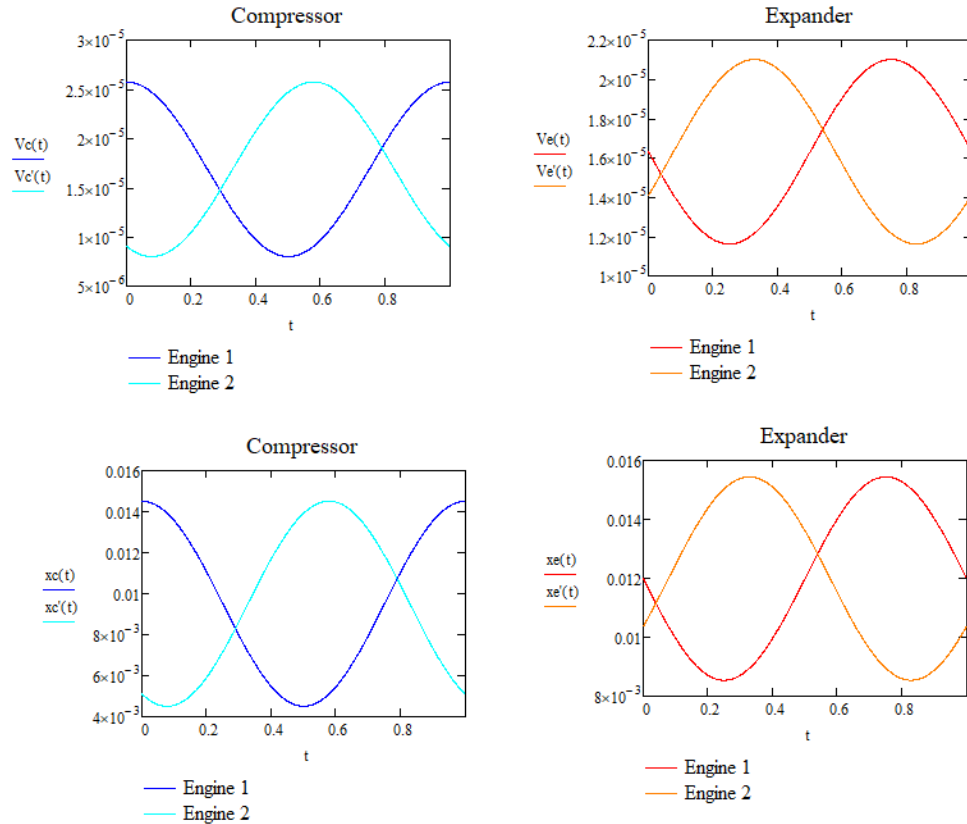


Figure 54. Velocities and Displacements when operating 152° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle.

5.4 Operating 160° out of phase.

From the previous simulation, we conclude that by increasing again the degrees, the more the intersection point will be on the x-axis resulting to no parallel flows towards the heaters. That's the reason why in this simulation we increased the degrees just by 8 and we will examine the case when we operate 160° out of phase. Similarly, with previous simulations, Fig. 55 and Fig 56 illustrate the flows and the velocities with displacements respectively. In more details, as shown in Fig. 55, at approximately 190° the two flows are intersecting exactly at the x-axis. This means that there is no parallel flow towards the heaters and the parallel flow can be observed at around 20° but we can easily overcome it by using heat from the environment.

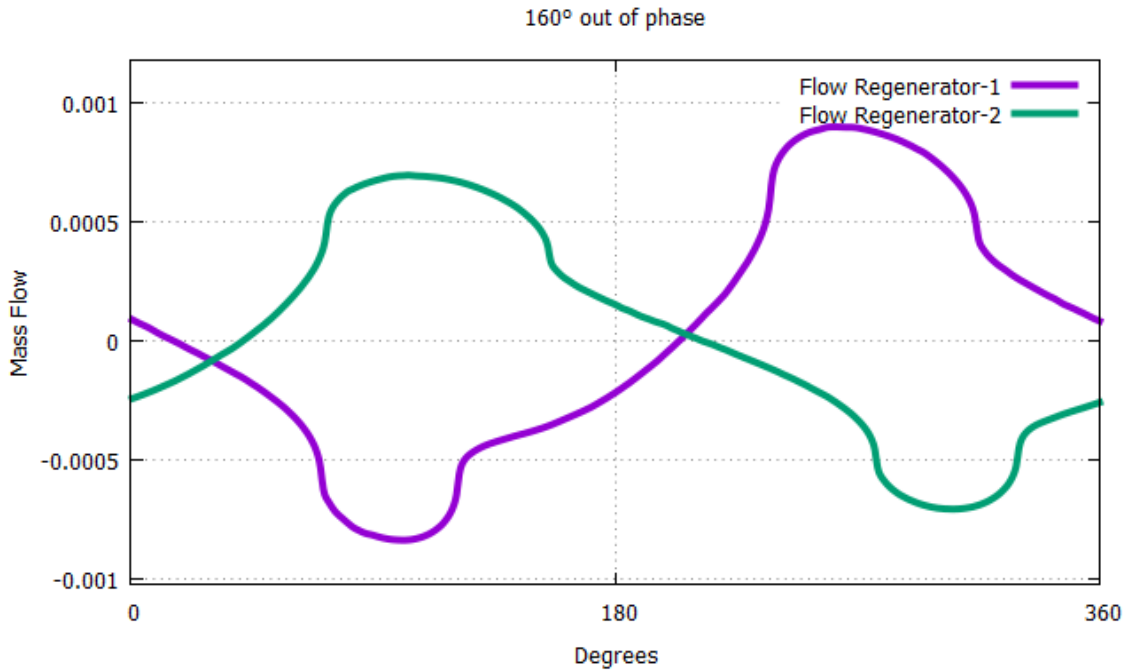


Figure 55. Mass Flows when operating 160° out of phase.

At 20° we observe a parallel flow towards the coolers and at 190° the two flows are intersecting at the x-axis. As a result, there is no parallel flow towards the system heaters.

From the exporting data when operating 160° out of phase we calculated,

$$W_{total} = -0,930 \text{ J/c}$$

$$Q_{total} = -0,538 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 57,87 \%$$

From the above, we conclude that at 160° there is no parallel flow towards the system heater and the SSR is starting to provide cooling power with a Coefficient of performance (COP) equal to 57,87%. At Fig. 57 and Fig. 58 are illustrated the press and volume changes of all the cylinders during a cycle so the operating range is more understandable.

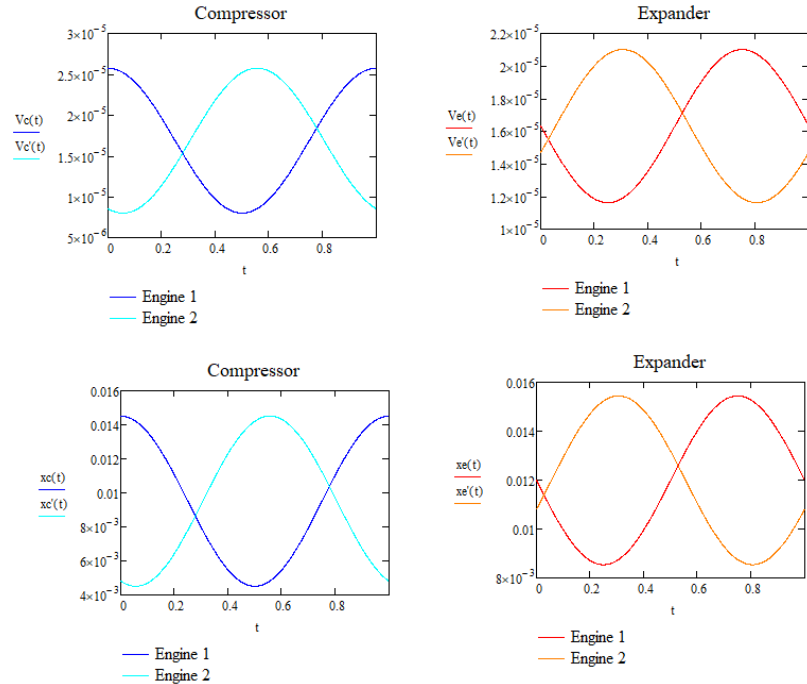


Figure 56. Velocities and Displacements when operating 160° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle

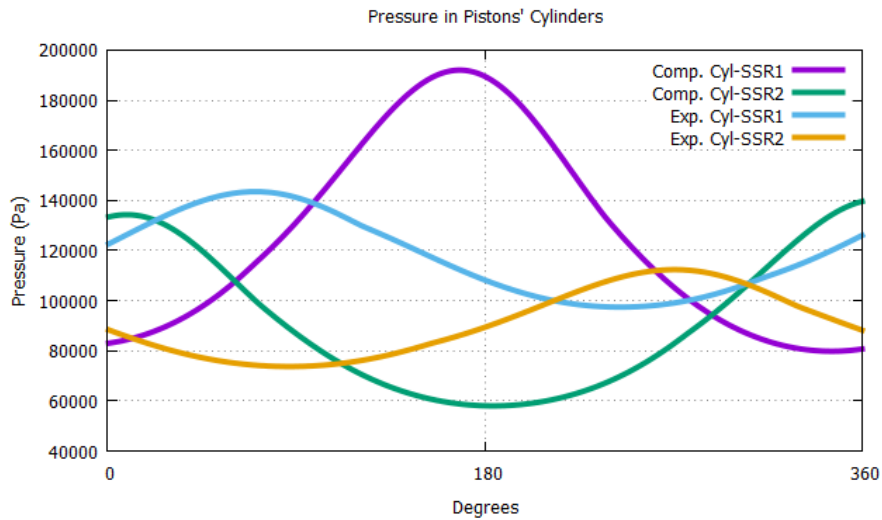


Figure 57. Pressure changes when operating 160° out of phase.

It is normal that the pressures don't have the same oscillation width because is proportional to the initial positions.

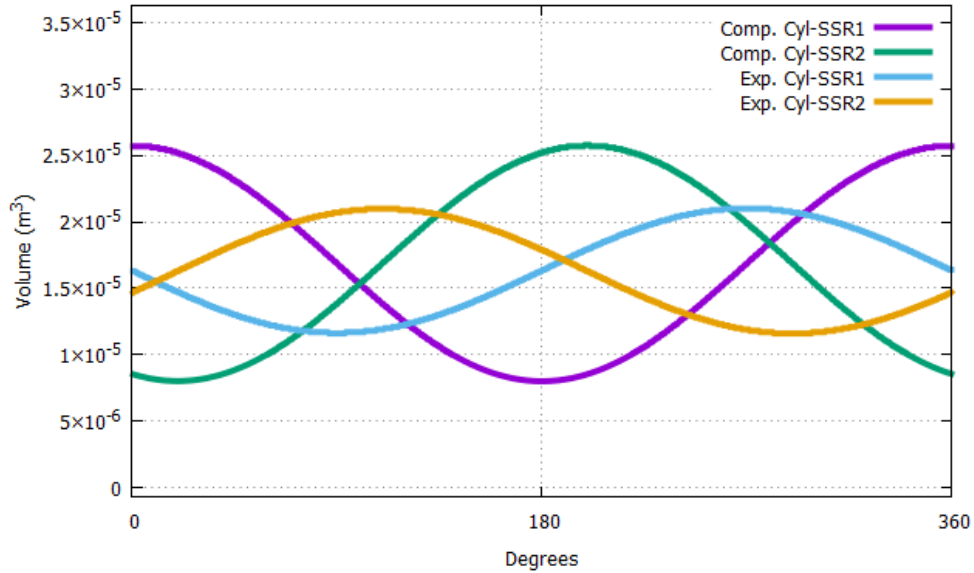


Figure 58. Volume changes when operating 160° out of phase.

As we can observe the volumes of the compressors of the two machines have the same oscillation width. This means that the volumes are completely equal. Similarly, the volumes of the expanders of the two machines are equal too. The difference between the compressor's volume and expander's volume was made on purpose during the design and it is due the fundamental principles that governing the SSR in order to operate properly.

5.6 Operating 170° out of phase.

Now that we concluded that the machine starts providing cooling power when it operates with 160° out of phase, we will try to determine the operating range by gradually increasing the degrees difference by 10° until we will detect the parallel flow problem in the Mass Flow illustrations. The Fig. 59 and Fig 60 illustrate the flows and the velocities with displacements respectively. Similarly, with the previous simulation, as it is shown in Fig. 59 there no parallel flow towards the heaters and the counter flow exists almost during the whole cycle. Only exception at 40°, where there is a parallel flow towards the system coolers but we can easily confront it by using heat from the environment. Now that we can assure that the machine works properly we will calculate the Coefficient of Performance by using the exported data. Also, the pressure and volume changes when operating at 170° are presented in Fig. 61 and Fig. 62 respectively.

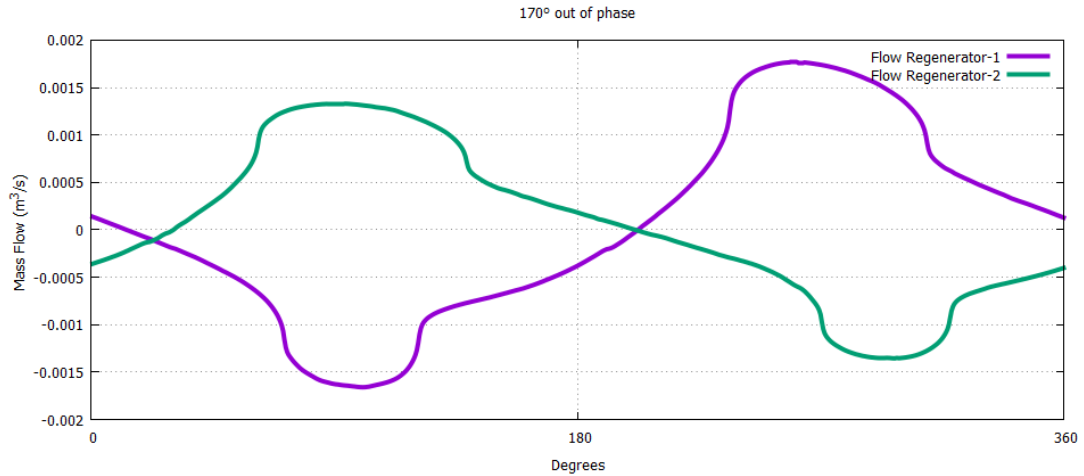


Figure 59. Mass Flows when operating 170° out of phase.

At 40° we observe a parallel flow towards the coolers and at 200° the two flows are intersecting at the x-axis. As a result, there is no parallel flow towards the system heaters and the machine provides cooling power.

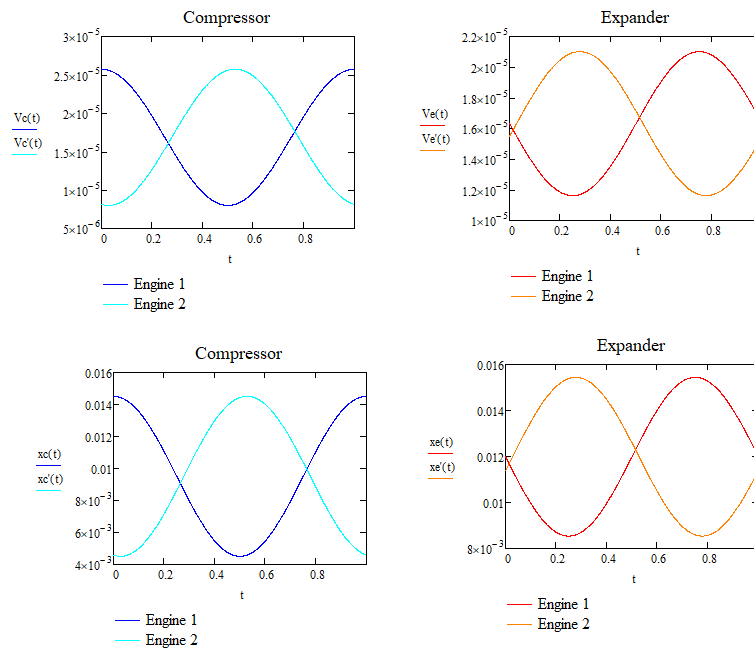


Figure 60. Velocities and Displacements when operating 170° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle

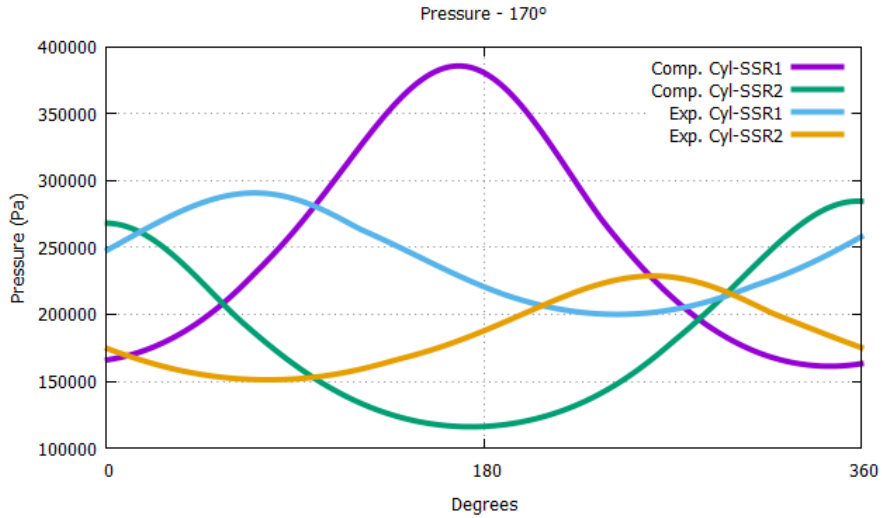


Figure 61. Pressure changes when operating 170° out of phase.

It is normal that the pressures don't have the same oscillation width because is proportional to the initial positions.

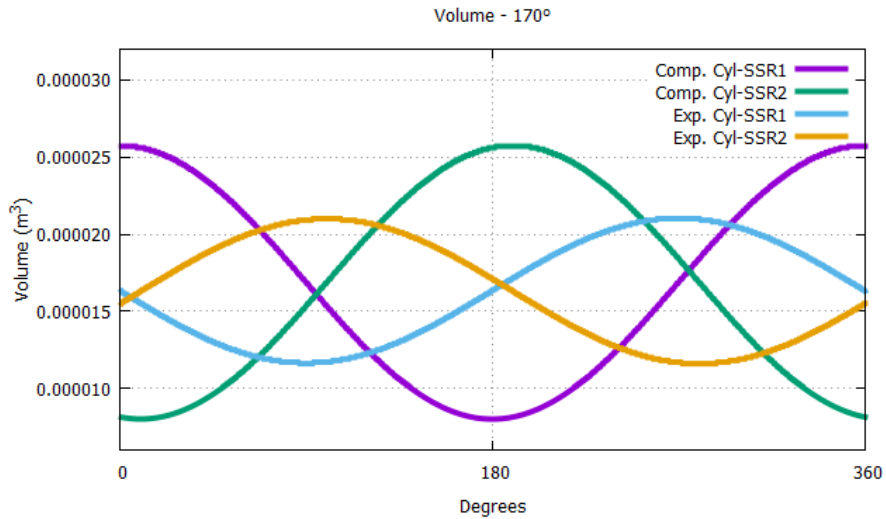


Figure 62. Volume changes when operating 170° out of phase.

As we can observe the volumes of the compressors of the two machines have the same oscillation width. This means that the volumes are completely equal. Similarly, the volumes of the expanders of the two machines are equal too. The difference between the compressor's volume and expander's volume was made on purpose during the design and it is due the fundamental principles that governing the SSR in order to operate properly.

From the exporting data when operating 170° out of phase we calculated,

$$W_{total} = -1,849 \text{ J/c}$$

$$Q_{total} = -0,591 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 31,98 \%$$

The Coefficient of Performance indicates that despite the small incensement in the operating degrees difference there is a huge COP reduction to almost the half as when operating at 160°. That means that the system is very sensitive in degrees changes and the machine's optimum operation will be achieved for a specific degree and not for range of degrees. After the COP for each simulation will be calculated, through a polynomial approach we will determine the optimum design point so it is important to examine the SSR for a variety of degrees difference.

5.7 Operating 180° out of phase.

In this section, we examine the Coefficient of Performance (COP) when we operate 180° out of phase. As shown in Fig. 59, there is no parallel flow towards the system heaters and both intersection points are below the x-axis.

From the exporting data when operating 180° out of phase we calculated,

$$W_{total} = -0,0119 \text{ J/c}$$

$$Q_{total} = -0,0026 \text{ watt}$$

Resulting to

$$COP = \frac{W_{total}}{Q_{total}} = 22,61 \%$$

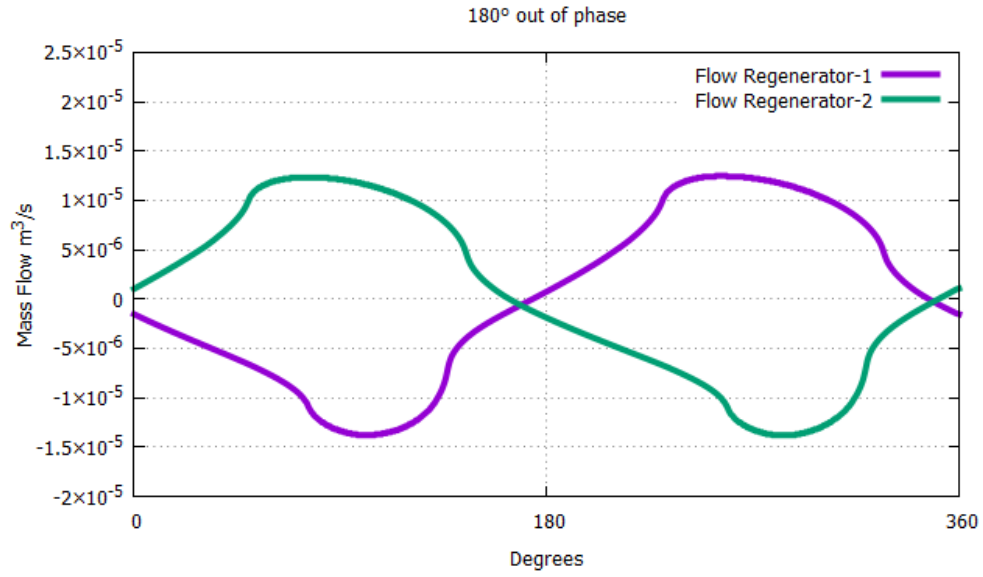


Figure 63. Mass Flows when operating 180° out of phase.

At 170° and at 350° we observe a parallel flow towards the coolers. As a result, there is no parallel flow towards the system heaters

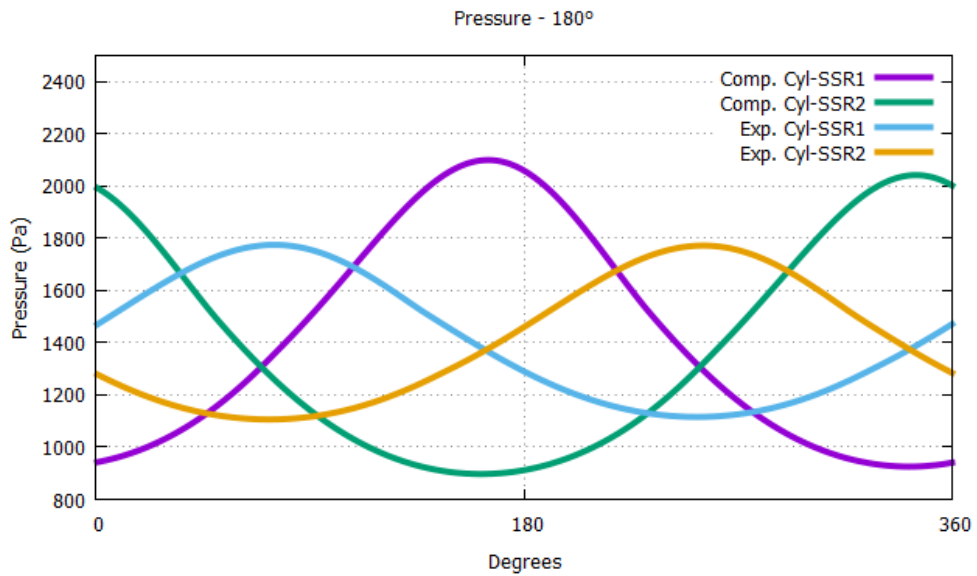


Figure 64. Pressure changes when operating 180° out of phase.

It is normal that the pressures don't have the same oscillation width because is proportional to the initial positions.

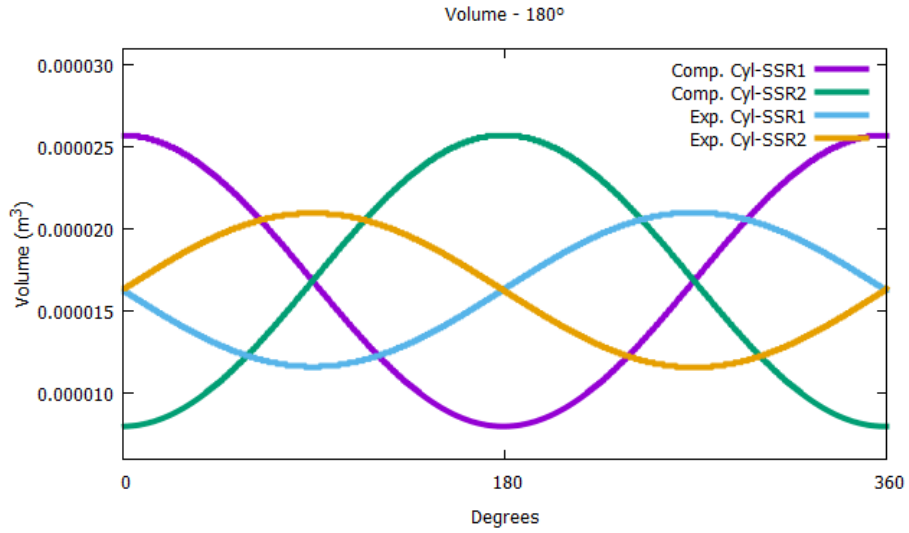


Figure 65. Volume changes when operating 180° out of phase.

Opposite velocities result to exactly opposite volume changes when operating at 180° out of phase.

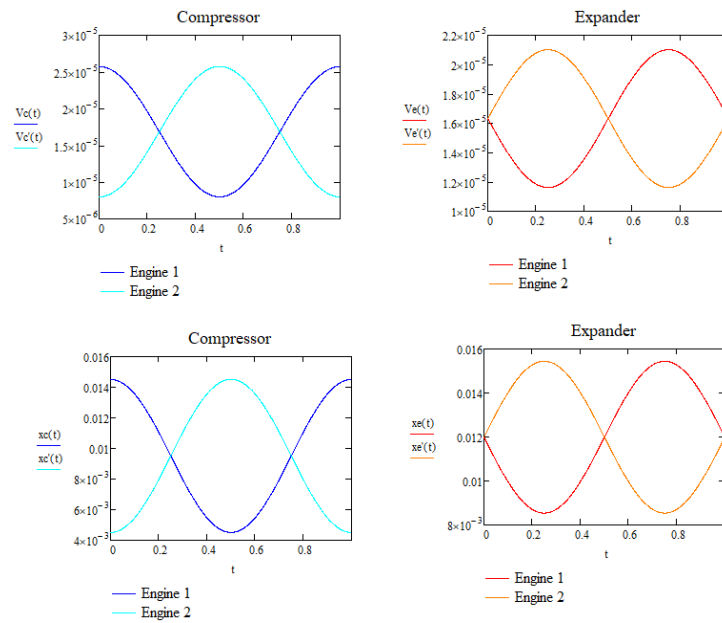


Figure 66. Velocities and Displacements when operating 180° out of phase.

Above are illustrated all the velocity and displacement changes for all piston during the duration of one cycle

As we can result from the illustrations, when operating at 180° out of phase, the SSR provides cooling power but again with a significantly lower COP than the initial simulation. In more details, we determined that starting from almost 60% COP when operating at the lower bound at approximately 160° , the increasement of the degree difference will result in COP decrease. At 180° the COP has already dropped by almost 38%. This means that gradually the machine will reach the upper bound (in terms of degrees difference) and start to present again parallel flow towards the heater, and the SSR won't be able to provide cooling power. In order to determine the upper bound, we will increase the degrees difference and restart the simulation. The results are presented in the next sections.

5.8 Operating 200° out of phase.

In this section, the phase difference is increased by 10° since the previous simulation. As shown in Fig. 67, parallel flow towards the cooler exist in this case too. The difference between the cases is that it is present every time in different time during the cycle. In this case, as illustrated in Fig. 67, parallel flow is presented at approximately 180° . Also, at the end of the cycle, we can see that intersection point is just a bit above x-axis. In this point, we have to say that the illustrations' accuracy is proportional to the time step size and to every how many steps the data are exported. Considering this and that the interval between the intersection point and x-axis is small, we can assume that the parallel flow towards the heater it is not presented yet but it will start in the next few degrees. So probably, the 200° is the upper bound but we will confirm during the next simulation where we will increase again the degrees but just a few just to observe if the interval between the x-axis and the intersection point is increased, resulting to parallel flow towards the system heaters. Also, the pressure and volume changes when operating at 170° are presented in Fig. 68 and Fig. 69 respectively and we will calculate the Coefficient of Performance (COP) using the exported data.

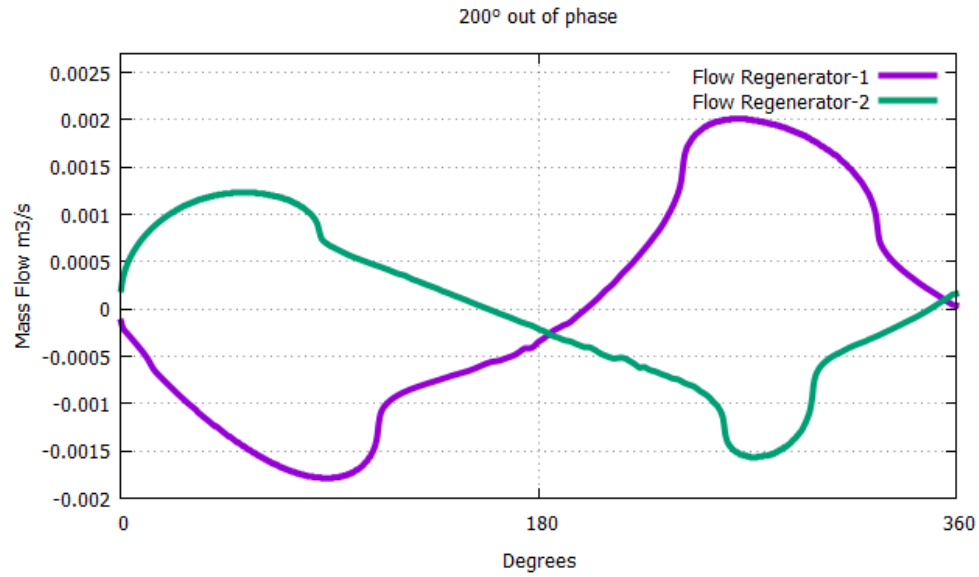


Figure 67. Mass Flows when operating 200° out of phase.

At 180° we observe a parallel flow towards the system coolers and at 355° the two flows are intersecting at the x-axis. As a result, there is no parallel flow towards the system heaters but it will start at the next degrees increasement.

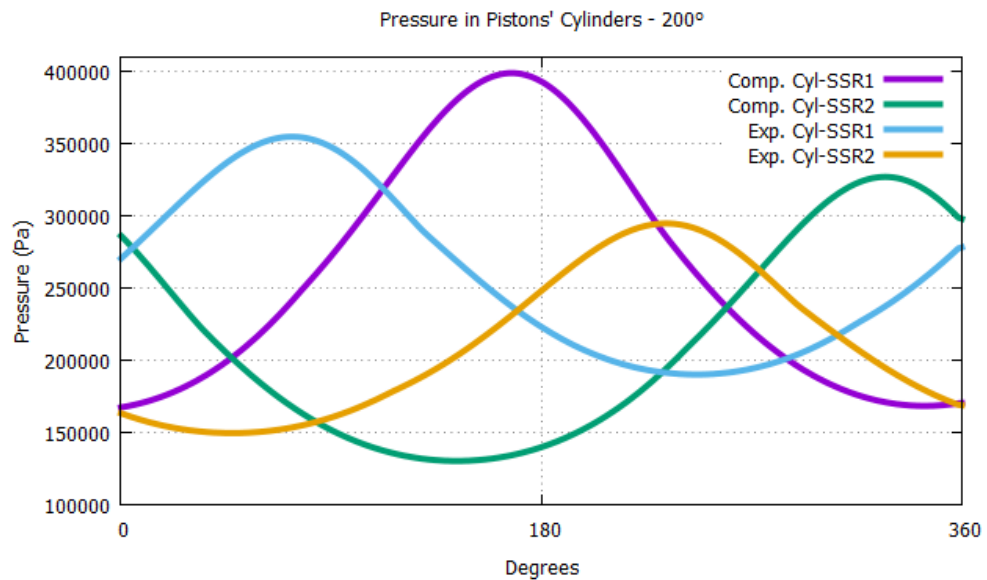


Figure 68. Pressure changes when operating 200° out of phase.

It is normal that the pressures don't have the same oscillation width because is proportional to the initial positions.

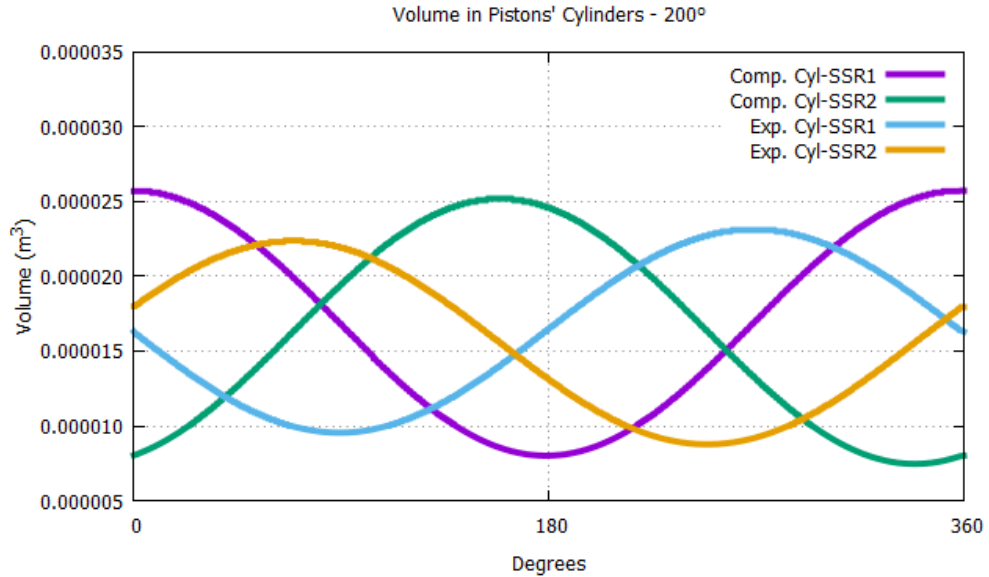


Figure 69. Volume changes when operating 200° out of phase.

As we can observe the volumes of the compressors of the two machines have the same oscillation width. This means that the volumes are completely equal. Similarly, the volumes of the expanders of the two machines are equal too. The difference between the compressor's volume and expander's volume was made on purpose during the design and it is due the fundamental principles that governing the SSR in order to operate properly.

From the exporting data when operating 200° out of phase we calculated,

$$W_{\text{total}} = -2,468 \text{ J/c}$$

$$Q_{\text{total}} = -0,651 \text{ watt}$$

Resulting to

$$COP = \frac{W_{\text{total}}}{Q_{\text{total}}} = 26,38 \%$$

As we can see, the COP is higher than the previous simulation. This will result to a parabolic between the COPs and the degrees increasement, as we expected from the result from the 1D analysis and we will explain in the next and final chapter.

5.9 Operating 204° out of phase.

In this simulation, we increased the degrees but just a few just to observe if the interval between the x-axis and the intersection point is increased, resulting to parallel flow towards the system heaters. As shown in Fig. 70, at approximately 100° it is clear that the interval is greater than the previous simulation and in parallel flow towards the system heaters exists resulting to zero cooling power.

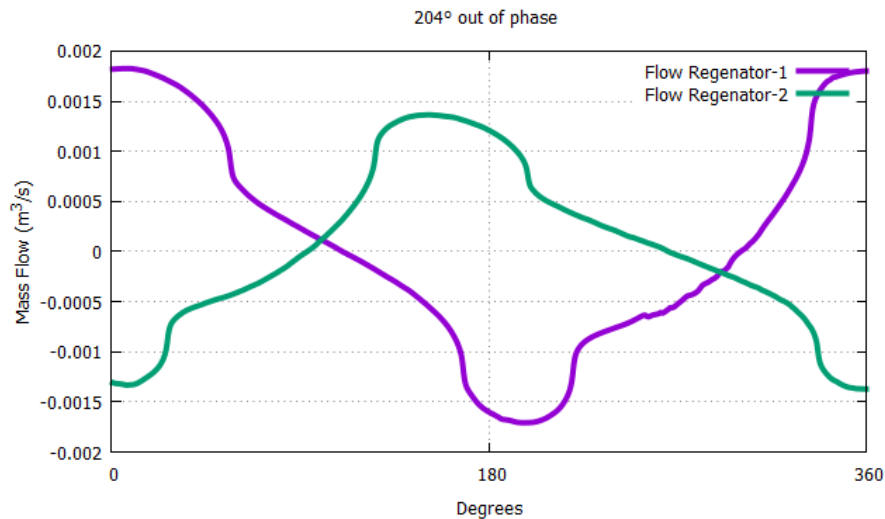


Figure 70. Mass Flows when operating 204° out of phase.

At 100° we observe a parallel flow towards the system heaters, something that can potentially eliminate the SSR's cooling power.

As a conclusion, after we completed and presented all the simulations we ended up that when the recuperator design is used in the SSR, the efficiency and all the workings of the cycle are greatly impacted by the phase difference that is used between the two coolers. The phase difference is between 160° and 200° is where the destructive parallel flow is not detected with optimum Coefficient of Performance at 160°. Besides this range, parallel flow towards the heater eliminated the cooling abilities of the machine.

Chapter 6.

Polynomial Regression – Results Comparison

In this chapter, we will analyse and illustrate the correlation between the Coefficient of Performance and the degrees' difference by using the data that we calculated in the previous chapter. With polynomial regression, we will model the relationship between the changes in degrees and the corresponding conditional mean of COP. After this, the results will be compared to the 1D analysis and to the Schmidt analytical model. [9], [10]

6.1 Polynomial Regression – COP

In this section, we will use numerical analysis and the data from the previous simulations to correlate the COP with every degree in the operating range that we determined in the previous chapter.

Our results from the previous data processing are,

Degrees Difference	Coefficient of Performance
160°	57,87 %
170°	31,98 %
180°	22,61 %
200°	26,38 %

- The above data are presented in the Fig. 71.
- Eq. 31 presents the matrix that was used for the polynomial regression.

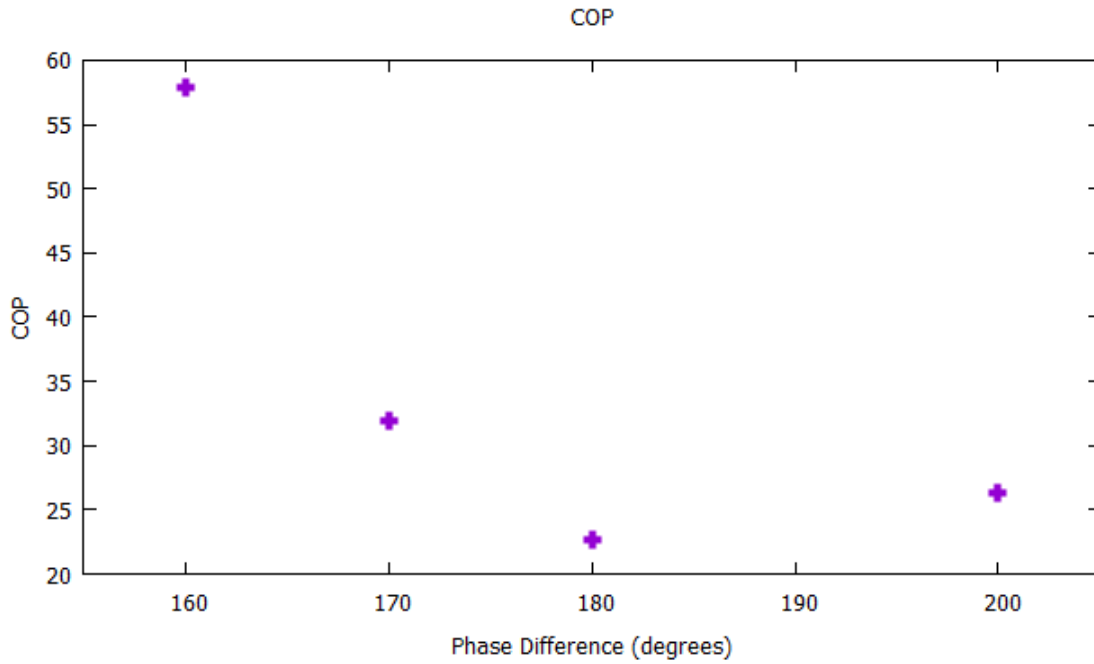


Figure 71. Correlation between COP and Phase Difference.

The correlation between COP and the phase difference will be a parabolic curve.

Using the following data and the matrix for the Polynomial Regression third order we determined the equation that best approaches the COP data with an estimated error equal to 10^{-7} .

i	0	1	2	3
x	160	170	180	200
y	57,87	31,98	22,61	26,38

$$\begin{bmatrix}
 N + 1 & \sum_{i=0}^N x_i & \dots & \sum_{i=0}^N x_i^n \\
 \sum_{i=0}^N x_i & \sum_{i=0}^N x_i^2 & \dots & \sum_{i=0}^N x_i^{n+1} \\
 \vdots & \vdots & \ddots & \vdots \\
 \sum_{i=0}^N x_i^n & \sum_{i=0}^N x_i^{n+1} & \dots & \sum_{i=0}^N x_i^{2n}
 \end{bmatrix}
 \begin{bmatrix}
 a_0 \\
 a_1 \\
 \vdots \\
 a_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 \sum_{i=0}^N y_i \\
 \sum_{i=0}^N y_i x_i \\
 \vdots \\
 \sum_{i=0}^N y_i x_i^n
 \end{bmatrix}
 \quad (40)$$

where

$$\begin{aligned}
 N &= \text{Number of "i"} \\
 n &= \text{Polynomial order} \\
 a_i &= \text{constant variables}
 \end{aligned}$$

So, the Eq. 40 is equal to

$$\begin{bmatrix}
 4 & \sum_{i=0}^4 x_i & \dots & \sum_{i=0}^4 x_i^3 \\
 \sum_{i=0}^4 x_i & \sum_{i=0}^N x_i^2 & \dots & \sum_{i=0}^N x_i^4 \\
 \vdots & \vdots & \ddots & \vdots \\
 \sum_{i=0}^4 x_i^3 & \sum_{i=0}^N x_i^4 & \dots & \sum_{i=0}^N x_i^6
 \end{bmatrix}
 \begin{bmatrix}
 a_0 \\
 a_1 \\
 \vdots \\
 a_3
 \end{bmatrix}
 =
 \begin{bmatrix}
 \sum_{i=0}^4 y_i \\
 \sum_{i=0}^4 y_i x_i \\
 \vdots \\
 \sum_{i=0}^4 y_i x_i^3
 \end{bmatrix}
 \quad (41)$$

The above matrix (Eq. 41) will end up in an equation that correlates the phase difference and the COP with the following form

$$a_0 + a_1 x + a_2 x^2 + a_3 x^3 = COP \quad (43)$$

where the x is the phase difference that the SSR operates.

The results of the Eq. 32 are presented in the following table

Variable	a_0	a_1	a_2	a_3
Value	8237	-127,45	0,6574	-0,0011

By replacing the above values to Eq. 44 we end up to our final equation

$$8237 - 127,25x + 0,6574x^2 - 0,0011x^3 = COP \quad (44)$$

Where x is the phase difference. The Fig. 72 illustrates the correlation between this equation and the data that we have calculated.

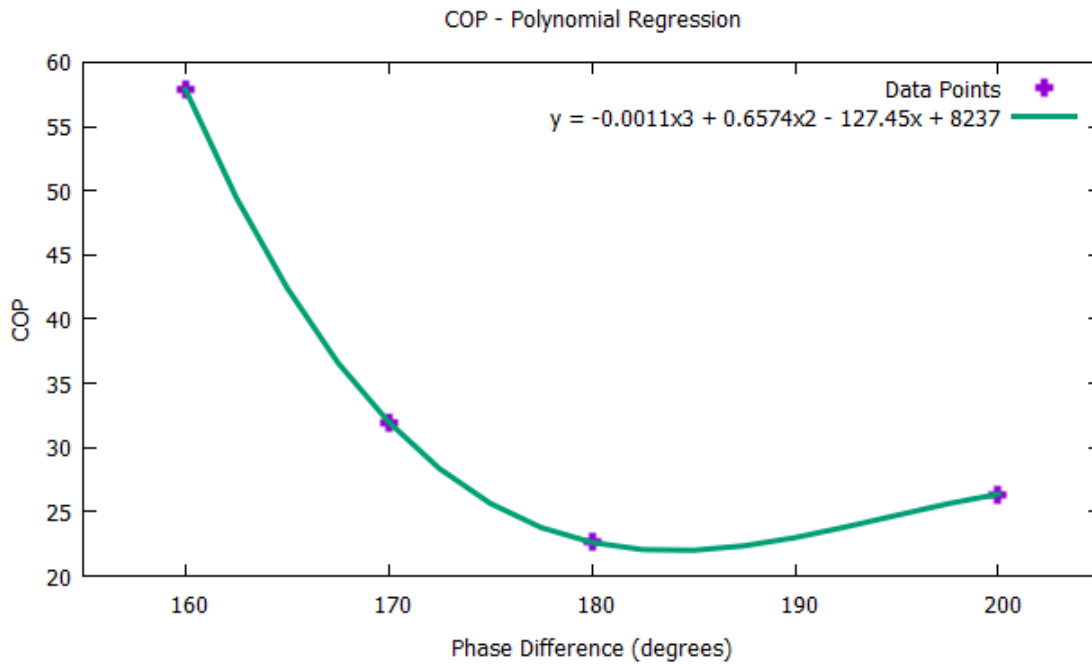


Figure 72. Correlation between phase difference and COP

By using polynomial regression third order we approached the data from our calculations with an error approximately 3×10^{-7} .

From the above figure, we can assume that for 160° out of phase the SSR operates optimally. The COP is decreased proportional with the degrees increase. By deriving the Eq. (44) we can find the local minimum point of the function at approximately 183° where the SSR operates with the lowest COP. All the results are presented in the following table.

	Degrees Diff.	COP
Optimum Design	160°	57,87 %
Worst Design	184°	21,97 %
Operating Range	$160^\circ \sim 200^\circ$	-

6.2 Comparison with the 1D model.

As stated and calculated [10], Fig. 73 illustrates the correlation between the COP and the Phase Difference as resulted from the 1D analysis. The destructive parallel flow is not detected between $X=152\dots209^\circ$. Within these, it is certain that the cooler will work with positive cooling power and the optimum phase difference was to be found at 152° with $COP = 53,024\%$.

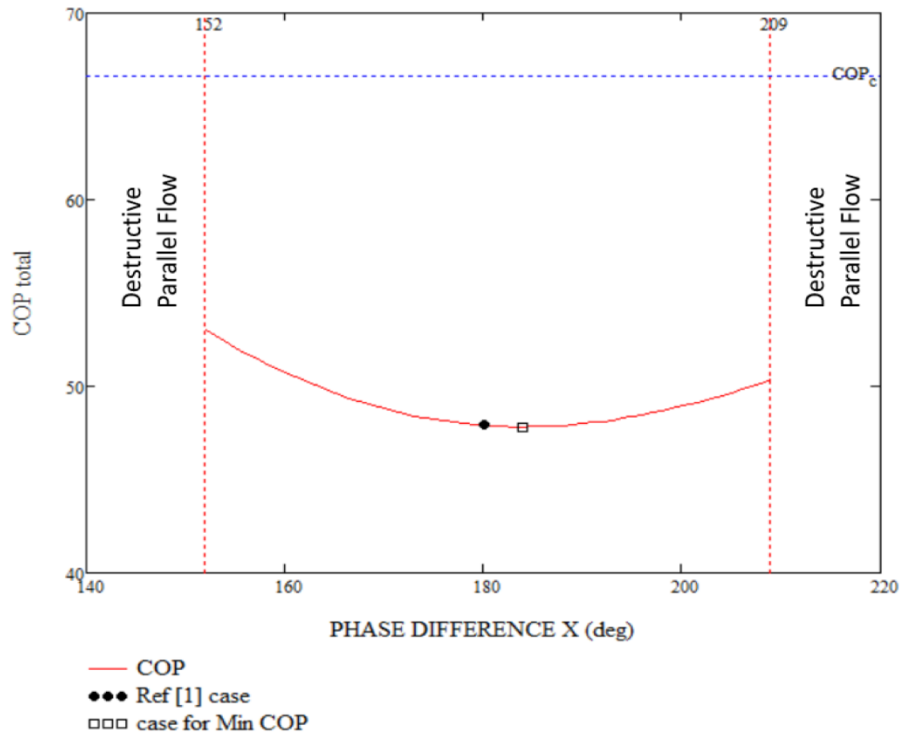


Figure 73. Correlation between COP and Phase Difference – 1D

No parallel flow towards the system heater is detected between 152° and 209° with the optimum phase difference at 152° as resulted from the 1D analysis [9], [10].

Comparing the results with the 3D solution, both models have the best behaviour efficiency wise, just after the destructive parallel flow has ended, at 152° in the 1D model and at the 160° in the 3D model. In both illustrations, Fig 72 and Fig. 73, the polynomial regression follows a parabolic path with the minimum point just after the 180° . The differences are initially the operating range, where in 3D solutions is 17° less. This is maybe

due that as we saw at chapter 4 during the initialization process, after several cycles the values are stabilizing near a final value. The 1D solutions is much less time consuming and more cycles were made ending to a more stable system than the 3D. Another difference between the two systems that has to be taken into account is the interval difference between the minimum and maximum COP value. In 1D the interval is equal to less than 10% when in the 3D is approximately 35%. A justification for that it would be that in 3D model that significant pressure drop was noticed due to the geometry of the machine in the sections when the fluid passes from the recuperator to the heater/cooler. Because of the very low overall pressure of the system this mean value of the pressure drop exceeds 30% of the mean pressure leading to non-uniformity of the pressure in each cryocooler. As a result, same system operates with different work and consistently with different COP in 1D and 3D model.

6.3 Comparison with the Schmidt model.

As priory stated in chapter 3, the coefficient of performance of the refrigerator, COP, is defined as the heat transferred to the expander divided by the total work absorbed by the refrigerator. So according to the COP definition we have for $T_e = 0.4\text{K}$, $T_c = 1\text{K}$

$$COP = -\frac{Q_e}{W_c + W_e} = \frac{T_e}{T_c + T_e} = 66.67\% \quad (45)$$

As it to be expected, the COP calculated with the Schmidt model is always in all circumstances greater than the COP calculated by using CFD results.

Conclusion

In this thesis, through the CFD analysis that was made, we were able to determine the optimal phase difference that provides the better efficiency when is applied to a Superfluid Stirling Refrigerator. Initially, by setting multiple cases and gradually increase the phase difference that the machine operates in each case, we determined the operating range that the SSR provides cooling power. Secondly, studying the results, we determined that this range is from $160 \sim 180^\circ$ and the cooling abilities are eliminated when the destructive parallel flow towards the system heaters is appeared. Also, that the best behaviour efficiency wise, is appeared just after the destructive parallel flow has ended at 160° with $COP = 57.87\%$. After that, according to the data and by using a polynomial regression, we described the equation that correlates the COP with the phase difference for the 3D model. Finally, we compared the results of the 3D model with the 1D and Schmidt model, concluding that COP in Schmidt model is always greater from the other two under all circumstances and that in both 3D and 1D models COP follows a parabolic path with a minimum value at approximately 184° .

Συμπέρασμα

Έπειτα από τις μελέτες που έγιναν, προέκυψε ότι μία SSR μπορεί να παρέχει ψυκτική ισχύ όταν λειτουργεί από 160° έως 180° εκτός φάσης. Πέρα από αυτό το εύρος, η ψυκτικές ικανότητες εξαλείφονται λόγω της ύπαρξης ομορροής στον εναλλάκτη προς τα «hot boxes» ρων μηχανών. Επίσης υπολογίστηκε ότι το βέλτιστο σημείο λειτουργίας είναι αμέσως μόλις παύει να υπάρχει η ομορροή, δηλαδή στις 160° όπου η μηχανή λειτουργεί με βαθμό απόδοσης COP 57.87%. Τέλος όπως προέκυψε από τις συγκρίσεις με τα μοντέλα 1D – Schmidt, συμπεράναμε ότι ο βαθμός απόδοσης στο Schmidt μοντέλο είναι, ανεξαρτήτως τις μοίρες λειτουργίας, πάντοτε μεγαλύτερος από τα άλλα δύο όπως το περιμέναμε καθώς επίσης και ότι η συσχέτιση του COP με την διαφορά φάσης ακολουθεί και στο 3D και στο 1D ένα παραβολικό προφίλ με ελάχιστη τιμή περίπου στις 184° .

Bibliography/ References

- [1] J. G. Brisson & A. B. Patel, “A Simple Model for a superfluid Stirling Refrigerator at High Operating Temperature”, Journal of Low Temperature Physics. Vol 116, Nos 5/6, 1999
- [2] A. B. Patel, “The Development of High Cooling Power and Low Ultimate Temperature Superfluid Stirling Refrigerators”, PhD Thesis, MIT, 1999
- [3] F. Rondeaux et al., “Thermal Conductivity Measurements of Epoxy Systems at Low Temperature”, Proceedings Cryogenics Engineering Conference, Madison, Wisconsin, USA, 2001
- [4] M. Barucci et al., “Low temperature Thermal conductivity of Kapton and Upilex”, Cryogenics, vol. 40, pp.145-147, 2000
- [5] V. Kotsubo, G. Swift, “Superfluid Stirling-Cycle Refrigeration Below 1 Kelvin”, Journal of Low Temperature Physics, Vol. 83, Nos. ¾, 1991
- [6] G. Chaudhry, “Thermodynamic Properties of liquid 3He-4He mixtures between 0.15 K and 1.8 K”, PhD Thesis, MIT, 2009
- [7] ANSYS Inc, “ANSYS Fluent UDF Manual”, Release 15.0, Southpointe, 275 Technology Drive, Canonsburg, November 2013
- [8] G-R. Domenikos, “Thermophysical Modeling of Cryogenic Helium”, Diploma Thesis, NTUA, 2017
- [9] G-R. Domenikos, E. Rogdakis, P. Bitsikas, “Computational Analysis of Cryogenic Stirling Refrigerator”, International Mechanical Engineering Congress & Exposition, Salt Lake City, Utah, USA, 2019
- [10] G-R. Domenikos, E. Rogdakis, P. Bitsikas, “THERMODYNAMIC MODELLING OF SUPERFLUID STIRLING CRYOCOOLERS”, International Mechanical Engineering Congress & Exposition, Salt Lake City, Utah, USA, 2019
- [11] E. Rogdakis, P. Bitsikas, G. Dogkas, “Numerical Simulation of Prime Mover Stirling Engine by Finite Volume Method”, International Stirling Engine Conference, Newcastle, UK, pp451-464, 2016