THE DISSERTATION ENTITLED

Thermodynamic modelling and optimization of Modified Rankine cycle

SUBMITTED TO

INSTITUTE OF ENGINEERING AND TECHNOLOGY, LUCKNOW

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE

AWARD OF THE DEGREE OF

BACHELORS OF TECHNOLOGY

IN

MECHANICAL ENGINEERING

SUBMITTED BY

AMIT SAGAR (1505240011)

MAHEEP DWIVEDI (1505240031)

NAMIT KASERA (1505240035)

SAMBHRANT MAURYA (1505240048)

VAIBHAV SINGH (1505240057)

GUIDED BY

DR. SHAILENDRA SINHA

2018-2019

Mechanical Engineering Department

Institute of Engineering & Technology

(An autonomous constituent of Dr. A. P. J. Abdul Kalam Technical University)

Lucknow–226021, Uttar Pradesh, India.

Certificate

This is to certify that the dissertation entitled **"Performance analysis and optimization of Modified Rankine cycle"** submitted by **Amit Sagar (1505240011), Maheep Dwivedi (1505240031), Namit Kasera (1505240035), Sambhrant Maurya (1505240048) and Vaibhav Singh (1505240057)** in partial fulfilment of the requirement for the award of the degree in **BACHELORS OF TECHNOLOGY** in **MECHANICAL ENGINEERING** of the Institute of Engineering and Technology, Lucknow is a record of their own work carried out under my supervision. The matter embodied in this dissertation has not been submitted elsewhere for a degree or diploma in any other university.

Dr Shailendra Sinha

Professor

Department of Mechanical Engineering

(Guide)

2018-2019

Mechanical Engineering Department

Institute of Engineering and Technology

Lucknow–226021, Uttar Pradesh, India.

Acknowledgements

We would like to express my sincere gratitude, regards and thanks to respected **Dr Shailendra Sinha,** Professor, Department of Mechanical Engineering, IET Lucknow, for providing valuable insights, setting up clear and well-defined goals during the course of this work, who is apart from being our guide has been a great mentor and motivator. It is one of our achievements to work under his guidance. He is constant source of encouragement and momentum that any intricacy becomes simple. We gained a lot of invaluable guidance and prompt suggestions from him during my entire dissertation work. We remain indebted to him forever and we take pride to work under his guidance.

Our thanks are due for all our Colleague and Friends, who were always beside us and maintained our high spirit throughout the course of the study

At last, the utmost sense of gratitude is felt towards our parents and family. We humbly thank God for every success in my life.

AMIT SAGAR

MAHEEP DWIVEDI

NAMIT KASERA

SAMBHRANT MAURYA

VAIBHAV SINGH

Abstract

The Rankine cycle is a very important cycle in the power plant industry. Most of the steam powered power plants run on the Rankine cycle and many times it is required to tweak the performance of the Rankine cycle so that maximum efficiency can be obtained. For this purpose, the ideal Rankine cycle can be modified to incorporate reheating or regeneration or a combination of both. Pressures at which reheating or regeneration is carried out effects the performance of the plant.

Improving the efficiency of thermodynamic cycles plays a fundamental role in reducing the cost of thermal power plants. In this project a computer code has been developed for studying the effect of different parameters on the efficiency of the Rankine cycle. This project intends to model the Rankine cycle thermodynamically and find the optimal conditions for attainment of maximum efficiency for the given parameters.

In this project thermodynamic modelling and optimization of a combination of Rankine cycle with reheat and regeneration is done for single level of reheat and two levels of open type regeneration. The procedure is general in form and additional constraints such as the steam qualities at the exits of the different turbine stages are considered.

Contents

List of Figures

Nomenclature

NOTATIONS

h: Specific enthalpy, KJ/Kg

s: Specific entropy, KJ/Kg K

q: Specific heat, KJ/Kg

T: Temperature, K

η: Efficiency

η_B: Boiler efficiency

η_i : Isentropic efficiency

η_m: Mechanical efficiency

T_{bp}: Boiling point temperature, K

$P_{critical}$: Critical Pressure, Mpa

$P_{maximum}$: Max. pressure, Mpa

Q_{out}: Heat rejected, KJ/Kg

Q_{in}: Heat input, KJ/ Kg

$Q_{heating}$: Heat added in the economizer

Q_{evap}: Heat added in the evaporator

Q_{sht}: Heat added in the superheater, KJ/Kg

Q_{rej}: Heat rejected, KJ,

V: Specific volume

SUBSCRIPTS

B: Boiler

C: Condenser

I: Isentropic

in: Input

out: Output

P: Pump

CHAPTER 1

INTRODUCTION

1.1 The Simple Rankine cycle

The **Rankine cycle** is a model used to predict the performance of steam turbine systems. It was also used to study the performance of reciprocating steam engines. The Rankine cycle is an idealized thermodynamic cycle of a heat engine that converts heat into mechanical work while undergoing phase change. It is an idealized cycle in which friction losses in each of the four components are neglected. The heat is supplied externally to a closed loop, which usually uses water as the working fluid. It is named after **William John Macquorn Rankine**, a Scottish polymath and Glasgow University professor.

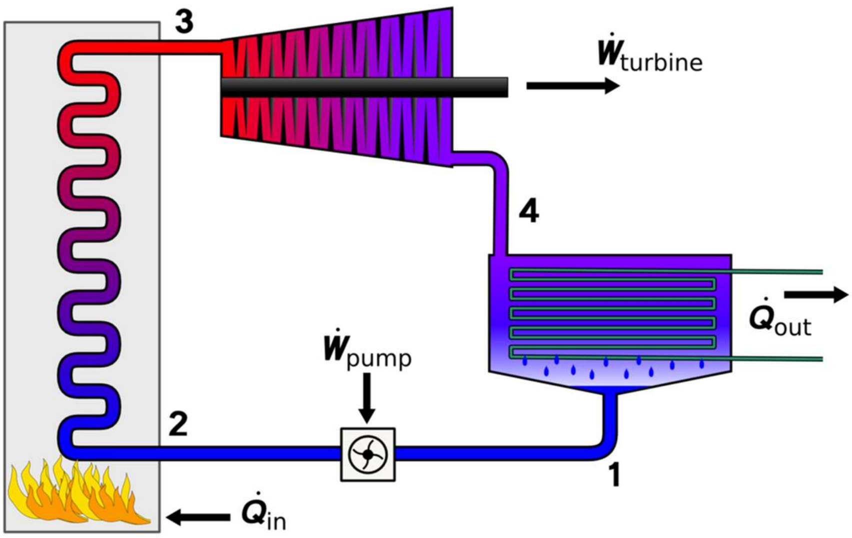


Figure 1-1: Physical layout of a plant running on Rankine cycle

1.1.1 Thermodynamic analysis of Simple Rankine Cycle

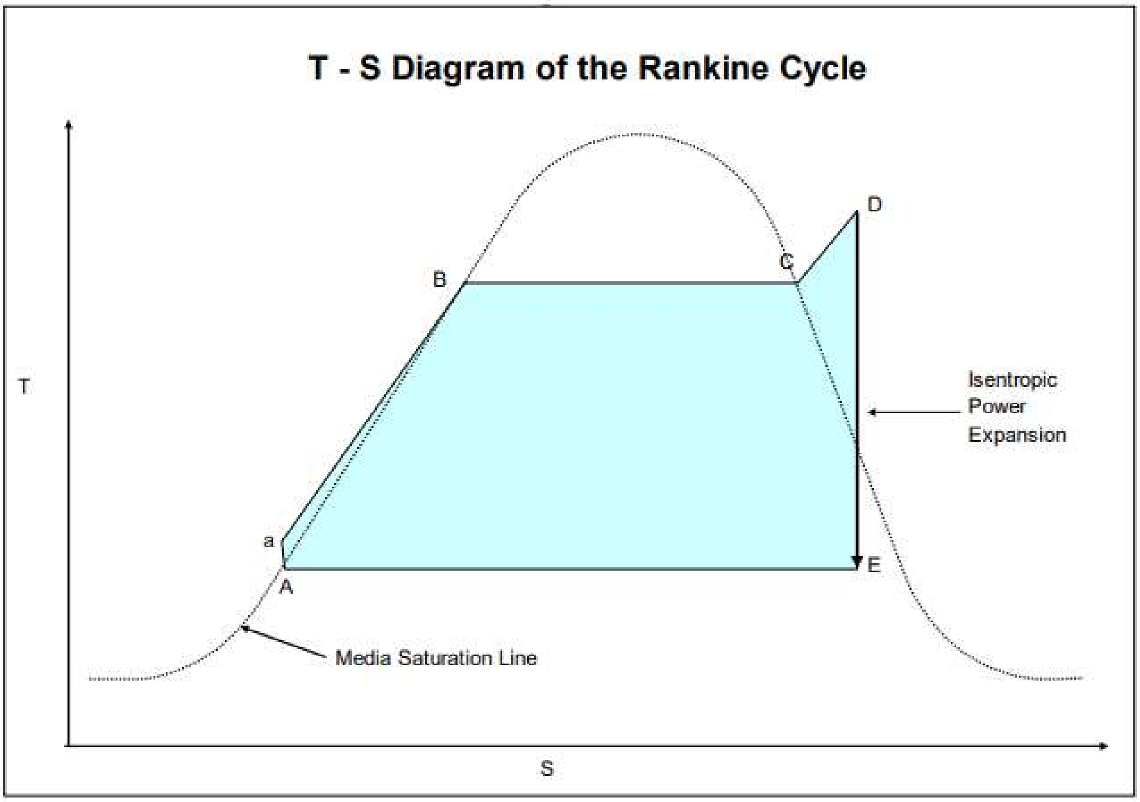


Figure 1-2 T-s diagram of simple Rankine cycle

Heat is added to the cycle in water heating, ($Q_{heating}$) the HP water along line (A - B). More heat is added to evaporate the liquid, (Q_{evap}) along line (B - C), while a third amount of heat is added (Q_{sht}) along line (C - D), to superheat the saturated steam. After the power expansion work (W) done by the cycle along line (D - E), the remaining latent heat in the low-pressure exhaust steam is rejected (Q_{rej}) out of the cycle by water cooled condensing along line (E - A). The power developed by the cycle is represented by the total enclosed (blue) area in figure 1-2, and heat rejection may be judged by the length of line (E - A) on the diagram. The thermodynamic cycle efficiency is given by:

$$\eta_{rankine} = \frac{W_{power}}{(Q_{heating} + Q_{evap} + Q_{sht})}$$

and because the sum of all energy entering (+) and leaving (-) the cycle adds to zero:

$$Q_{input} = (Q_{heating} + Q_{evap} + Q_{sht}) = W_{power} + Q_{rej}$$

$$\eta_{rankine} = \frac{W_{power}}{(W_{power} + Q_{rej})}$$

while it can be shown that the ideal, or Carnot efficiency of a heat engine operating between the same high temperature reservoir (T_{hot}), and low temperature reservoir (T_{cold}), is limited to a maximum of:

$$\eta_{carnot} = \frac{(T_{hot} - T_{cold})}{T_{hot}}$$

Maximum (Carnot) efficiency would be attained when all heat input is done at a single high temperature, unlike the Rankine cycle.

1.2 Thermodynamic analysis of single reheated irreversible Rankine cycle

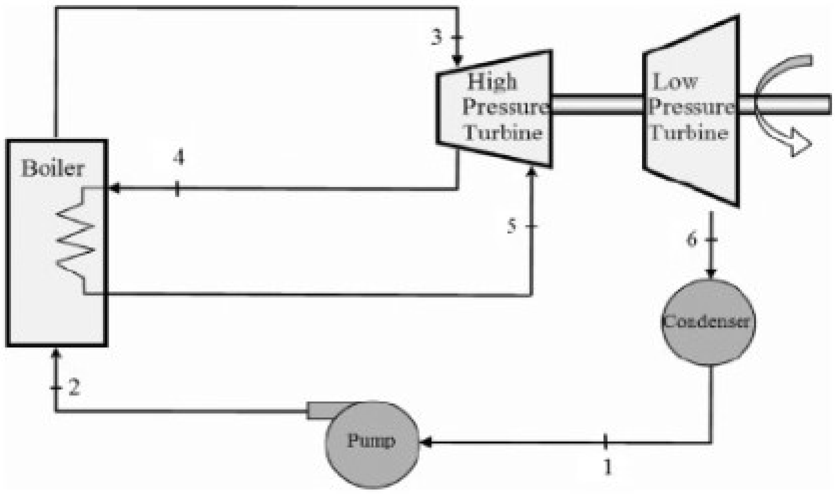


Figure 1-3 Schematic of single Reheat Rankine cycle plant

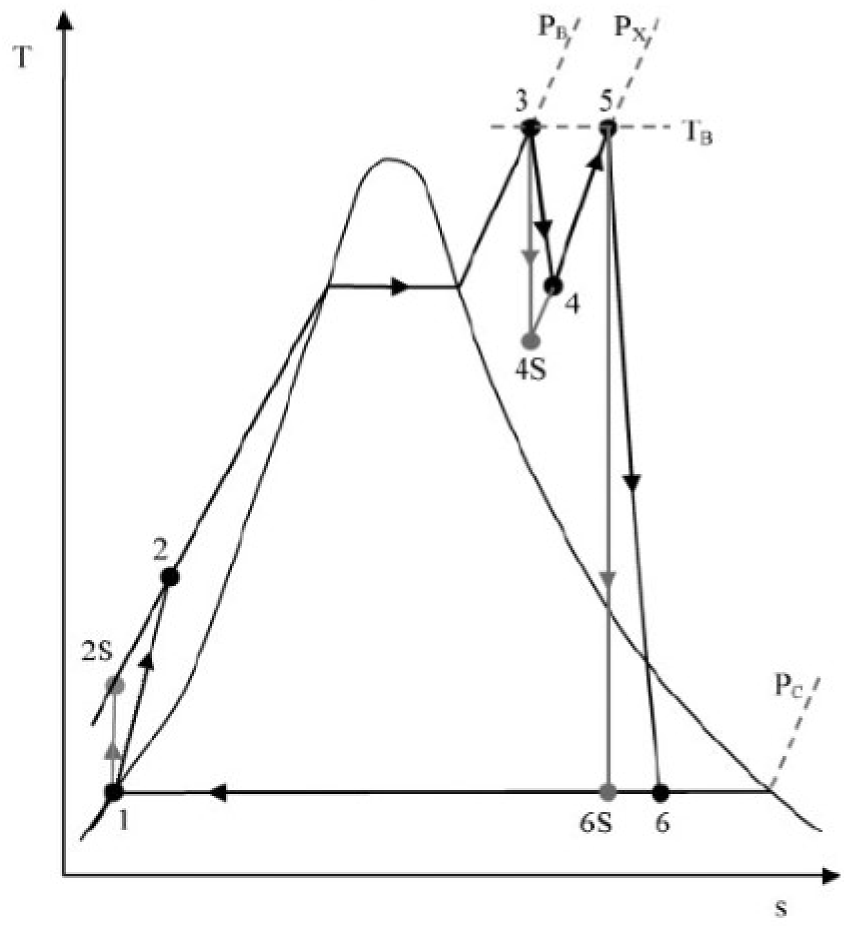


Figure 1-4 T-s diagram of single Reheat Rankine cycle

The flow and temperature–specific entropy T–s diagrams for the thermodynamic processes of a standard irreversible single reheat Rankine cycle operating with water are shown in Fig. 1-4. In the diagram, process 1–2S is an isentropic (reversible adiabatic) work input of the pump, while process 1–2 is an irreversible adiabatic process that takes into account the internal irreversibilities in the real cycle process. The heat addition occurs in two steps. Processes 2–3 and 4–5 are heat additions in the boiler by heating and reheating respectively. Processes 3–4S and 5–6S are isentropic work outputs of high-pressure turbine (HPT) and low-pressure turbine (LPT), while processes 3–4 and 5–6 take into account internal irreversibilities in the real Rankine cycle. A constant temperature heat rejection process by the condenser 6–1 completes the cycle. The heat added by heating and reheating in the boiler during the constant pressure processes 2–3 and 4–5 is:

$$q_{\text{in}} = \frac{(h_3 - h_2) + (h_5 - h_4)}{\eta_{\text{B}}} \tag{1}$$

and the heat rejection process in the condenser during the constant process 6–1 is

$$q_{out} = h_6 - h_1 \qquad (2)$$

The net work output and thermal efficiency of the single reheated Rankine cycle can be written respectively as:

$$w_{net} = w_{in} - w_{out} = [(h_3 - h_{4S}) + (h_5 - h_{6S}) - w_{pump}]\eta_i\eta_m \quad (3)$$

$$\eta_{th} = \frac{w_{net}}{q_{in}} = 1 - \frac{q_{out}}{q_{in}}$$
$$= \left[\frac{(h_3 - h_{4S}) + (h_5 - h_{6S}) - w_{pump}}{(h_3 - h_2) + (h_5 - h_4)}\right]\eta_i\eta_m\eta_B \qquad (4)$$

where the work input to the pump is given as:

$$w_{pump} = v_1(P_2 - P_1) \qquad (5)$$

1.3 Thermodynamic analysis of Regenerative Rankine cycle

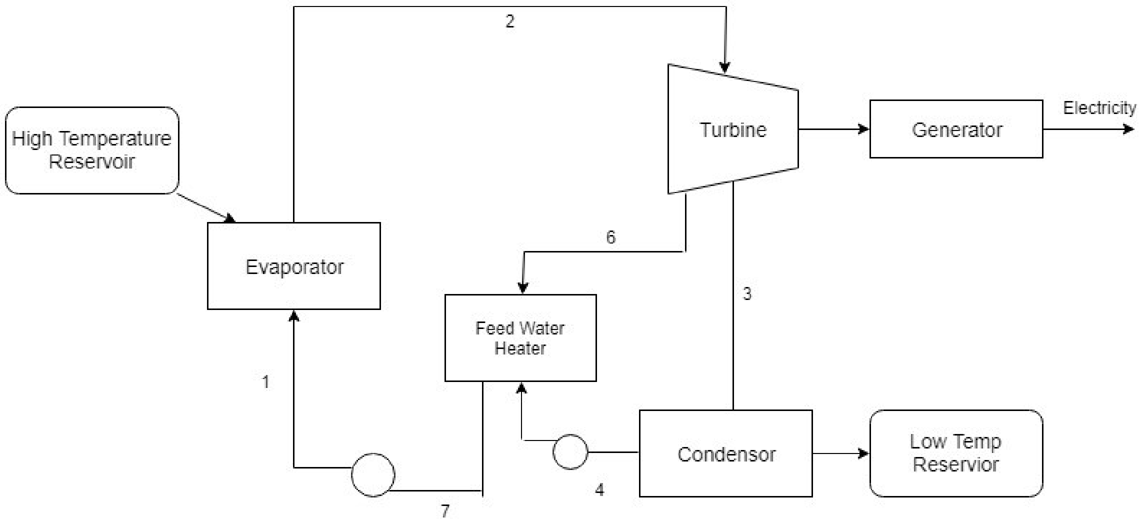


Figure 1-5 Schematic of the Regenerative Rankine Cycle

As observed in figure 1-5 there are 7 processes in the regenerative Rankine cycle. When steam from boiler is sent into turbine, not all of it is expanded in it. A part of it bled out (6) to be used in feed water heater for regeneration process. The remaining steam is expanded in the turbine and the steam obtained at the turbine exit (7) is then passed into condenser where it goes through condensation process (7-1), which is then pumped into feed water heater (1-2). In feed water heater this working fluid is then mixed with bled out steam for regeneration process which rises the temperature of working fluid through heat addition inside feed water heater. This working fluid is then pumped into boiler (3-4) where heat addition takes place and steam obtained through boiler (5) is then sent to turbine for expansion (5-7).

The heat addition in feed water heater is internal and hence it is not reflected in total heat addition, along with heat addition in boiler, while calculating cycle efficiency.

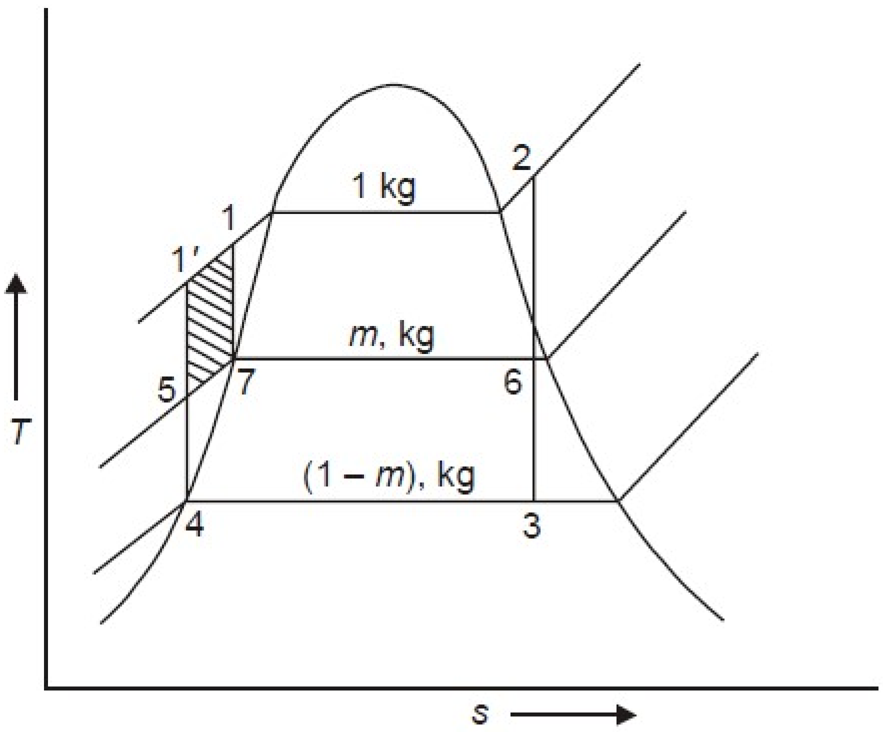


Figure 1-6 T-s representation for regenerative Rankine cycle with single open type feed water heater

For the regenerative cycle considered, with unit mass of steam leaving boiler and '*m*' kg of steam bled out for feed water heating:

Steam turbine work = $(h_2 - h_6) + (1 - m) \cdot (h_6 - h_3)$

Pump work = $(1 - m) \cdot (h_5 - h_4) + 1 \cdot (h_1 - h_7)$

Net work = $\{(h_2 - h_6) + (1 - m) \cdot (h_6 - h_3)\} - \{(1 - m) \cdot (h_5 - h_4) + (h_1 - h_7)\}$

Heat added = $1 \cdot (h_2 - h_1)$

$$\eta_{\text{regenerative}} = \frac{\{(h_2 - h_6) + (1 - m)(h_6 - h_3)\} - \{(1 - m)(h_5 - h_4) + (h_1 - h_7)\}}{(h_2 - h_1)}$$

1.4 Reheat-Regenerative Rankine cycle

Concept of regenerative Rankine cycle with feedwater heater will provide better positive result in respect of increase in thermal efficiency of the Rankine cycle or in simple way we can say that thermal efficiency of the steam power cycle will be improved quite good by using the concept of regenerative Rankine cycle with feedwater heater as compared with reheat cycle.

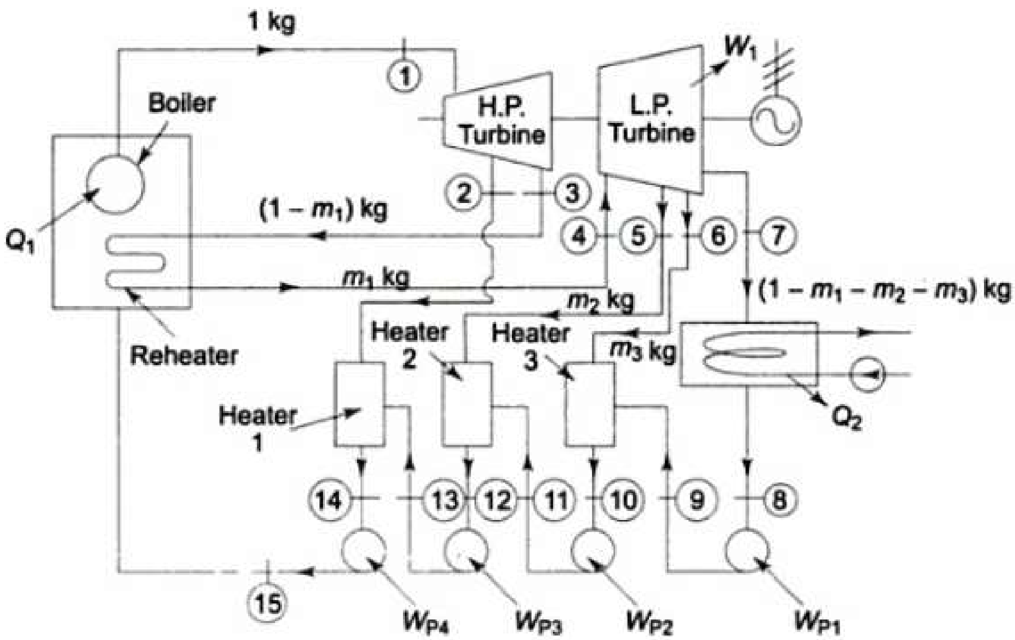


Figure 1-7 Schematic of reheat-regenerative cycle plant

As we can see in block diagram, high pressure and high temperature steam enters to the high pressure turbine at state 1 and as we are also considering here the concept of regeneration hence we must note it here that all steam will not be expanded through the high pressure turbine up to pressure corresponding to state 3 but also certain quantity of steam will be extracted from the high pressure turbine and its state is displayed by state 2 in figure 1-7.

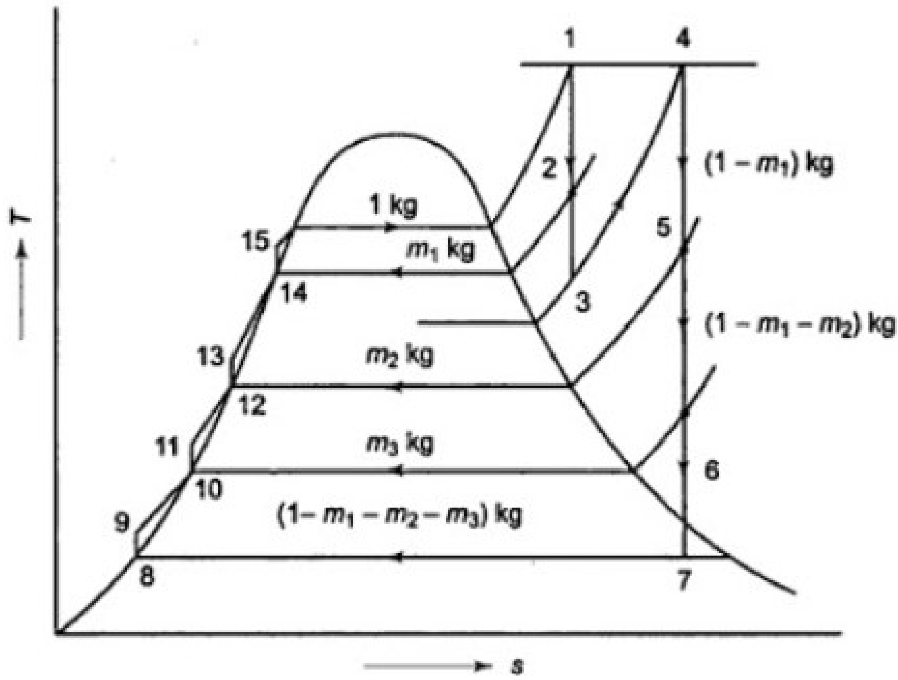


Figure 1-8 T-s diagram for reheat-regenerative cycle

Steam coming from reheater will enter to the low-pressure turbine at state 4 and similarly as studied earlier certain quantity of steam will be extracted here from low pressure turbine before the complete expansion of steam up to the condenser pressure.

So as shown in figure, steam will be extracted from low pressure turbine at state 5 and state 6 and rest quantity of steam will be expanded up to the condenser pressure i.e. up to state 7. Steam will enter to the condenser at state 7 and will go undergo condensation process during the process 7 to 8. Now working fluid will be pumped with the help of feed pump by process 8-9.

Now extracted steam at state 6 and working fluid (pumped by the feed pump WP_1) at state 9 will exchange heat energy with each other in to the feedwater heater 3.

Therefore, working fluid will be heated in to the feedwater heater and we have displayed this process of heat energy addition to the working fluid in feedwater heater 9-10.

Similarly working fluid will secure heat energy during the process 11-12 and 13-14 in feedwater heater 2 and feedwater 1 respectively.

Process 15-1 will indicate here the heat energy addition to the working fluid in boiler and we will consider this heat energy addition as input energy addition. Working fluid will also secure heat energy in reheater during the process 3-4 and hence heat energy added during the process 3-4 will also be taken as input heat energy.

Now, we have recently studied that heat energy will also be added to the working fluid during the process of feedwater heater but that heat energy addition will be internal as that heat energy addition will be done by the concept of regeneration process.

1.5 Thesis Outline

The present work is divided into following chapters:

Chapter 1. Introduction introduces basic concept related to the Rankine Cycle and the Modified Rankine Cycle.

Chapter 2. Literature Review contains the survey of various literatures related to the Rankine cycle. Number of literatures are discussed under this chapter which are very helpful for further understanding of the developments in Rankine cycle.

Chapter 3. Thermodynamic Modelling Algorithm presents a generalized flowchart-based algorithm on how to approach optimization-based scenarios of The Rankine Cycle in a C/C++ based environment.

Chapter 4. Analysis contains performance analysis and optimization of various modifications to the Rankine Cycle, viz. The Reheat Rankine Cycle, The Regenerative Rankine Cycle and The Reheat-Regenerative Rankine cycle where the number of reheats can vary from zero to one and number of feed water heaters can vary from zero to two.
Analysis is based on C++ and graphs are drawn with the aid of MS Excel.

Chapter 5. Results and Discussion: Presents the results of the current analysis and discusses the reasons for the observations and variations in the curves obtained in the analysis process.

1.6 Objectives of the present work

- To develop a C program to calculate the optimum parameters of Rankine/Modified Rankine cycle for maximum efficiency.

- To develop a thermodynamic model of the Rankine cycle and its modifications to study the variation of parameters like efficiency and dryness fraction with reheat/regeneration pressures.

CHAPTER 2

LITERATURE REVIEW

William John Macquorn Rankine [1] a Scottish engineer, advanced the study of heat engines by publishing the "*Manual of the Steam Engine and Other Prime Movers*" in 1859. Rankine developed a complete theory of the steam engine and indeed of all heat engines. Together with Rudolf Clausius and William Thomson (Lord Kelvin), he was a contributor to the thermodynamics, particularly focusing on the first of the three thermodynamic laws. William J.M. Rankine developed a complete theory of the steam engine in his famous manual of 1859. He was a founding contributor to the science of thermodynamics. His manuals of engineering science and practice were used for many decades after their publication in the 1850s and 1860s. He published several hundred papers and notes on science and engineering topics, from 1840 onwards.

Thomas Howard [2] built an engine working on the Rankine Cycle, with a design power of 24 hp (about 18 kW), worked for a brief period in Rotherhithe, Surrey, UK, as early as 1825–1826. On January 9th 1885, the British journal *The Engineer* published an article that describes the results of a test carried out by a commission of the US naval engineers on a launch engine. The commission made a rigorous comparison with the performance of the same engine in case steam was the working fluid.

Du Tremblay [3], around 1850, an engineer from Lyon, developed a binary (or cascading) heat engine using steam in the high-temperature engine and "ether" (probably diethyl ether, much more volatile than water) in the bottoming machine. After evaporation in the boiler and expansion in the cylinders, the steam released its thermal energy of condensation to the second engine, causing the evaporation of the ether which, expanding in another cylinder, produced additional work.

Habib et al. [5] studied the first and second law procedures for the optimisation of the reheat level in reheat regeneration thermal power plants. The procedure was used for a thermal power plant having two reheat pressure stages and open feed water heaters. He calculated and optimised the second law efficiency of the steam generator, turbine cycle and plant, taking into account the irreversibilities and some constraints, such as the steam qualities.

Dincer and Al-Muslim [6] performed a thermodynamic analysis of a single reheat Rankine cycle for the steam power plants based on the first and second laws of thermodynamic. He studied the energy and exergy efficiencies for different system parameters, such as boiler temperature, boiler pressure, mass fraction ratio and work output. In his study, the results of the efficiencies were compared with the real data and literature and found good agreement.

David A. Tillman [7] proved at Philo Station and then again at the Twin Branch Station the advances of the reheat cycle giving the promise of efficiency improvements to come. Reheat not only effectively reduced the penalty of the latent heat of vaporization in steam discharged from the low-pressure end of the turbine cycle, but it also improved the quality of the steam at the low-pressure end of the turbines by reducing condensation and the formation of water droplets within the turbine. While it was not immediately adopted after World War II, it was almost universally adopted by 1956.

Baumann [8] in 1930 analysed the development of steam cycle power during years, using high steam pressure and temperature, with the objective to improve the efficiency of the power plants. He analysed the regenerative cycle to improve the efficiency of Rankine cycle. In the regenerative cycle, the water entering in solar collectors is hotter than the Rankine cycle and becomes better for solar power plants. The condensate of Rankine cycle at low temperature had greater irreversibility when the water was mixing in the collectors and this decreased cycle efficiency. The steam extracted from the turbine at different stages to the feed water heater with small fractions of vapor released by the turbine, reduced the irreversibility associated with the exchange of energy in the feed water heaters. The vapor or water through feed water heaters increased the average temperature of the heat source.

Bejan [9] in 1988**,** showed that the efficiency of the regenerative cycle depends on the distribution of the turbine, that is, the temperature difference between a feed water heater and the heater adjacent.

Haywood [10] in 1949 had affirmed that the enthalpy difference in the output between each feed water heater and the heater adjacent must be constant to obtain a maximum efficiency.

Ying [11] in 1999 presented a new approach for solar power utilization, i.e. using solar heat to replace the extracted steam to heat the feed water heater in the regenerative cycle plant. They analysis of a three-stage regenerative cycle plant and shows that, by using solar energy, the work generated can be increased up to 30% while the ratio of the work generated by the saved steam to the exergy supplied by the solar heat reaches 101.28%. By using low-grade thermal energy to replace the extracted steam to heat the feed water heater in the regenerative Rankine power plant, the exergy merit index of the energy hits extremely high values and is far superior to the corresponding exergy efficiencies in other power systems with the same waste heat as the heat source alone, and the aided system can run more efficiencies than the conventional regenerative Rankine plant. Then, it can be observed that the utility the Regenerative cycle with feed water heaters is important way in solar power plants.

Weir [12] in 1999 also discussed the optimal distribution of feed water heater along the turbine for the regenerative cycle. In his work he distinguishes two types of extractions in the same cycle: extraction of a condensation turbine and a reheating turbine, which does not represent the proposal of this work.

Acar [13] conducted a research about reheat regenerative Rankine cycle, where he analysed a steam power cycle based on the second law by its implementation to the reheat regenerative Rankine cycle.

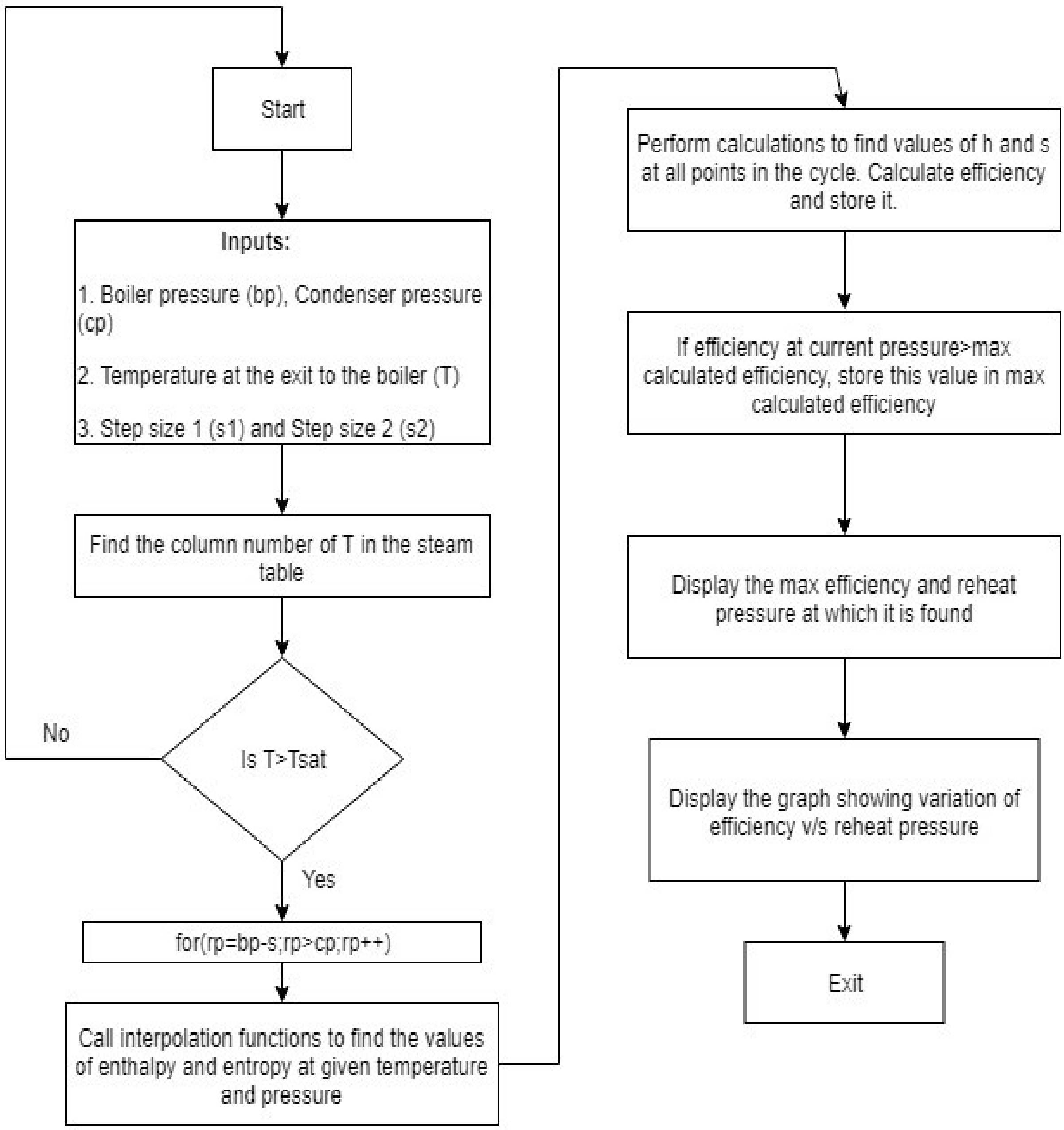
Retzlaff and Ruegger [14] explained the efforts of advancement in the state-of-the-art in steam turbine technology, which is related to the improvement in the thermodynamic efficiency by increasing the temperature and pressure at which heat is added to the power cycle. They investigated steam turbines for ultra-supercritical powerplants and reported heat rate improvement of single and double reheat steam cycles with ultra-supercritical steam conditions on their work.

Fraidenraich [15] in 2013 developed an analytical modelling of direct steam generation solar power plants and they use Rankine Cycle as the power block. They choice of evaporation temperature of 290˚C in their analyses that have the advantages of near-maximum turbine work with a relatively small pressure in the collectors. The use of regenerative cycle could improve the results of efficiency and the work. This work can give an overview on conditions could use the regenerative cycle.

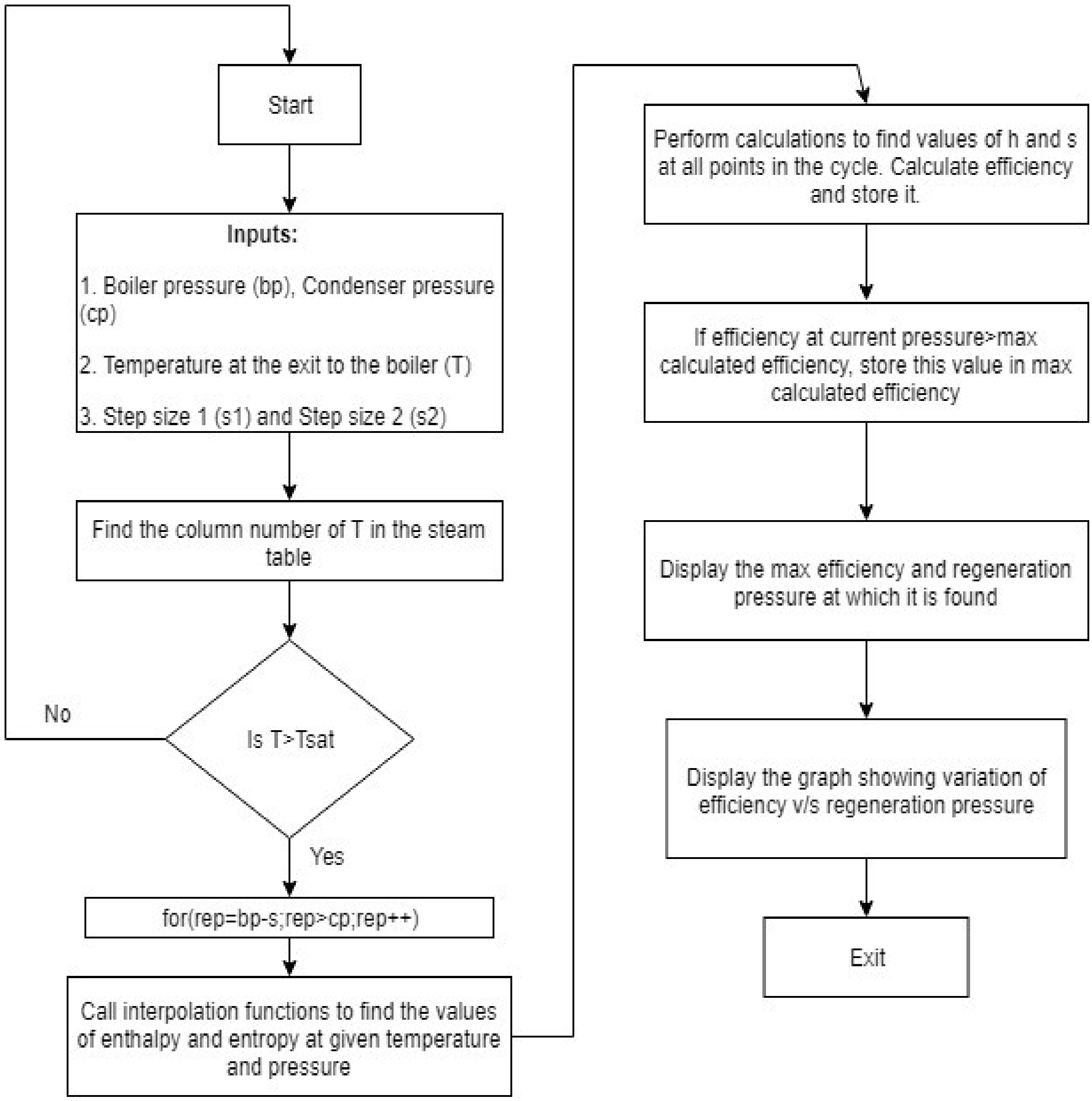
CHAPTER 3

THERMODYNAMIC MODELLING ALGORITHM FOR RANKINE/MODIFIED RANKINE CYCLE

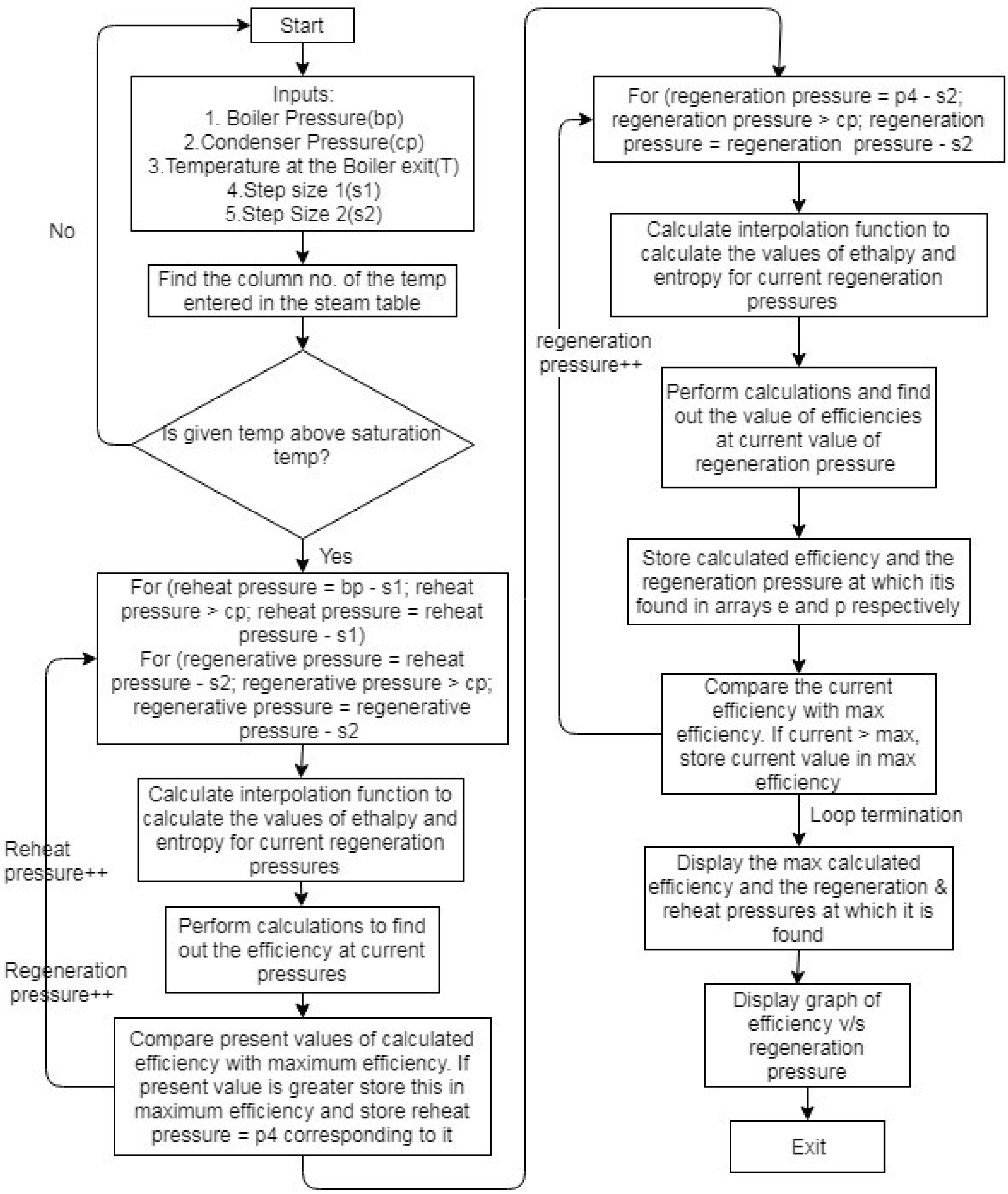
3.1 Algorithm for Single Reheat Rankine Cycle



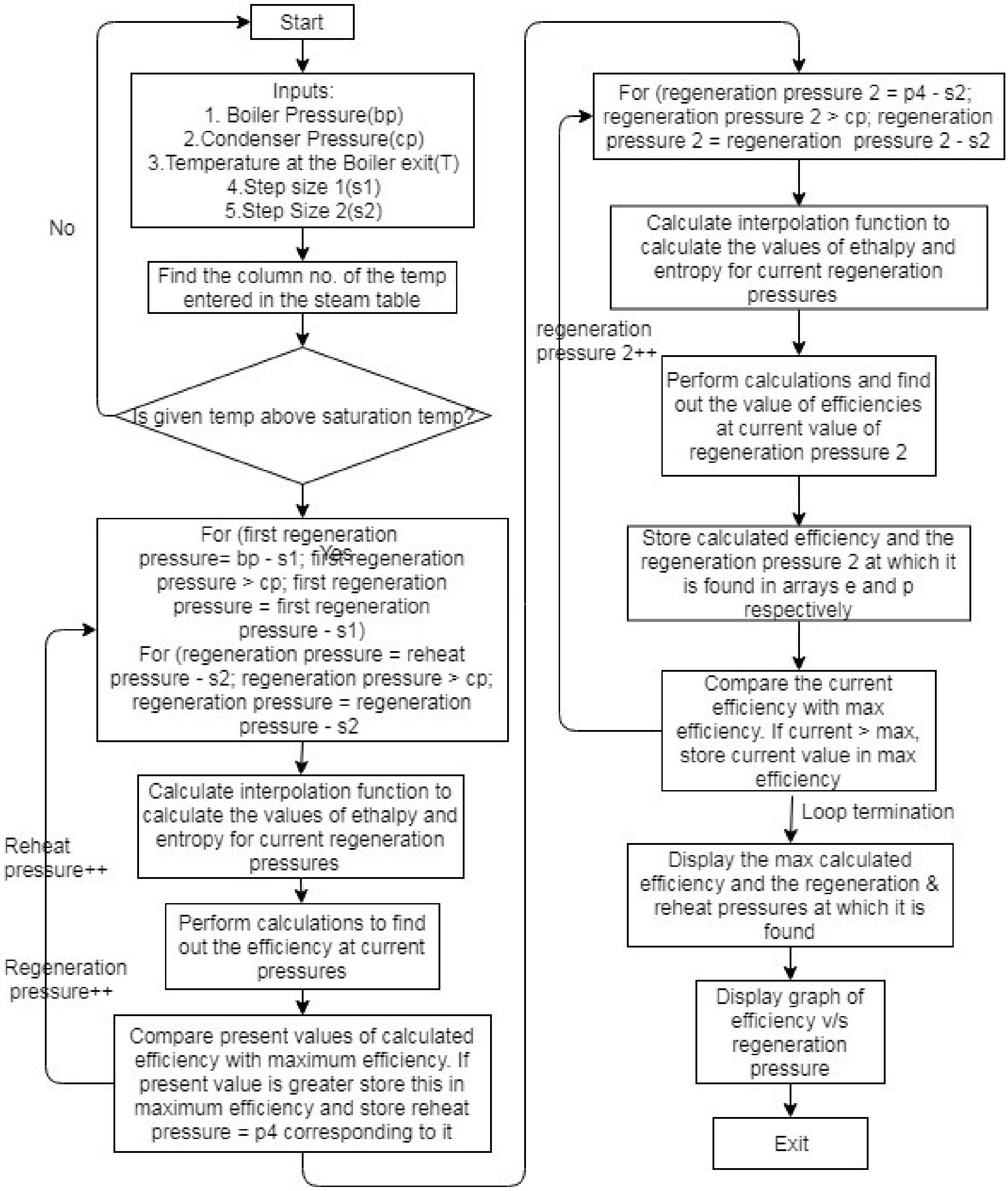
3.2 Algorithm for Single regenerative Rankine cycle



3.3 Algorithm for single reheat single regenerative Rankine cycle



3.4. Algorithm for double regenerative Rankine cycle



CHAPTER 4

ANALYSIS OF RANKINE/MODIFIED RANKINE CYCLE

Sample input:

Boiler pressure: 150 bar

Condenser pressure: 0.5 bar

Temperature at exit of boiler: 560ºC

4.1 Output of Simple Rankine cycle

Maximum efficiency obtained is 0.3794 and the steam dryness factor at maximum efficiency is 0.8387

4.2 Analysis of the effect of boiler temperature on efficiency when boiler pressure is kept constant at 150 bar in simple Rankine cycle.

The variation of efficiency with temperature at boiler exit keeping boiler pressure constant at is shown in the graph below.

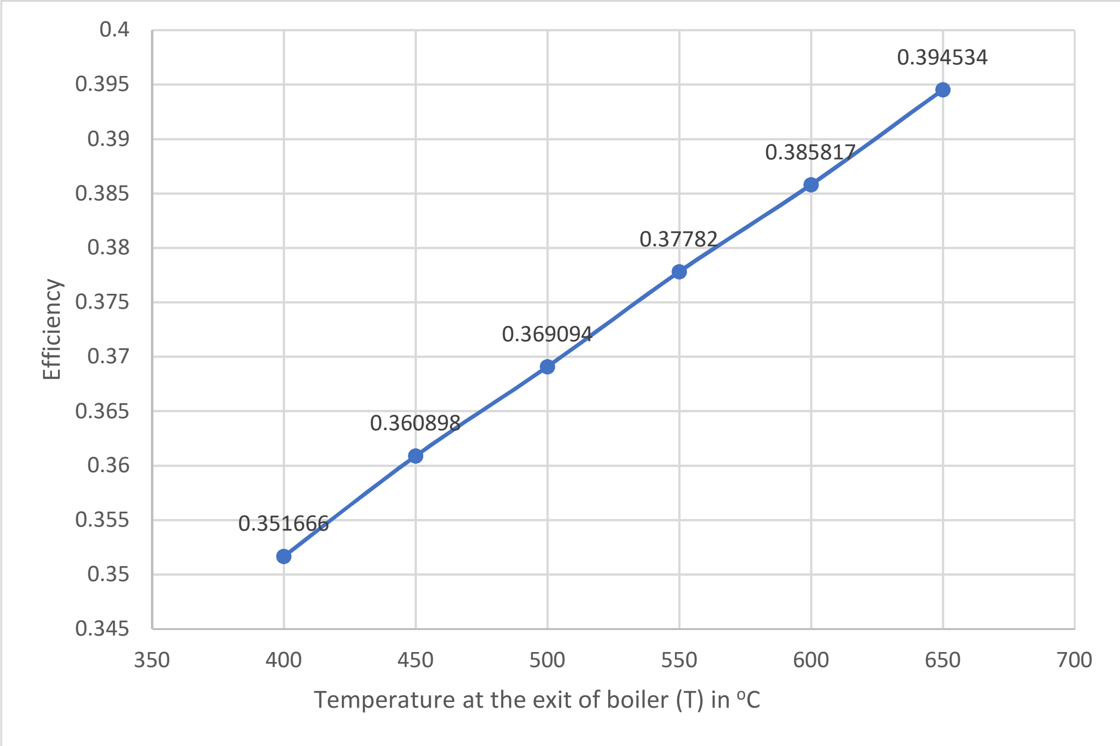


Figure 4-1 Efficiency v/s Temperature at boiler exit at constant Boiler Pressure

It can be observed from the graph above that the efficiency of the cycle increases with the increase in boiler exit temperature when boiler pressure is kept constant.

4.3 Analysis of the effect of boiler temperature on the dryness fraction of the steam at the exit of turbine/inlet to condenser when boiler pressure is kept constant in simple Rankine cycle.

The variation of the dryness fraction at the exit of turbine/inlet to condenser with respect to the temperature at boiler exit when boiler pressure is kept constant is shown in the graph below.

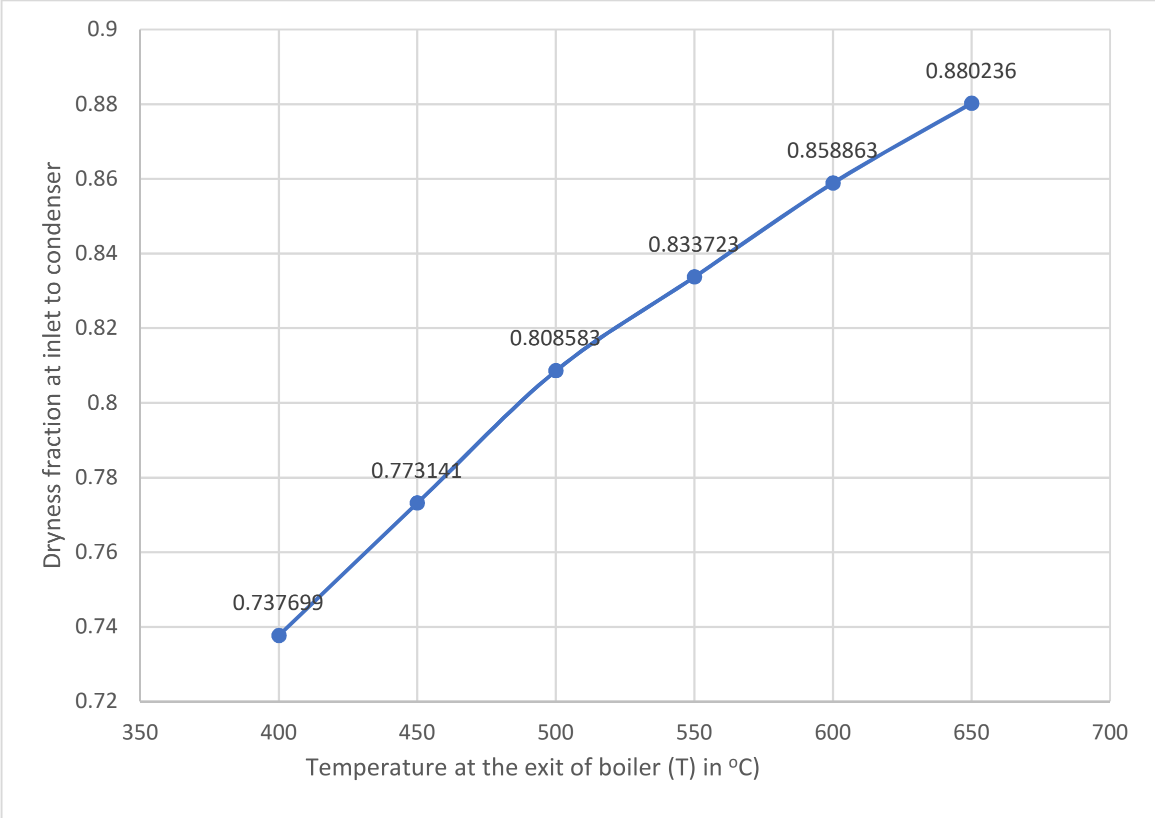


Figure 4-2 Dryness fraction (x) v/s Temperature at boiler exit(T) at constant Boiler Pressure

It can be observed from the graph above that the dryness fraction at the exit of turbine/entry to the condenser increases with increase in temperature at boiler exit when boiler pressure is kept constant.

4.4 Analysis of the effect of boiler pressure on the cycle efficiency when boiler exit temperature is kept constant in simple Rankine cycle.

The variation of efficiency with boiler pressure when boiler exit temperature is kept constant is shown in the graph below.

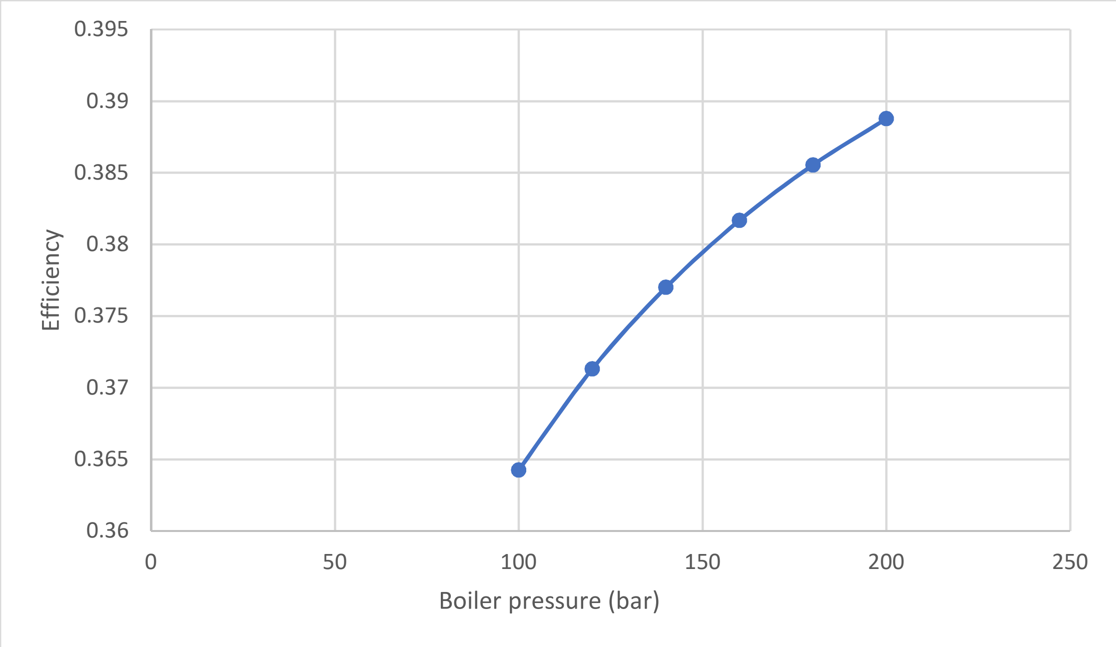


Figure 4-3 Efficiency vs boiler pressure at constant temperature at boiler exit

It can be observed that the efficiency of the cycle increases with the increase in boiler pressure from 100 bar to 200 bar and reaches a maximum of 0.388775 at 200 bar.

4.5 Analysis of the effect of boiler pressure on the steam dryness fraction at the exit of turbine/inlet to condenser when boiler temperature is kept constant at 560 degree Celsius in simple Rankine cycle.

The variation of dryness fraction with respect to boiler pressure when boiler exit temperature is kept constant is shown in the graph below.

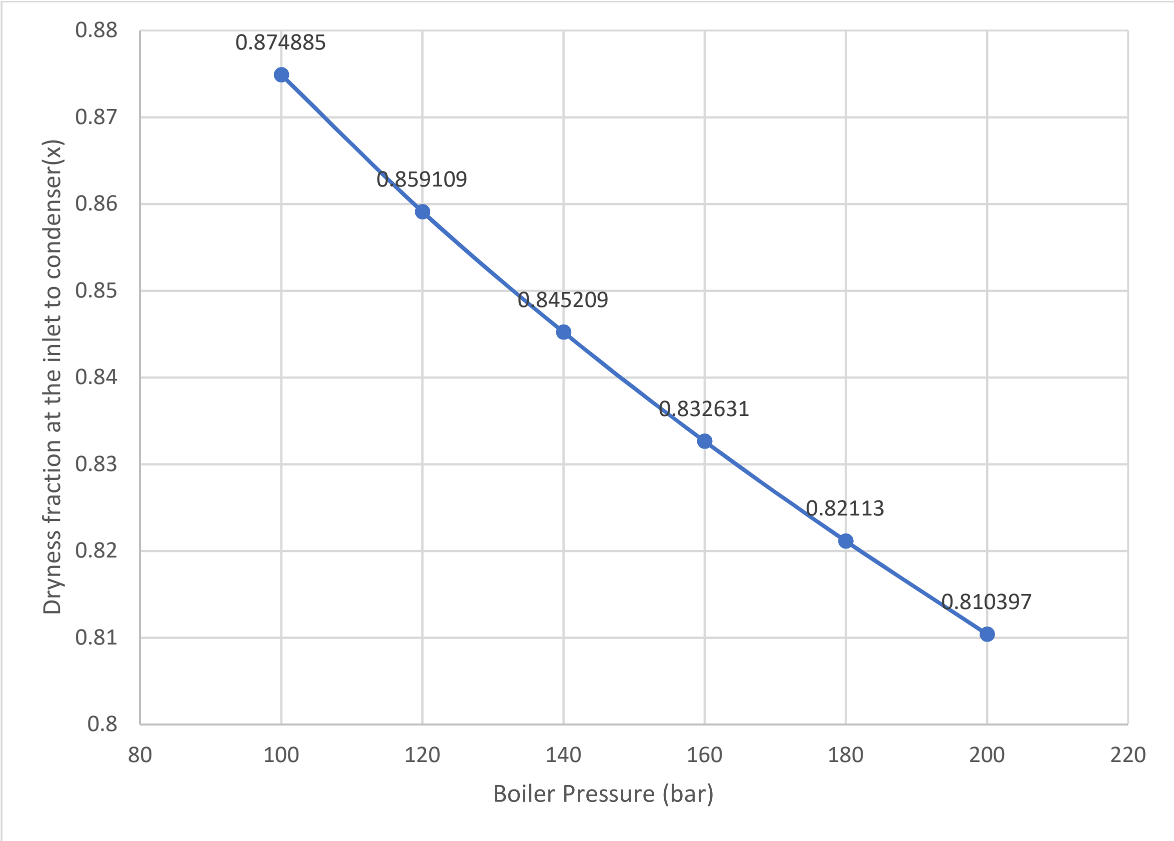


Figure 4-4 Dryness fraction (x) v/s Boiler pressure at constant temperature at the boiler exit

It can be observed from the graph above that the dryness fraction decreases when the boiler pressure is increased at a constant boiler exit temperature.

4.6 Analysis of the effect of reheat pressure on the cycle efficiency and dryness fraction at different boiler pressures when temperature at boiler exit is kept constant.

The variation of cycle efficiency with respect to reheat pressure at different boiler pressure, when the boiler exit temperature is kept constant, is shown in the graph below.

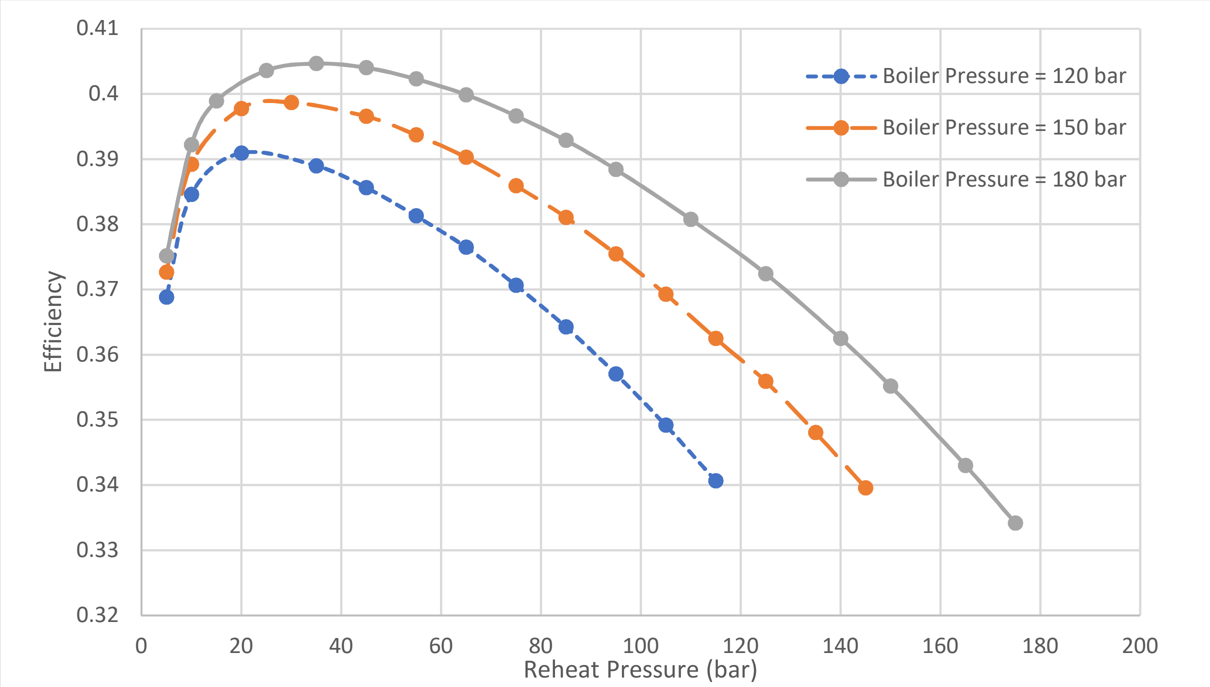


Figure 4-5 Efficiency v/s Reheat Pressure when temperature at the exit of boiler is constant (T)

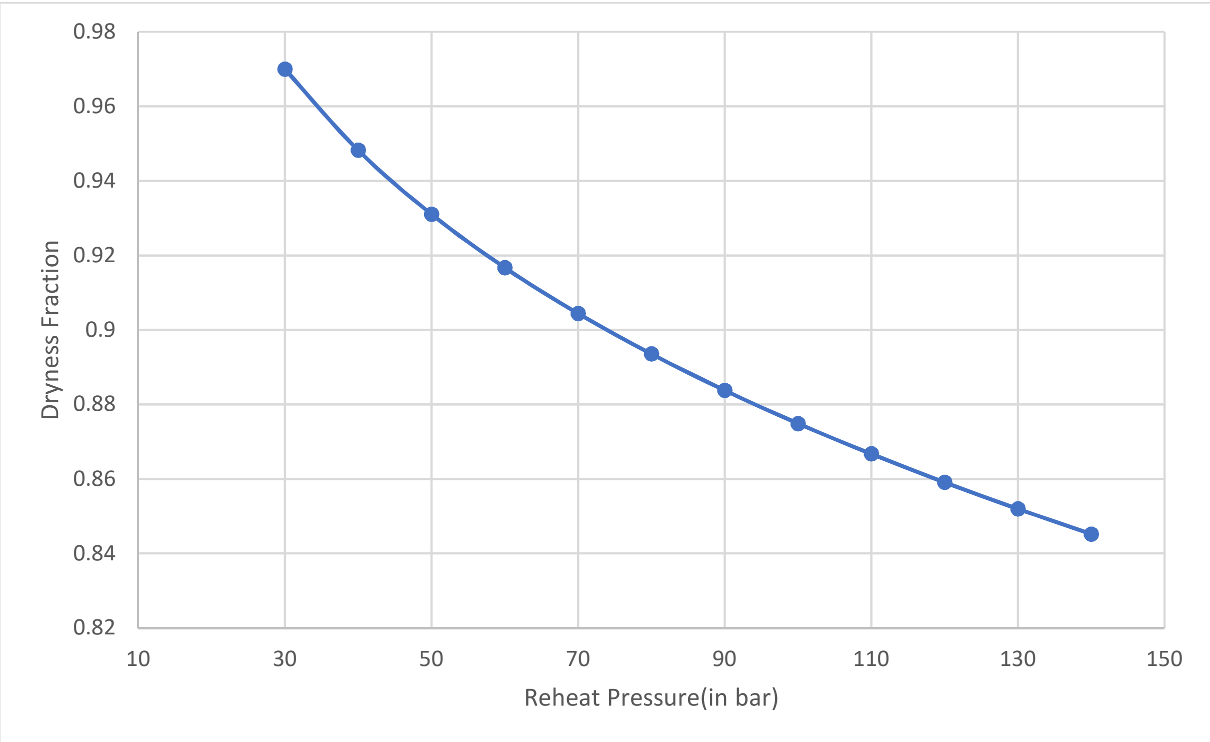


Figure 4-6 Dryness Fraction v/s Reheat Pressure

- It can be observed from the graph above that the cycle efficiency first increases, reaches a maximum at 0.398695 at 30 bar in all the three boiler pressures and then starts decreasing with increasing reheat pressure. The step size of 15 bar for reheat pressure is used to vary reheat pressure from 135 bar to 15 bar.

- It can also be seen that the peak of maximum efficiency shifts upward with the increase in boiler pressure.

4.7 Analysis of the effect of regeneration pressure (open type) on the cycle efficiency when the temperature at boiler exit is kept constant.

The variation of efficiency with the regeneration pressure when the temperature at boiler exit is kept constant is shown in the graph below.

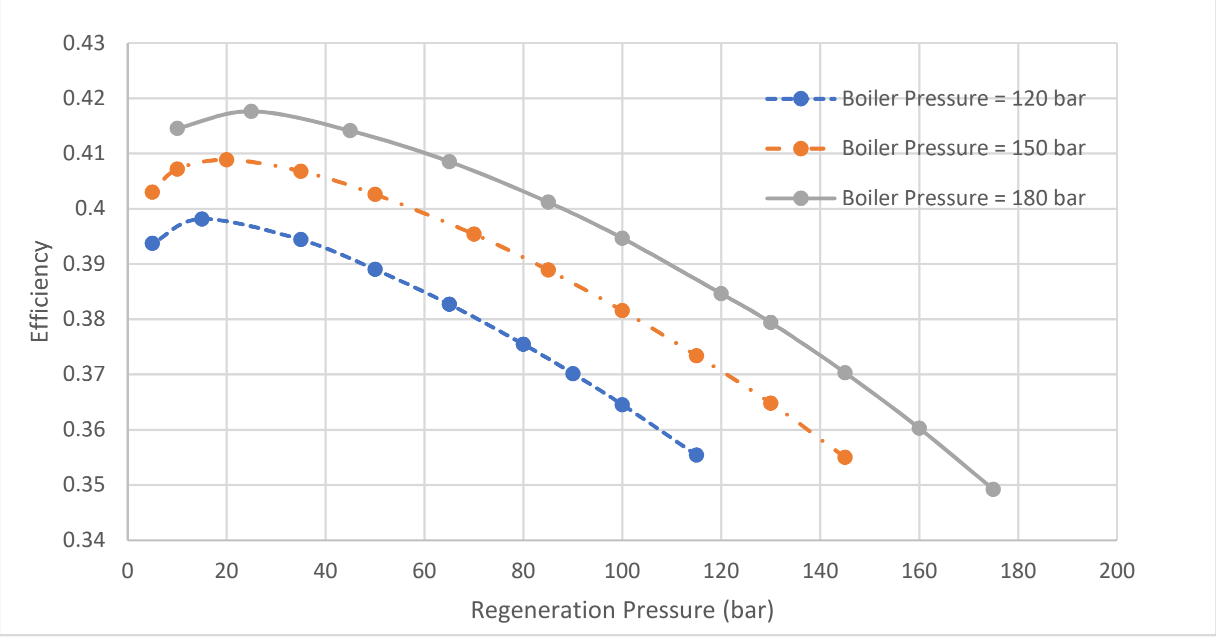


Figure 4-7 Efficiency v/s Regeneration Pressure when Temperature at boiler exit is constant

- It can be observed from the graph above that the cycle efficiency first increases, attains a maximum of 0.408861 at 20 bar regeneration pressure and then starts decreasing with the increasing regeneration pressure.
- The step size of 10 bar is used to vary regeneration pressure from 140 bar to 10 bar.
- It can also be seen that the maximum efficiency peak shifts upward with the increase in boiler pressure.

4.8 Analysis and optimization of single regenerative Rankine cycle (closed feed water heater type)

The variation of cycle efficiency with respect to regeneration pressure in case of closed type feedwater heater is shown in the graph below.

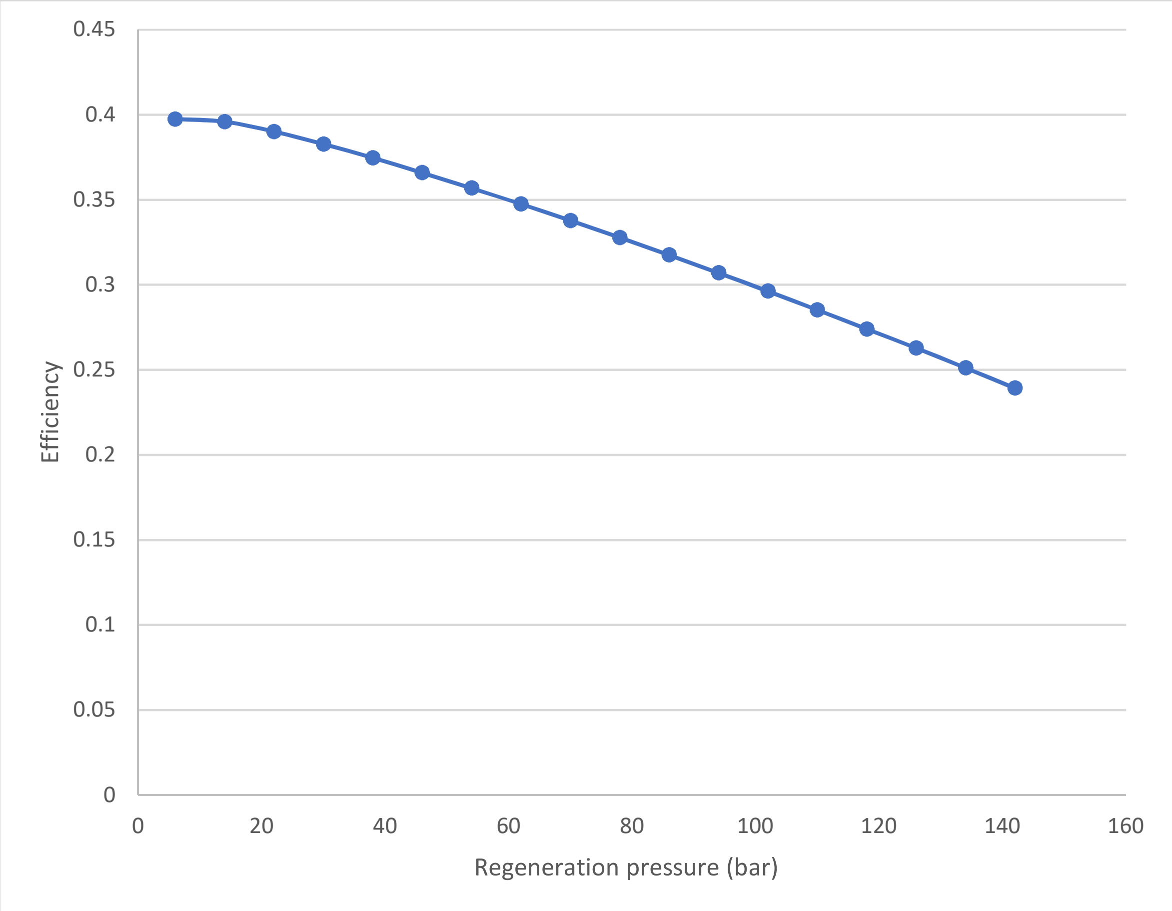


Figure 4-8 Efficiency v/s Regeneration Pressure

- The efficiency of regenerative Rankine cycle first goes on increasing as we decrease the regeneration pressure in step size of 8 bar from the boiler pressure. The efficiency becomes maximum at regeneration pressure=6 bar.

- The efficiency after reaching the peak point of 0.3974 decreases till we reach a step size above the condenser pressure.

- The value of dryness fraction at the entry to the condenser remains constant at all values of regeneration pressure.

4.9 Analysis and optimization of Rankine cycle with two open feed water heater arrangement

The variation of efficiency with respect to 2nd regeneration pressure while the 1st regeneration pressure is kept at the optimum, optimum - step size & optimum + step size is shown in the graph below:

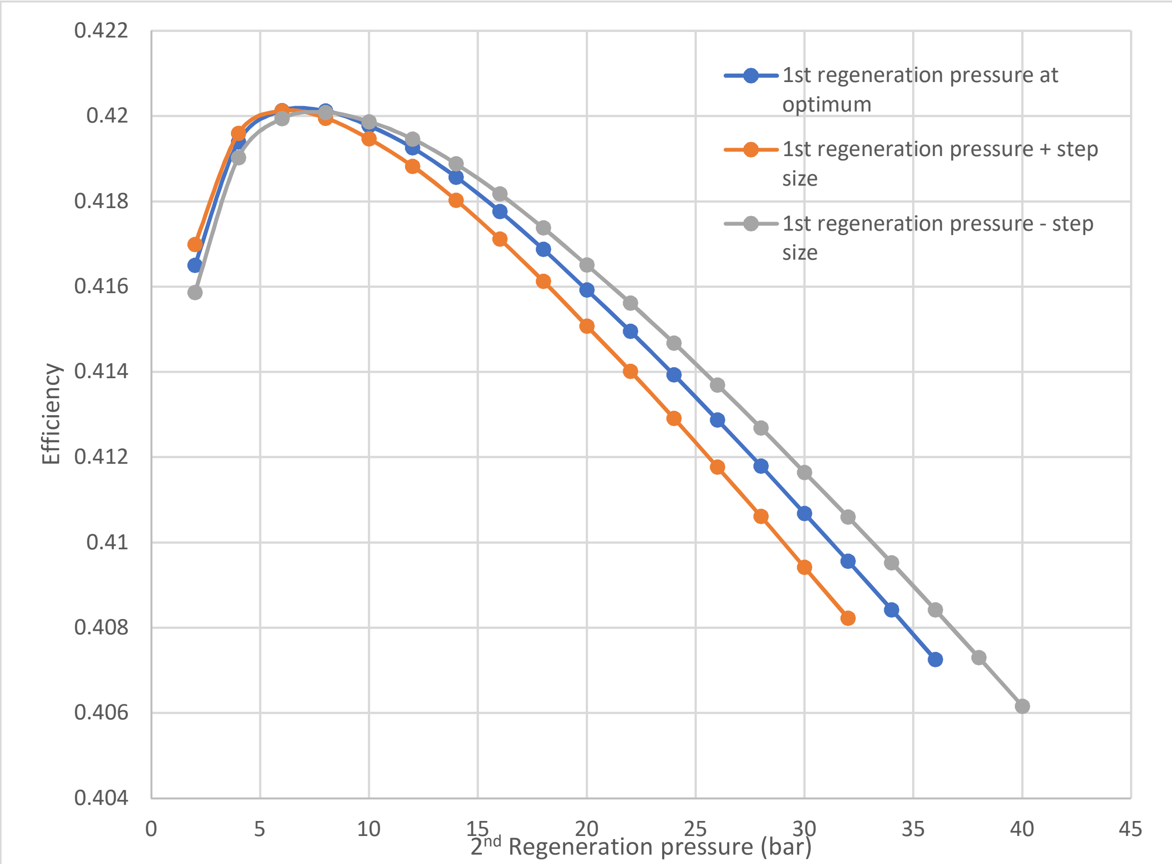


Figure 4-9 Variation of efficiency with 2nd Regeneration pressure

- The efficiency of regenerative Rankine cycle first goes on increasing as we decrease the regeneration pressure in step size of 4 bar from the boiler pressure for first regeneration pressure and step size of 2 bar from boiler pressure for second regeneration pressure. The efficiency becomes maximum at first regeneration pressure=38 bar and second regeneration pressure of 6 bar.

- The efficiency after reaching the peak point of 0.4201 decreases till we reach a step size above the condenser pressure.

- The value of dryness fraction at the entry to the condenser remains constant at all values of regeneration pressure.

4.10 Analysis and optimization of Rankine Cycle with single reheat and one open type feed water heater arrangement

The variation of efficiency with respect to regeneration pressure when reheat pressure is kept at optimum, optimum – step size & optimum + step size is shown in the graph below.

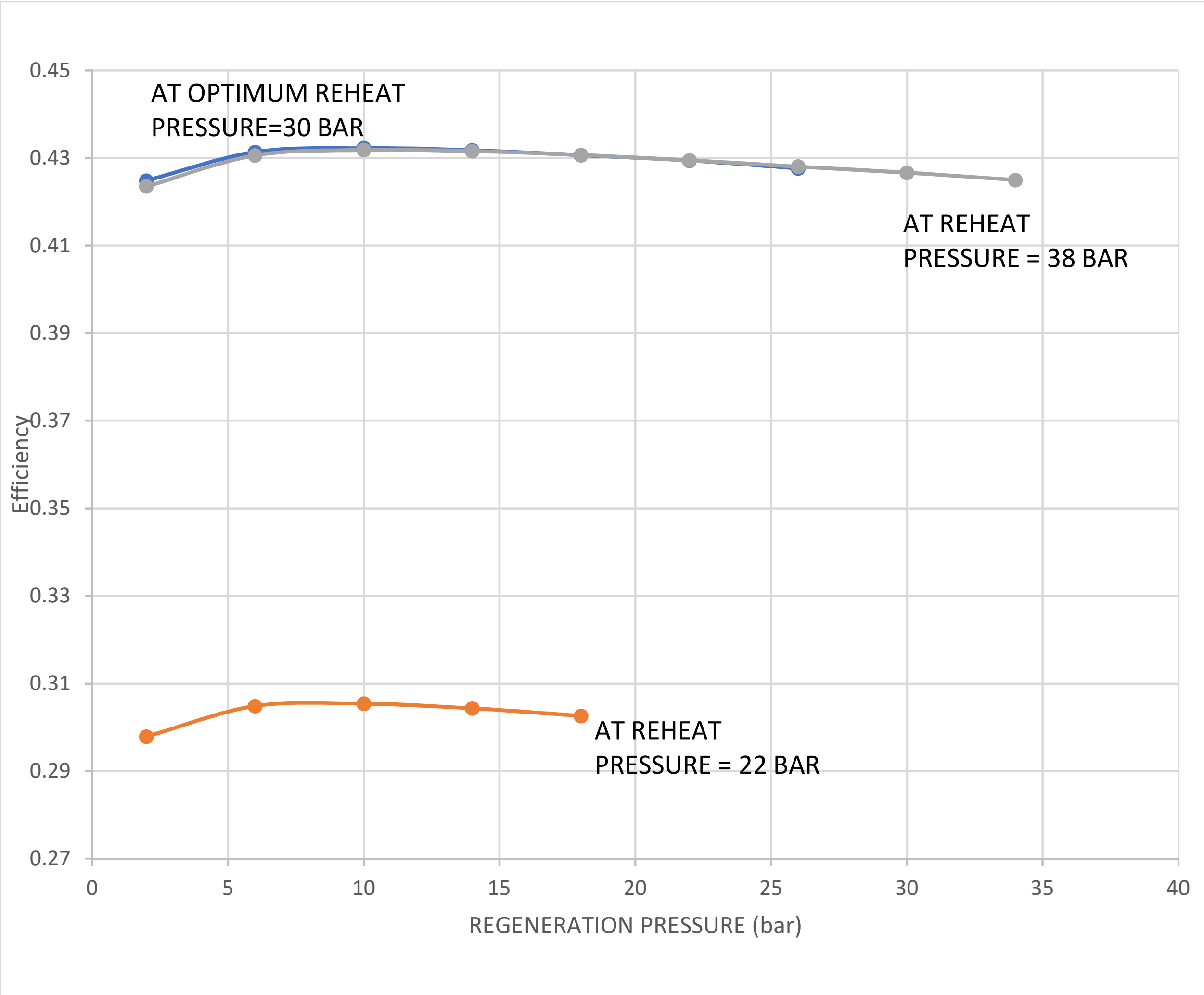


Figure 4-10 Efficiency v/s Regeneration pressure at different reheat pressures

- As we increase reheat pressure around optimum, the optimum regeneration pressure increased.

- Maximum efficiency of the cycle is 0.432264 which is obtained at reheat pressure 30 bar and regeneration pressure 10 bar.
- If optimum reheat pressure is reduced by 8 bar, the maximum efficiency is obtained at a regeneration pressure of 10 bar and efficiency is 0.431164.
- If optimum reheat pressure is increased by 8 bar, the maximum efficiency is obtained at a regeneration pressure of 10 bar and efficiency is 0.431827.

4.11 Analysis and optimization of Rankine Cycle with single reheat and two open type feed water heater arrangements

The variation of the efficiency with respect to the 2nd regeneration pressure while reheat pressure is kept at optimum value and 1st regeneration pressure is kept optimum, optimum + step size and optimum – step size is shown in the graph below.

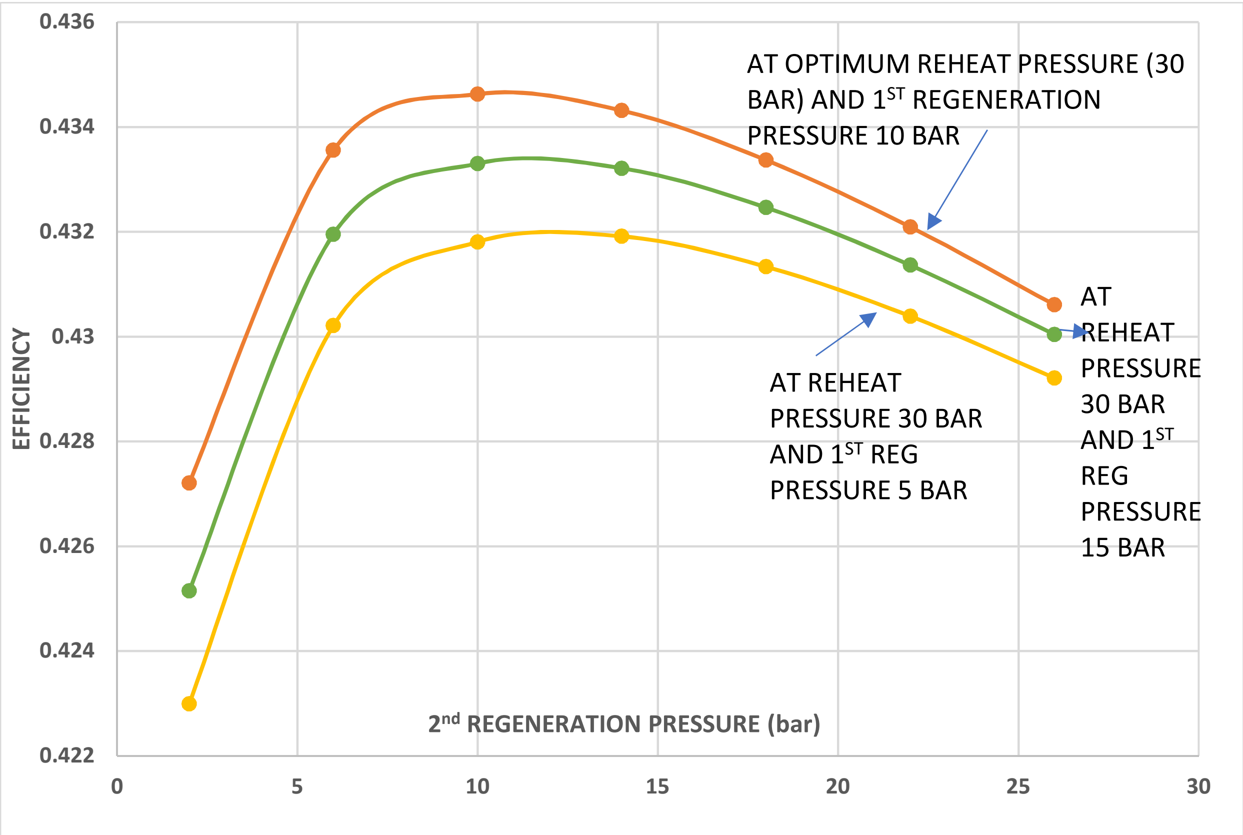


Figure 4-11 Efficiency v/s 2nd Regeneration pressure at different reheat pressures

- Maximum efficiency of the cycle is 0.434626 which is obtained at reheat pressure 30 bar and regeneration pressure 10 bar.
- If 1st regeneration pressure is reduced by 5 bar, the maximum efficiency is obtained at a regeneration pressure of 10 bar and efficiency is 0.431864.
- If 1st regeneration pressure is increased by 5 bar, the maximum efficiency is obtained at a regeneration pressure of 10 bar and efficiency is 0.433303.
- The peaks of all the curves is obtained at a second regeneration pressure of 10 bar.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 List of results of the analysis performed:

- For simple Rankine cycle the maximum efficiency obtained is 0.3794 and dryness fraction at given condition is 0.08387.
- The efficiency of the cycle and dryness fraction increases with the increase in boiler exit temperature when boiler pressure is kept constant. Hence, high degree of superheat is desirable for both efficiency and dryness fraction.
- The efficiency of the cycle increases with the increase in boiler pressure from 100 bar to 200 bar and reaches a maximum of 0.388775 at 200 bar. The dryness fraction decreases when the boiler pressure is increased at a constant boiler exit temperature. Hence for high boiler pressure we require reheating to increase the dryness fraction.
- For a single reheat Rankine cycle, where the reheat pressure is varied from 15 bar to 135 bar with a step size of 15, maximum efficiency obtained is 0.398695 at reheat pressure 30 bar and dryness fraction of steam at inlet to condenser is 0.9700.
- For single regeneration Rankine cycle (open type feedwater heater), where regeneration pressure is varied from 10 bar to 140 bar with a step size of 10, maximum efficiency obtained is 0.408861 at regeneration pressure 20 bar and dryness fraction of steam at the inlet of condenser is 0.8387.
- For single regeneration Rankine cycle (closed type feedwater heater), where regeneration pressure is varied from 4 bar to 140 bar with a step size of 8, maximum efficiency obtained is 0.3974 at regeneration pressure 6 bar and dryness fraction of steam at the inlet of condenser is 0.8387.
- For Rankine cycle with two open type feedwater heaters where 2nd regeneration pressure is varied from 2 bar to 40 bar with a step size of 2, maximum efficiency obtained is 0.4201 at 1st regeneration pressure 38 bar and 2nd regeneration pressure 6 bar.
- For Rankine cycle with single reheat and open type feedwater heater, where regeneration pressure is varied keeping reheat pressure constant at optimum,

optimum – step size & optimum + step size values, maximum efficiency obtained is 0.4322 at reheat pressure 30 bar and regeneration pressure 10 bar

- For Rankine cycle with single reheat and two open type feedwater heaters, where 1st regeneration pressure is varied with a step size of 8, 2nd regeneration pressure is varied with step size 4 bar and reheat pressure is varied with a step size of 6 bar, maximum efficiency of 0.4346 is obtained at reheat pressure of 30 bar, 1st regeneration pressure of 36 bar and 2nd regeneration pressure of 10 bar.

5.2 Comparison of outputs of efficiency for the given parameters on various modifications of Rankine cycle:

Inputs:
Boiler pressure: 150 bar
Condenser pressure: 0.5 bar
Temperature at exit of boiler: 560^{o}C

TYPE OF CYCLE	EFFICIENCY OBTAINED
Simple Rankine cycle	37.94 %
Rankine cycle with 1 reheat	39.86 %
Rankine cycle with 1 regeneration (open type)	40.88 %
Rankine cycle with 1 regeneration (closed type)	39.74 %
Rankine cycle with 1 reheat and 1 regeneration (open type)	42.01 %
Rankine cycle with two regenerations (Both open type)	43.22 %
Rankine cycle with 1 reheat and 2 regenerations (Both open type)	43.46 %

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

The analysis shown clearly shows how with further optimizations, the efficiency of Rankine cycle can be increased.

1. For the given temperature and pressure, the ideal Rankine cycle gives only 39.5 % efficiency.
2. On introducing a single reheater in the cycle, a maximum efficiency of 39.86 % can be obtained if reheat pressure is kept at 30 bar.
3. On introducing a single feed water heater in the simple Rankine cycle, a maximum efficiency of 40.88 % can be obtained if regeneration pressure is kept at 10 bar.
4. On introducing a single reheater in combination with a single open type feed water heater, a maximum efficiency of 43.22 % at reheat pressure 30 bar and regenerative pressure 10 bar.
5. On introducing a single reheater in combination with two open type feed water heaters, a maximum efficiency of 43.46 % can be obtained at reheat pressure 30 bar, first regenerative pressure 36 bar and second regenerative pressure 10 bar.

6.2 Future Scope

- The current program can be extended and generalised for more number of reheaters and feed water heaters.
- Extension of the current program for thermodynamic analysis of Combined cycle power plant.

REFERENCES

[1] **Man Wang , Jiangfeng Wang, Yuzhu Zhao Pan Zhao Yiping Dai** (2012) Thermodynamic analysis and optimization of a solar-driven regenerative organic Rankine cycle (ORC) based on flat-plate solar collectors.

[2] **Meherwan P. Boyce** (2012) Theoretical and Actual Cycle Analyses, in **Gas Turbine Engineering Handbook (Fourth Edition).**

[3] **Bashar Dan-Asabe , Nasser Sayed Nasser** (2014) Rankine Cycle Optimisation at Constant Boiler Temperature with Fuel Economy.

[4] **Yasin Ust , Guven Gonca** (2011) Determination of optimum reheat pressures for single and double reheat irreversible Rankine cycle.

[5] **Dev Kumar Patel** (2015) Improve Steam Turbine efficiency by use of Reheat Rankine cycle.

APPENDIX

C code for analysing Rankine/Modified Rankine cycle:

```
#include <stdio.h>
#include "koolplot.h"
#include <math.h>

int j=0,j2=0,flag=0,flag1=0,status=0;
float lg=0;
float gl=0;
float a[187][14]=
{
{0,0.6119,0.000995,205.94,205.93,0.9007,2500.02,2499.12,0.9001,2374.02,2373.12,0.0013,9.1582,9.1595},
{2,0.7066,0.000995,179.63,179.63,9.2488,2504.40,2495.15,9.2481,2377.48,2368.24,0.0297,9.1052,9.0755},
{4,0.8140,0.000996,157.04,157.04,17.5909,2508.60,2491.01,17.5901,2380.76,2363.17,0.0604,9.0531,8.9928},
{6,0.9357,0.000996,137.59,137.59,25.9279,2512.64,2486.72,25.9269,2383.90,2357.97,0.0908,9.0020,8.9113},
{8,1.0732,0.000997,120.82,120.82,34.2606,2516.58,2482.31,34.2595,2386.91,2352.66,0.1209,8.9519,8.8309},
{10,1.2282,0.000997,106.31,106.31,42.5897,2520.42,2477.83,42.5885,2389.84,2347.25,0.1508,8.9026,8.7518}
{12,1.4026,0.000998,93.74,93.74,50.9160,2524.19,2473.27,50.9146,2392.70,2341.79,0.1804,8.8542,8.6738},
{14,1.5985,0.000999,82.83,82.83,59.2401,2527.90,2468.66,59.2385,2395.51,2336.27,0.2098,8.8066,8.5969},
{16,1.8180,0.000999,73.33,73.33,67.5625,2531.58,2464.02,67.5607,2398.27,2330.71,0.2389,8.7599,8.5211},
{18,2.0635,0.001000,65.04,65.04,75.8837,2535.23,2459.34,75.8817,2401.01,2325.13,0.2678,8.7141,8.4463},
{20,2.3376,0.001000,57.80,57.80,84.2043,2538.85,2454.65,84.2020,2403.73,2319.53,0.2964,8.6690,8.3725},
{22,2.6431,0.001001,51.47,51.46,92.5247,2542.46,2449.94,92.5220,2406.43,2313.91,0.3249,8.6247,8.2998},
{24,2.9830,0.001002,45.90,45.90,100.845,2546.06,2445.21,100.842,2409.12,2308.28,0.3531,8.5811,8.2280},
{26,3.3604,0.001002,41.02,41.02,109.166,2549.65,2440.48,109.163,2411.81,2302.65,0.3811,8.5384,8.1572},
{28,3.7789,0.001003,36.72,36.71,117.488,2553.23,2435.74,117.484,2414.49,2297.01,0.4090,8.4963,8.0874},
{30,4.2420,0.001004,32.92,32.92,125.811,2556.81,2431.00,125.807,2417.17,2291.36,0.4366,8.4550,8.0184},
{32,4.7536,0.001005,29.57,29.56,134.136,2560.39,2426.25,134.131,2419.84,2285.71,0.4640,8.4143,7.9503},
{34,5.3181,0.001005,26.60,26.60,142.462,2563.96,2421.50,142.456,2422.52,2280.06,0.4913,8.3744,7.8831},
{36,5.9398,0.001006,23.96,23.96,150.790,2567.53,2416.74,150.784,2425.19,2274.40,0.5183,8.3351,7.8168},
{38,6.6235,0.001007,21.62,21.62,159.120,2571.09,2411.97,159.113,2427.86,2268.74,0.5452,8.2964,7.7512},
{40,7.3743,0.001008,19.54,19.54,167.452,2574.65,2407.20,167.444,2430.52,2263.08,0.5719,8.2584,7.6865},
{42,8.1975,0.001009,17.69,17.69,175.786,2578.20,2402.41,175.778,2433.18,2257.41,0.5985,8.2210,7.6226},
{44,9.0987,0.001009,16.04,16.03,184.123,2581.75,2397.63,184.114,2435.84,2251.73,0.6248,8.1843,7.5594},
{46,10.084,0.001010,14.56,14.56,192.463,2585.29,2392.83,192.452,2438.50,2246.04,0.6510,8.1481,7.4970},
{48,11.160,0.001011,13.23,13.23,200.805,2588.82,2388.02,200.794,2441.14,2240.35,0.6771,8.1125,7.4354},
{50,12.333,0.001012,12.05,12.04,209.150,2592.34,2383.19,209.137,2443.78,2234.65,0.7030,8.0775,7.3745},
{52,13.610,0.001013,10.98,10.98,217.498,2595.86,2378.36,217.484,2446.42,2228.93,0.7287,8.0430,7.3143},
{54,14.999,0.001014,10.02,10.02,225.848,2599.36,2373.51,225.833,2449.04,2223.21,0.7543,8.0091,7.2547},
{56,16.507,0.001015,9.159,9.158,234.202,2602.85,2368.65,234.185,2451.66,2217.48,0.7798,7.9757,7.1959},
{58,18.143,0.001016,8.381,8.380,242.558,2606.34,2363.78,242.540,2454.27,2211.73,0.8051,7.9428,7.1377},
{60,19.916,0.001017,7.679,7.678,250.918,2609.80,2358.89,250.898,2456.87,2205.97,0.8302,7.9104,7.0802},
{62,21.834,0.001018,7.044,7.043,259.281,2613.26,2353.98,259.259,2459.46,2200.20,0.8552,7.8786,7.0234},
{64,23.906,0.001019,6.470,6.469,267.647,2616.70,2349.05,267.623,2462.04,2194.41,0.8801,7.8472,6.9671},
{66,26.144,0.001020,5.949,5.948,276.016,2620.13,2344.11,275.990,2464.61,2188.62,0.9048,7.8163,6.9115},
{68,28.557,0.001021,5.476,5.475,284.389,2623.54,2339.15,284.360,2467.16,2182.80,0.9294,7.7859,6.8564},
{70,31.156,0.001023,5.047,5.046,292.765,2626.94,2334.18,292.733,2469.71,2176.97,0.9539,7.7559,6.8020},
{72,33.952,0.001024,4.656,4.655,301.144,2630.32,2329.18,301.109,2472.24,2171.13,0.9782,7.7263,6.7481},
```

{74,36.957,0.001025,4.300,4.299,309.527,2633.69,2324.16,309.489,2474.76,2165.27,1.0024,7.6972,6.6948},
{76,40.184,0.001026,3.976,3.975,317.913,2637.04,2319.13,317.872,2477.27,2159.40,1.0265,7.6686,6.6421},
{78,43.645,0.001028,3.680,3.679,326.303,2640.37,2314.07,326.258,2479.76,2153.51,1.0505,7.6403,6.5899},
{80,47.353,0.001029,3.409,3.408,334.696,2643.69,2308.99,334.648,2482.25,2147.60,1.0743,7.6125,6.5382},
{82,51.322,0.001030,3.162,3.161,343.093,2646.99,2303.90,343.040,2484.72,2141.68,1.0980,7.5850,6.4870},
{84,55.567,0.001032,2.935,2.934,351.494,2650.27,2298.78,351.437,2487.17,2135.74,1.1216,7.5579,6.4364},
{86,60.102,0.001033,2.727,2.726,359.899,2653.53,2293.64,359.837,2489.62,2129.78,1.1450,7.5313,6.3862},
{88,64.942,0.001034,2.537,2.536,368.308,2656.78,2288.47,368.240,2492.04,2123.80,1.1684,7.5050,6.3366},
{90,70.104,0.001036,2.361,2.360,376.720,2660.01,2283.29,376.648,2494.46,2117.81,1.1916,7.4790,6.2874},
{92,75.603,0.001037,2.200,2.199,385.137,2663.21,2278.08,385.059,2496.86,2111.80,1.2147,7.4534,6.2387},
{94,81.457,0.001039,2.052,2.051,393.558,2666.40,2272.84,393.474,2499.25,2105.77,1.2377,7.4282,6.1905},
{96,87.683,0.001040,1.915,1.914,401.984,2669.57,2267.58,401.893,2501.62,2099.73,1.2606,7.4033,6.1427},
{98,94.299,0.001042,1.789,1.788,410.414,2672.72,2262.30,410.316,2503.98,2093.66,1.2833,7.3787,6.0954},
{100,101.325,0.001043,1.673,1.672,418.849,2675.84,2256.99,418.743,2506.32,2087.57,1.3060,7.3545,6.0485}
{102,108.778,0.001045,1.566,1.565,427.289,2678.95,2251.66,427.175,2508.64,2081.47,1.3285,7.3306,6.0020}
{104,116.678,0.001046,1.466,1.465,435.733,2682.03,2246.30,435.611,2510.95,2075.34,1.3510,7.3070,5.9560}
{106,125.047,0.001048,1.374,1.373,444.183,2685.09,2240.91,444.052,2513.25,2069.19,1.3733,7.2837,5.9103}
{108,133.905,0.001050,1.289,1.288,452.638,2688.13,2235.49,452.498,2515.52,2063.03,1.3955,7.2606,5.8651}
{110,143.273,0.001051,1.210,1.209,461.099,2691.14,2230.04,460.948,2517.78,2056.83,1.4177,7.2379,5.8203}
{112,153.173,0.001053,1.137,1.136,469.565,2694.13,2224.57,469.404,2520.03,2050.62,1.4397,7.2155,5.7758}
{114,163.628,0.001055,1.069,1.068,478.038,2697.10,2219.06,477.865,2522.25,2044.38,1.4616,7.1933,5.7318}
{116,174.662,0.001057,1.005,1.004,486.516,2700.04,2213.52,486.332,2524.46,2038.12,1.4834,7.1715,5.6880}
{118,186.297,0.001058,0.946389,0.945331,495.001,2702.95,2207.95,494.804,2526.64,2031.84,1.5051,7.1498,
5.6447},
{120,198.559,0.001060,0.891572,0.890512,503.493,2705.84,2202.35,503.282,2528.81,2025.53,1.5267,7.1285,
5.6017},
{122,211.472,0.001062,0.840500,0.839438,511.991,2708.70,2196.71,511.766,2530.96,2019.19,1.5483,7.1074,
5.5591},
{124,225.062,0.001064,0.792881,0.791817,520.496,2711.53,2191.04,520.257,2533.09,2012.83,1.5697,7.0865,
5.5168},
{126,239.354,0.001066,0.748448,0.747382,529.009,2714.34,2185.33,528.754,2535.19,2006.44,1.5910,7.0659,
5.4749},
{128,254.377,0.001068,0.706958,0.705890,537.530,2717.11,2179.58,537.258,2537.28,2000.02,1.6123,7.0455,
5.4332},
{130,270.156,0.001070,0.668188,0.667118,546.058,2719.86,2173.80,545.769,2539.34,1993.57,1.6334,7.0254,
5.3919},
{132,286.720,0.001072,0.631933,0.630861,554.595,2722.57,2167.98,554.287,2541.38,1987.09,1.6545,7.0054,
5.3510},
{134,304.097,0.001074,0.598007,0.596933,563.140,2725.25,2162.11,562.813,2543.40,1980.59,1.6754,6.9857,
5.3103},
{136,322.317,0.001076,0.566238,0.565162,571.693,2727.90,2156.21,571.347,2545.39,1974.05,1.6963,6.9662,
5.2699},
{138,341.408,0.001078,0.536469,0.535391,580.256,2730.52,2150.26,579.888,2547.36,1967.47,1.7171,6.9469,
5.2298},
{140,361.402,0.001080,0.508556,0.507476,588.828,2733.10,2144.27,588.438,2549.31,1960.87,1.7378,6.9279,
5.1900},
{142,382.328,0.001082,0.482365,0.481283,597.410,2735.65,2138.24,596.996,2551.23,1954.23,1.7585,6.9090,
5.1505},
{144,404.219,0.001084,0.457774,0.456690,606.002,2738.16,2132.16,605.564,2553.12,1947.55,1.7790,6.8903,
5.1113},
{146,427.106,0.001086,0.434672,0.433585,614.604,2740.64,2126.03,614.140,2554.98,1940.84,1.7995,6.8718,
5.0723},
{148,451.022,0.001089,0.412954,0.411865,623.217,2743.07,2119.86,622.726,2556.82,1934.10,1.8199,6.8535,
5.0336},
{150,476.000,0.001091,0.392524,0.391433,631.841,2745.47,2113.63,631.322,2558.63,1927.31,1.8402,6.8353,
4.9952},

{152,502.073,0.001093,0.373295,0.372202,640.477,2747.84,2107.36,639.928,2560.41,1920.49,1.8604,6.8174,4.9570},
{154,529.277,0.001095,0.355186,0.354090,649.124,2750.16,2101.04,648.544,2562.17,1913.62,1.8806,6.7996,4.9190},
{156,557.644,0.001098,0.338120,0.337023,657.783,2752.44,2094.66,657.170,2563.89,1906.72,1.9006,6.7819,4.8813},
{158,587.212,0.001100,0.322029,0.320930,666.454,2754.68,2088.23,665.808,2565.58,1899.77,1.9206,6.7645,4.8438},
{160,618.016,0.001102,0.306849,0.305747,675.138,2756.88,2081.74,674.457,2567.24,1892.79,1.9406,6.7472,4.8066},
{162,650.092,0.001105,0.292519,0.291414,683.836,2759.04,2075.20,683.117,2568.87,1885.75,1.9604,6.7300,4.7696},
{164,683.477,0.001107,0.278985,0.277878,692.546,2761.15,2068.60,691.790,2570.47,1878.68,1.9802,6.7130,4.7328},
{166,718.210,0.001110,0.266195,0.265085,701.271,2763.22,2061.95,700.474,2572.03,1871.56,2.0000,6.6961,4.6962},
{168,754.328,0.001112,0.254102,0.252990,710.010,2765.24,2055.23,709.171,2573.56,1864.39,2.0196,6.6794,4.6598},
{170,791.870,0.001115,0.242662,0.241547,718.764,2767.22,2048.45,717.881,2575.06,1857.18,2.0392,6.6628,4.6236},
{172,830.875,0.001117,0.231834,0.230717,727.532,2769.15,2041.62,726.604,2576.52,1849.92,2.0588,6.6464,4.5876},
{174,871.384,0.001120,0.221580,0.220461,736.316,2771.03,2034.71,735.340,2577.95,1842.61,2.0782,6.6301,4.5518},
{176,913.436,0.001122,0.211865,0.210743,745.116,2772.87,2027.75,744.091,2579.34,1835.25,2.0976,6.6139,4.5162},
{178,957.072,0.001125,0.202656,0.201531,753.931,2774.65,2020.72,752.855,2580.69,1827.84,2.1170,6.5978,4.4808},
{180,1002.34,0.001128,0.193922,0.192794,762.764,2776.39,2013.62,761.634,2582.01,1820.38,2.1363,6.5818,4.4456},
{182,1049.27,0.001130,0.185635,0.184504,771.613,2778.07,2006.46,770.427,2583.29,1812.87,2.1555,6.5660,4.4105},
{184,1097.91,0.001133,0.177767,0.176634,780.480,2779.71,1999.23,779.236,2584.54,1805.30,2.1747,6.5502,4.3756},
{186,1148.30,0.001136,0.170295,0.169159,789.364,2781.29,1991.93,788.060,2585.74,1797.68,2.1938,6.5346,4.3408},
{188,1200.50,0.001139,0.163195,0.162057,798.266,2782.82,1984.56,796.899,2586.91,1790.01,2.2129,6.5191,4.3062},
{190,1254.53,0.001141,0.156446,0.155304,807.187,2784.30,1977.11,805.755,2588.03,1782.28,2.2319,6.5037,4.2718},
{192,1310.45,0.001144,0.150027,0.148882,816.127,2785.72,1969.60,814.627,2589.12,1774.49,2.2508,6.4883,4.2375},
{194,1368.30,0.001147,0.143919,0.142772,825.086,2787.09,1962.01,823.516,2590.17,1766.65,2.2697,6.4731,4.2033},
{196,1428.14,0.001150,0.138105,0.136955,834.064,2788.41,1954.34,832.422,2591.18,1758.75,2.2886,6.4579,4.1693},
{198,1489.99,0.001153,0.132568,0.131415,843.063,2789.67,1946.61,841.345,2592.14,1750.80,2.3074,6.4428,4.1354},
{200,1553.92,0.001156,0.127293,0.126137,852.082,2790.87,1938.79,850.286,2593.07,1742.78,2.3262,6.4279,4.1017},
{202,1619.96,0.001159,0.122266,0.121106,861.123,2792.02,1930.89,859.244,2593.95,1734.71,2.3449,6.4129,4.0681},
{204,1688.17,0.001163,0.117472,0.116309,870.184,2793.11,1922.92,868.222,2594.79,1726.57,2.3635,6.3981,4.0346},
{206,1758.60,0.001166,0.112899,0.111733,879.268,2794.14,1914.87,877.217,2595.59,1718.38,2.3822,6.3833,4.0012},

{208,1831.29,0.001169,0.108535,0.107366,888.373,2795.11,1906.74,886.232,2596.35,1710.12,2.4007,6.3686,3.9679},
{210,1906.30,0.001172,0.104369,0.103196,897.501,2796.02,1898.52,895.267,2597.06,1701.80,2.4193,6.3540,3.9347},
{212,1983.67,0.001176,0.100390,0.099214,906.653,2796.88,1890.22,904.321,2597.74,1693.41,2.4377,6.3394,3.9017},
{214,2063.46,0.001179,0.096588,0.095409,915.828,2797.67,1881.84,913.395,2598.36,1684.97,2.4562,6.3249,3.8687},
{216,2145.71,0.001182,0.092955,0.091773,925.026,2798.40,1873.38,922.489,2598.95,1676.46,2.4746,6.3104,3.8358},
{218,2230.49,0.001186,0.089481,0.088295,934.250,2799.07,1864.82,931.605,2599.49,1667.88,2.4929,6.2960,3.8031},
{220,2317.83,0.001189,0.086158,0.084969,943.498,2799.68,1856.19,940.741,2599.98,1659.24,2.5113,6.2817,3.7704},
{222,2407.80,0.001193,0.082979,0.081785,952.772,2800.23,1847.46,949.899,2600.44,1650.54,2.5295,6.2673,3.7378},
{224,2500.45,0.001197,0.079935,0.078738,962.071,2800.72,1838.65,959.079,2600.84,1641.77,2.5478,6.2530,3.7053},
{226,2595.84,0.001200,0.077021,0.075820,971.397,2801.14,1829.74,968.281,2601.21,1632.93,2.5660,6.2388,3.6728},
{228,2694.01,0.001204,0.074229,0.073025,980.750,2801.50,1820.75,977.506,2601.53,1624.02,2.5841,6.2246,3.6404},
{230,2795.02,0.001208,0.071555,0.070346,990.131,2801.80,1811.67,986.754,2601.80,1615.05,2.6023,6.2104,3.6081},
{232,2898.94,0.001212,0.068991,0.067779,999.539,2802.03,1802.49,996.025,2602.03,1606.00,2.6204,6.1963,3.5759},
{234,3005.81,0.001216,0.066532,0.065316,1008.98,2802.19,1793.22,1005.32,2602.21,1596.89,2.6384,6.1821,3.5437},
{236,3115.69,0.001220,0.064174,0.062954,1018.44,2802.30,1783.85,1014.64,2602.35,1587.71,2.6564,6.1680,3.5116},
{238,3228.65,0.001224,0.061911,0.060687,1027.94,2802.33,1774.39,1023.99,2602.44,1578.46,2.6744,6.1539,3.4795},
{240,3344.74,0.001228,0.059739,0.058511,1037.46,2802.30,1764.84,1033.36,2602.49,1569.13,2.6923,6.1398,3.4475},
{242,3464.02,0.001233,0.057654,0.056421,1047.02,2802.21,1755.19,1042.75,2602.49,1559.74,2.7102,6.1258,3.4155},
{244,3586.55,0.001237,0.055651,0.054414,1056.61,2802.05,1745.43,1052.17,2602.45,1550.28,2.7281,6.1117,3.3836},
{246,3712.39,0.001242,0.053727,0.052486,1066.23,2801.82,1735.58,1061.62,2602.36,1540.74,2.7460,6.0977,3.3517},
{48,3841.61,0.001246,0.051878,0.050632,1075.89,2801.52,1725.63,1071.10,2602.23,1531.13,2.7638,6.0836,3.3199},
{250,3974.26,0.001251,0.050100,0.048850,1085.58,2801.16,1715.58,1080.60,2602.05,1521.44,2.7815,6.0696,3.2880},
{252,4110.40,0.001256,0.048391,0.047135,1095.30,2800.73,1705.43,1090.14,2601.82,1511.68,2.7993,6.0555,3.2562},
{254,4250.11,0.001260,0.046747,0.045486,1105.06,2800.23,1695.17,1099.70,2601.55,1501.85,2.8170,6.0414,3.2245},
{256,4393.44,0.001265,0.045164,0.043899,1114.85,2799.66,1684.81,1109.29,2601.23,1491.94,2.8346,6.0273,3.1927},
{258,4540.47,0.001270,0.043641,0.042371,1124.69,2799.02,1674.34,1118.92,2600.87,1481.95,2.8523,6.0132,3.1610},
{260,4691.25,0.001275,0.042175,0.040900,1134.56,2798.32,1663.76,1128.57,2600.46,1471.89,2.8699,5.9991,3.1293},
{262,4845.85,0.001281,0.040763,0.039483,1144.47,2797.54,1653.08,1138.26,2600.01,1461.75,2.8874,5.9850,3.0976},

{264,5004.33,0.001286,0.039403,0.038117,1154.42,2796.70,1642.28,1147.98,2599.51,1451.53,2.9050,5.9708,
3.0659},
{266,5166.78,0.001291,0.038093,0.036801,1164.41,2795.79,1631.38,1157.74,2598.97,1441.23,2.9225,5.9566,
3.0342},
{268,5333.25,0.001297,0.036829,0.035532,1174.44,2794.80,1620.36,1167.53,2598.38,1430.86,2.9399,5.9424,
3.0025},
{270,5503.82,0.001303,0.035612,0.034309,1184.52,2793.75,1609.23,1177.35,2597.75,1420.40,2.9574,5.9282,
2.9708},
{272,5678.56,0.001309,0.034437,0.033129,1194.64,2792.62,1597.98,1187.21,2597.07,1409.86,2.9748,5.9139,
2.9391},
{274,5857.53,0.001315,0.033304,0.031990,1204.81,2791.43,1586.62,1197.11,2596.35,1399.23,2.9921,5.8995,
2.9074},
{276,6040.80,0.001321,0.032211,0.030891,1215.03,2790.16,1575.13,1207.05,2595.58,1388.53,3.0094,5.8852,
2.8757},
{280,6420.58,0.001333,0.030138,0.028805,1235.61,2787.41,1551.80,1227.05,2593.91,1366.86,3.0440,5.8562,
2.8123},
{282,6617.23,0.001340,0.029155,0.027816,1245.98,2785.93,1539.95,1237.12,2593.01,1355.89,3.0612,5.8417,
2.7805},
{284,6818.48,0.001346,0.028206,0.026859,1256.40,2784.38,1527.98,1247.22,2592.06,1344.84,3.0784,5.8271,
2.7488},
{286,7024.42,0.001353,0.027288,0.025935,1266.88,2782.75,1515.87,1257.38,2591.07,1333.69,3.0955,5.8125,
2.7170},
{288,7235.11,0.001360,0.026402,0.025042,1277.42,2781.06,1503.64,1267.58,2590.03,1322.46,3.1126,5.7977,
2.6852},
{290,7450.65,0.001367,0.025545,0.024178,1288.01,2779.29,1491.27,1277.83,2588.96,1311.13,3.1296,5.7830,
2.6533},
{292,7671.10,0.001374,0.024717,0.023343,1298.67,2777.44,1478.77,1288.13,2587.83,1299.71,3.1466,5.7681,
2.6215},
{294,7896.54,0.001382,0.023917,0.022535,1309.39,2775.53,1466.14,1298.48,2586.67,1288.19,3.1635,5.7532,
2.5896},
{296,8127.07,0.001389,0.023142,0.021753,1320.18,2773.54,1453.36,1308.89,2585.46,1276.57,3.1804,5.7382,
2.5578},
{298,8362.76,0.001397,0.022393,0.020996,1331.03,2771.47,1440.44,1319.35,2584.20,1264.85,3.1972,5.7231,
2.5259},
{300,8603.69,0.001405,0.021669,0.020263,1341.96,2769.34,1427.38,1329.87,2582.91,1253.04,3.2139,5.7079,
2.4940},
{302,8849.96,0.001413,0.020967,0.019554,1352.96,2767.13,1414.16,1340.45,2581.57,1241.11,3.2306,5.6927,
2.4621},
{304,9101.63,0.001421,0.020288,0.018867,1364.04,2764.84,1400.80,1351.10,2580.18,1229.08,3.2472,5.6773,
2.4302},
{306,9358.81,0.001430,0.019631,0.018201,1375.19,2762.48,1387.29,1361.81,2578.76,1216.95,3.2636,5.6619,
2.3982},
{308,9621.58,0.001439,0.018995,0.017556,1386.43,2760.05,1373.62,1372.59,2577.29,1204.70,3.2800,5.6463,
2.3664},
{310,9890.03,0.001448,0.018379,0.016931,1397.76,2757.54,1359.78,1383.44,2575.77,1192.34,3.2962,5.6307,
2.3345},
{312,10164.24,0.001457,0.017782,0.016325,1409.17,2754.96,1345.79,1394.36,2574.22,1179.86,3.3123,5.6150
,2.3027},
{314,10444.32,0.001466,0.017203,0.015737,1420.68,2752.30,1331.62,1405.36,2572.62,1167.26,3.3282,5.5991
,2.2709},
{316,10730.34,0.001476,0.016643,0.015167,1432.28,2749.56,1317.28,1416.44,2570.98,1154.54,3.3439,5.5832
,2.2393},
{318,11022.41,0.001486,0.016100,0.014614,1443.98,2746.75,1302.77,1427.61,2569.30,1141.69,3.3594,5.5671
,2.2077},
{320,11320.63,0.001496,0.015573,0.014078,1455.79,2743.87,1288.08,1438.86,2567.57,1128.71,3.3746,5.5510
,2.1764},

```
{322,11625.08,0.001506,0.015063,0.013557,1467.71,2740.91,1273.20,1450.20,2565.80,1115.60,3.3895,5.5347
,2.1452},
{324,11935.86,0.001517,0.014568,0.013052,1479.74,2737.87,1258.13,1461.64,2563.99,1102.35,3.4041,5.5183
,2.1142},
{326,12253.07,0.001527,0.014088,0.012561,1491.88,2734.76,1242.87,1473.17,2562.13,1088.96,3.4182,5.5018
,2.0836},
{328,12576.82,0.001538,0.013623,0.012084,1504.15,2731.57,1227.41,1484.81,2560.24,1075.43,3.4318,5.4851
,2.0533},
{330,12907.21,0.001550,0.013171,0.011622,1516.55,2728.30,1211.75,1496.55,2558.29,1061.74,3.4448,5.4684
,2.0235},
{332,13244.33,0.001561,0.012733,0.011172,1529.09,2724.96,1195.87,1508.41,2556.31,1047.90,3.4571,5.4515
,1.9944},
{334,13588.29,0.001573,0.012308,0.010735,1541.76,2721.54,1179.78,1520.38,2554.28,1033.90,3.4685,5.4344
,1.9659},
{336,13939.20,0.001585,0.011896,0.010311,1554.57,2718.04,1163.46,1532.48,2552.21,1019.74,3.4788,5.4173
,1.9384},
{338,14297.16,0.001598,0.011496,0.009898,1567.54,2714.46,1146.92,1544.70,2550.10,1005.40,3.4879,5.4000
,1.9121},
{340,14662.29,0.001611,0.011108,0.009497,1580.67,2710.81,1130.14,1557.05,2547.94,990.89,3.4953,5.3825,
1.8872},
{342,15034.68,0.001624,0.010731,0.009107,1593.96,2707.08,1113.11,1569.55,2545.74,976.19,3.5009,5.3649,
1.8641},
{344,15414.47,0.001637,0.010365,0.008728,1607.42,2703.26,1095.84,1582.19,2543.50,961.31,3.5040,5.3472,
1.8433},
{346,15801.74,0.001651,0.010009,0.008359,1621.07,2699.38,1078.31,1594.98,2541.21,946.23,3.5040,5.3294,
1.8254},
{348,16196.63,0.001665,0.009664,0.008000,1634.90,2695.41,1060.51,1607.93,2538.88,930.94,3.5002,5.3114,
1.8111},
{350,16599.25,0.001679,0.009330,0.007650,1648.92,2691.36,1042.44,1621.05,2536.50,915.45,3.4915,5.2932,
1.8017},
{352,17009.71,0.001694,0.009004,0.007311,1663.15,2687.24,1024.08,1634.34,2534.07,899.73,3.4764,5.2749,
1.7985},
{354,17428.13,0.001709,0.008689,0.006980,1677.60,2683.03,1005.43,1647.81,2531.60,883.79,3.4528,5.2565,
1.8036},
{356,17854.64,0.001724,0.008382,0.006658,1692.26,2678.75,986.48,1661.47,2529.09,867.61,3.4179,5.2378,1.
8199},
{358,18289.36,0.001740,0.008084,0.006344,1707.16,2674.38,967.22,1675.33,2526.52,851.19,3.3676,5.2191,1.
8515},
{360,18732.41,0.001756,0.007795,0.006039,1722.30,2669.94,947.64,1689.40,2523.91,834.51,3.2954,5.2002,1.
9047},
{362,19183.91,0.001773,0.007514,0.005741,1737.69,2665.41,927.72,1703.68,2521.26,817.57,3.1918,5.1811,1.
9893},
{364,19644.00,0.001790,0.007242,0.005452,1753.35,2660.81,907.45,1718.19,2518.55,800.36,3.0409,5.1619,2.
1210},
{366,20112.81,0.001807,0.006977,0.005170,1769.28,2656.12,886.84,1732.93,2515.79,782.86,2.8150,5.1425,2.
3276},
{368,20590.46,0.001825,0.006720,0.004895,1785.50,2651.35,865.85,1747.93,2512.99,765.06,2.4610,5.1230,2.
6620},
{370,21077.08,0.001843,0.006470,0.004627,1802.03,2646.50,844.48,1763.17,2510.13,746.95,1.8582,5.1033,3.
2451},
{372,21572.82,0.001862,0.006228,0.004366,1818.86,2641.57,822.71,1778.69,2507.22,728.53,0.5970,5.0835,4.
4864},
{374,22077.81,0.001881,0.005993,0.004111,1836.02,2636.56,800.54,1794.49,2504.26,709.77,-
19.7443,5.0635,24.8077}
};
float b[124][14]=
```

{
{1,6.99,0.000997,128.97,128.97,29.40,2514.61,2485.21,29.40,2385.64,2356.24,0.1058,8.9770,8.8713},
{2,17.50,0.001000,67.00,67.00,73.42,2534.32,2460.90,73.42,2400.31,2326.89,0.2606,8.7254,8.4648},
{3,24.09,0.001002,45.66,45.66,100.97,2546.23,2445.26,100.97,2409.25,2308.28,0.3544,8.5791,8.2247},
{4,28.98,0.001003,34.79,34.79,121.38,2554.99,2433.61,121.38,2415.82,2294.44,0.4225,8.4759,8.0534},
{5,32.90,0.001005,28.19,28.18,137.74,2561.99,2424.25,137.73,2421.06,2283.33,0.4763,8.3963,7.9200},
{6,36.19,0.001006,23.73,23.73,151.46,2567.86,2416.40,151.45,2425.46,2274.00,0.5208,8.3314,7.8106},
{7,39.03,0.001007,20.52,20.52,163.32,2572.92,2409.60,163.32,2429.25,2265.93,0.5590,8.2768,7.7178},
{8,41.54,0.001008,18.10,18.10,173.80,2577.38,2403.58,173.79,2432.59,2258.79,0.5924,8.2296,7.6372},
{9,43.79,0.001009,16.20,16.20,183.21,2581.38,2398.17,183.20,2435.58,2252.38,0.6221,8.1881,7.5660},
{10,45.84,0.001010,14.67,14.67,191.75,2585.00,2393.25,191.74,2438.29,2246.55,0.6489,8.1510,7.5021},
{15,54.00,0.001014,10.02,10.02,225.85,2599.36,2373.51,225.84,2449.04,2223.20,0.7543,8.0091,7.2547},
{20,60.09,0.001017,7.65,7.65,251.31,2609.96,2358.65,251.29,2456.97,2205.68,0.8313,7.9090,7.0777},
{25,64.99,0.001020,6.20,6.20,271.83,2618.41,2346.58,271.80,2463.30,2191.49,0.8924,7.8318,6.9394},
{30,69.13,0.001022,5.23,5.23,289.13,2625.46,2336.33,289.10,2468.58,2179.48,0.9432,7.7689,6.8257},
{35,72.71,0.001024,4.53,4.52,304.15,2631.52,2327.37,304.11,2473.12,2169.01,0.9869,7.7159,6.7291},
{40,75.89,0.001026,3.99,3.99,317.46,2636.85,2319.39,317.42,2477.11,2159.69,1.0252,7.6702,6.6450},
{45,78.74,0.001028,3.58,3.58,329.45,2641.61,2312.16,329.40,2480.68,2151.27,1.0594,7.6299,6.5705},
{50,81.35,0.001030,3.24,3.24,340.37,2645.91,2305.54,340.32,2483.90,2143.58,1.0903,7.5939,6.5036},
{55,83.74,0.001031,2.96,2.96,350.42,2649.84,2299.43,350.36,2486.84,2136.48,1.1185,7.5615,6.4429},
{60,85.96,0.001033,2.73,2.73,359.73,2653.46,2293.73,359.67,2489.55,2129.89,1.1446,7.5319,6.3873},
{65,88.02,0.001034,2.53,2.53,368.42,2656.82,2288.40,368.35,2492.07,2123.72,1.1687,7.5047,6.3360},
{70,89.96,0.001036,2.36,2.36,376.57,2659.94,2283.38,376.49,2494.41,2117.92,1.1912,7.4795,6.2883},
{75,91.79,0.001037,2.22,2.22,384.25,2662.87,2278.62,384.17,2496.61,2112.43,1.2123,7.4561,6.2438},
{80,93.51,0.001038,2.09,2.09,391.52,2665.63,2274.11,391.44,2498.67,2107.23,1.2322,7.4343,6.2021},
{85,95.15,0.001040,1.97,1.97,398.43,2668.23,2269.81,398.34,2500.62,2102.28,1.2510,7.4138,6.1628},
{90,96.72,0.001041,1.87,1.87,405.00,2670.70,2265.69,404.91,2502.47,2097.56,1.2688,7.3945,6.1256},
{95,98.21,0.001042,1.78,1.78,411.29,2673.04,2261.75,411.19,2504.23,2093.04,1.2858,7.3762,6.0904},
{100,99.63,0.001043,1.69,1.69,417.31,2675.27,2257.96,417.20,2505.90,2088.70,1.3020,7.3589,6.0569},
{125,105.99,0.001048,1.37,1.37,444.14,2685.08,2240.93,444.01,2513.25,2069.24,1.3734,7.2838,5.9104},
{150,111.37,0.001053,1.16,1.16,466.91,2693.20,2226.29,466.75,2519.34,2052.59,1.4330,7.2225,5.7895},
{175,116.06,0.001057,1.00,1.00,486.78,2700.13,2213.35,486.59,2524.54,2037.95,1.4843,7.1708,5.6865},
{200,120.23,0.001061,0.8855,0.8844,504.47,2706.17,2201.70,504.26,2529.07,2024.82,1.5295,7.1260,5.5966},
{225,123.99,0.001064,0.7930,0.7920,520.46,2711.52,2191.06,520.22,2533.09,2012.86,1.5699,7.0866,5.5167},
{250,127.43,0.001067,0.7185,0.7175,535.09,2716.32,2181.23,534.83,2536.69,2001.86,1.6065,7.0513,5.4448},
{275,130.60,0.001070,0.6571,0.6561,548.60,2720.67,2172.07,548.30,2539.95,1991.65,1.6400,7.0194,5.3794},
{300,133.54,0.001073,0.6057,0.6046,561.16,2724.63,2163.47,560.84,2542.93,1982.09,1.6710,6.9903,5.3193},
{325,136.29,0.001076,0.5619,0.5608,572.92,2728.28,2155.36,572.57,2545.67,1973.10,1.6997,6.9635,5.2637},
{350,138.87,0.001079,0.5241,0.5230,583.98,2731.64,2147.66,583.61,2548.20,1964.59,1.7266,6.9386,5.2120},
{375,141.31,0.001081,0.4913,0.4902,594.44,2734.77,2140.33,594.04,2550.55,1956.51,1.7518,6.9155,5.1637},
{400,143.62,0.001084,0.4624,0.4613,604.37,2737.68,2133.31,603.93,2552.74,1948.80,1.7756,6.8938,5.1182},
{425,145.81,0.001086,0.4368,0.4357,613.82,2740.41,2126.59,613.36,2554.79,1941.43,1.7982,6.8735,5.0753},
{450,147.91,0.001088,0.4139,0.4128,622.85,2742.97,2120.12,622.36,2556.71,1934.36,1.8196,6.8543,5.0347},
{475,149.91,0.001091,0.3934,0.3923,631.49,2745.37,2113.88,630.97,2558.52,1927.55,1.8400,6.8361,4.9961},
{500,151.84,0.001093,0.3748,0.3737,639.79,2747.64,2107.86,639.24,2560.23,1920.99,1.8595,6.8188,4.9594},
{525,153.68,0.001095,0.3580,0.3569,647.77,2749.79,2102.02,647.19,2561.85,1914.65,1.8781,6.8024,4.9243},
{550,155.46,0.001097,0.3426,0.3415,655.46,2751.83,2096.37,654.86,2563.38,1908.52,1.8960,6.7867,4.8907},
{575,157.17,0.001099,0.3286,0.3275,662.89,2753.76,2090.87,662.26,2564.83,1902.58,1.9132,6.7717,4.8585},
{600,158.83,0.001101,0.3156,0.3145,670.07,2755.60,2085.52,669.41,2566.22,1896.80,1.9298,6.7573,4.8275},
{625,160.43,0.001103,0.3037,0.3026,677.03,2757.35,2080.32,676.34,2567.54,1891.19,1.9458,6.7435,4.7977},
{650,161.98,0.001105,0.2927,0.2915,683.78,2759.02,2075.24,683.06,2568.79,1885.73,1.9612,6.7302,4.7690},
{675,163.49,0.001106,0.2824,0.2813,690.33,2760.61,2070.28,689.58,2569.99,1880.41,1.9762,6.7174,4.7412},
{700,164.95,0.001108,0.2728,0.2717,696.70,2762.13,2065.43,695.92,2571.14,1875.22,1.9906,6.7050,4.7144},
{725,166.37,0.001110,0.2639,0.2628,702.90,2763.59,2060.69,702.09,2572.24,1870.15,2.0047,6.6931,4.6884},
{750,167.75,0.001112,0.2556,0.2545,708.93,2764.99,2056.05,708.10,2573.29,1865.19,2.0183,6.6815,4.6632},
{775,169.09,0.001113,0.2478,0.2467,714.82,2766.33,2051.51,713.96,2574.30,1860.35,2.0315,6.6703,4.6388},

{800,170.40,0.001115,0.2404,0.2393,720.57,2767.61,2047.05,719.67,2575.27,1855.60,2.0444,6.6595,4.6151},
{825,171.68,0.001117,0.2335,0.2324,726.18,2768.85,2042.67,725.25,2576.20,1850.95,2.0570,6.6490,4.5920},
{850,172.93,0.001118,0.2270,0.2259,731.66,2770.03,2038.37,730.71,2577.10,1846.39,2.0692,6.6388,4.5696},
{875,174.15,0.001120,0.2208,0.2197,737.02,2771.17,2034.15,736.04,2577.96,1841.92,2.0811,6.6288,4.5477},
{900,175.35,0.001121,0.2150,0.2139,742.27,2772.27,2030.00,741.26,2578.79,1837.53,2.0927,6.6191,4.5264},
{925,176.51,0.001123,0.2094,0.2083,747.41,2773.33,2025.92,746.37,2579.59,1833.22,2.1041,6.6097,4.5056},
{950,177.66,0.001124,0.2042,0.2031,752.45,2774.35,2021.90,751.38,2580.36,1828.98,2.1152,6.6005,4.4853},
{975,178.78,0.001126,0.1992,0.1981,757.39,2775.33,2017.95,756.29,2581.10,1824.82,2.1261,6.5916,4.4655},
{1000,179.87,0.001127,0.1945,0.1933,762.23,2776.28,2014.05,761.10,2581.82,1820.72,2.1367,6.5828,4.4462}
{1250,189.80,0.001141,0.1571,0.1560,806.32,2784.16,1977.84,804.89,2587.79,1782.90,2.2321,6.5052,4.2730}
{1500,198.28,0.001154,0.1318,0.1307,844.32,2789.84,1945.52,842.59,2592.12,1749.53,2.3127,6.4407,4.1280}
{1750,205.72,0.001165,0.1135,0.1124,877.95,2793.99,1916.04,875.91,2595.31,1719.40,2.3828,6.3854,4.0026}
{2000,212.37,0.001176,0.099676,0.098500,908.29,2797.03,1888.74,905.94,2597.67,1691.74,2.4450,6.3367,3.8
917},
{2250,218.40,0.001187,0.088799,0.087612,936.04,2799.20,1863.16,933.37,2599.41,1666.04,2.5012,6.2931,3.7
920},
{2500,223.94,0.001197,0.080025,0.078828,961.70,2800.70,1839.01,958.71,2600.64,1641.94,2.5524,6.2535,3.7
010},
{2750,229.06,0.001206,0.072792,0.071586,985.63,2801.67,1816.03,982.31,2601.49,1619.17,2.5997,6.2170,3.6
173},
{3000,233.84,0.001216,0.066724,0.065508,1008.11,2802.18,1794.07,1004.47,2602.01,1597.55,2.6437,6.1832,
3.5396},
{3250,238.32,0.001225,0.061557,0.060332,1029.35,2802.33,1772.98,1025.37,2602.27,1576.90,2.6848,6.1517,
3.4669},
{3500,242.54,0.001234,0.057103,0.055869,1049.52,2802.17,1752.65,1045.20,2602.31,1557.11,2.7234,6.1220,
3.3985},
{3750,246.54,0.001243,0.053222,0.051979,1068.75,2801.74,1732.99,1064.09,2602.16,1538.07,2.7600,6.0939,
3.3339},
{4000,250.34,0.001252,0.049808,0.048557,1087.16,2801.09,1713.93,1082.15,2601.86,1519.71,2.7947,6.0672,
3.2725},
{4250,253.96,0.001260,0.046782,0.045522,1104.83,2800.24,1695.41,1099.48,2601.41,1501.94,2.8277,6.0417,
3.2140},
{4500,257.42,0.001269,0.044080,0.042811,1121.85,2799.22,1677.37,1116.14,2600.86,1484.71,2.8593,6.0174,
3.1581},
{4750,260.73,0.001277,0.041652,0.040375,1138.28,2798.04,1659.76,1132.21,2600.19,1467.98,2.8895,5.9940,
3.1044},
{5000,263.92,0.001286,0.039458,0.038173,1154.18,2796.74,1642.55,1147.75,2599.44,1451.69,2.9186,5.9714,
3.0528},
{5250,266.98,0.001294,0.037466,0.036171,1169.60,2795.31,1625.71,1162.81,2598.62,1435.81,2.9466,5.9497,
3.0031},
{5500,269.94,0.001303,0.035648,0.034345,1184.58,2793.78,1609.20,1177.41,2597.72,1420.30,2.9736,5.9286,
2.9550},
{5750,272.80,0.001311,0.033982,0.032671,1199.16,2792.16,1593.00,1191.62,2596.76,1405.14,2.9998,5.9082,
2.9084},
{6000,275.56,0.001319,0.032449,0.031130,1213.37,2790.45,1577.08,1205.45,2595.75,1390.30,3.0251,5.8883,
2.8633},
{6250,278.23,0.001328,0.031035,0.029708,1227.24,2788.66,1561.42,1218.94,2594.69,1375.75,3.0496,5.8690,
2.8194},
{6500,280.83,0.001336,0.029726,0.028390,1240.80,2786.81,1546.01,1232.11,2593.59,1361.47,3.0735,5.8502,
2.7767},
{6750,283.35,0.001344,0.028510,0.027166,1254.07,2784.89,1530.82,1245.00,2592.45,1347.45,3.0968,5.8319,
2.7351},
{7000,285.80,0.001352,0.027377,0.026025,1267.07,2782.92,1515.84,1257.61,2591.28,1333.67,3.1194,5.8139,
2.6945},
{7250,288.19,0.001361,0.026320,0.024960,1279.83,2780.89,1501.06,1269.97,2590.07,1320.10,3.1415,5.7964,
2.6548},

{7500,290.51,0.001369,0.025331,0.023962,1292.36,2778.82,1486.46,1282.09,2588.84,1306.75,3.1631,5.7792,
2.6160},
{7750,292.78,0.001377,0.024403,0.023026,1304.67,2776.71,1472.03,1294.00,2587.58,1293.58,3.1843,5.7623,
2.5780},
{8000,294.99,0.001385,0.023531,0.022145,1316.79,2774.55,1457.76,1305.70,2586.30,1280.60,3.2050,5.7458,
2.5408},
{8250,297.15,0.001394,0.022710,0.021316,1328.72,2772.36,1443.64,1317.22,2585.01,1267.79,3.2253,5.7295,
2.5043},
{8500,299.26,0.001402,0.021935,0.020533,1340.47,2770.14,1429.66,1328.56,2583.69,1255.13,3.2452,5.7136,
2.4684},
{8750,301.32,0.001410,0.021203,0.019793,1352.07,2767.89,1415.82,1339.73,2582.36,1242.63,3.2647,5.6979,
2.4331},
{9000,303.34,0.001419,0.020511,0.019092,1363.51,2765.61,1402.09,1350.75,2581.01,1230.26,3.2840,5.6824,
2.3984},
{9250,305.31,0.001427,0.019854,0.018427,1374.82,2763.30,1388.48,1361.62,2579.65,1218.03,3.3029,5.6672,
2.3643},
{9500,307.25,0.001435,0.019231,0.017795,1385.99,2760.97,1374.98,1372.35,2578.28,1205.92,3.3215,5.6522,
2.3306},
{9750,309.15,0.001444,0.018638,0.017194,1397.04,2758.62,1361.58,1382.96,2576.89,1193.93,3.3399,5.6374,
2.2975},
{10000,311.01,0.001452,0.018074,0.016622,1407.97,2756.24,1348.28,1393.44,2575.50,1182.05,3.3580,5.6228
,2.2647},
{10500,314.63,0.001469,0.017025,0.015555,1429.51,2751.44,1321.93,1414.09,2572.68,1158.60,3.3936,5.5941
,2.2005},
{11000,318.12,0.001486,0.016067,0.014581,1450.68,2746.58,1295.90,1434.33,2569.84,1135.51,3.4283,5.5662
,2.1378},
{11500,321.50,0.001503,0.015190,0.013687,1471.51,2741.66,1270.15,1454.23,2566.97,1112.75,3.4624,5.5388
,2.0764},
{12000,324.76,0.001521,0.014384,0.012864,1492.06,2736.70,1244.64,1473.81,2564.09,1090.28,3.4958,5.5120
,2.0162},
{12500,327.92,0.001538,0.013641,0.012103,1512.35,2731.69,1219.34,1493.13,2561.18,1068.06,3.5287,5.4858
,1.9571},
{13000,330.99,0.001555,0.012953,0.011398,1532.43,2726.66,1194.23,1512.21,2558.27,1046.06,3.5611,5.4600
,1.8989},
{13500,333.97,0.001573,0.012315,0.010742,1552.32,2721.59,1169.27,1531.08,2555.34,1024.26,3.5932,5.4347
,1.8415},
{14000,336.87,0.001591,0.011721,0.010131,1572.04,2716.50,1144.45,1549.77,2552.40,1002.62,3.6250,5.4098
,1.7848},
{14500,339.69,0.001609,0.011168,0.009559,1591.64,2711.39,1119.75,1568.32,2549.45,981.14,3.6564,5.3853,
1.7288},
{15000,342.43,0.001627,0.010650,0.009024,1611.12,2706.26,1095.13,1586.72,2546.50,959.78,3.6877,5.3611,
1.6734},
{15500,345.11,0.001645,0.010166,0.008521,1630.52,2701.11,1070.60,1605.02,2543.54,938.52,3.7189,5.3373,
1.6184},
{16000,347.73,0.001663,0.009711,0.008048,1649.84,2695.96,1046.12,1623.23,2540.58,917.35,3.7499,5.3138,
1.5639},
{16500,350.28,0.001681,0.009284,0.007602,1669.11,2690.79,1021.68,1641.37,2537.61,896.24,3.7808,5.2907,
1.5098},
{17000,352.77,0.001700,0.008881,0.007181,1688.34,2685.62,997.28,1659.45,2534.64,875.19,3.8118,5.2678,1.
4560},
{17500,355.21,0.001718,0.008501,0.006783,1707.55,2680.44,972.89,1677.48,2531.66,854.18,3.8427,5.2452,1.
4025},
{18000,357.60,0.001737,0.008143,0.006406,1726.76,2675.26,948.50,1695.49,2528.68,833.19,3.8737,5.2228,1.
3492},
{18500,359.94,0.001756,0.007804,0.006048,1745.97,2670.07,924.10,1713.48,2525.70,812.22,3.9047,5.2007,1.
2960},

```
{19000,362.23,0.001775,0.007482,0.005708,1765.20,2664.88,899.69,1731.47,2522.72,791.24,3.9359,5.1789,1.2430},
{19500,364.48,0.001794,0.007178,0.005384,1784.46,2659.70,875.24,1749.47,2519.73,770.25,3.9671,5.1573,1.1902},
{20000,366.68,0.001813,0.006889,0.005076,1803.76,2654.51,850.75,1767.49,2516.73,749.24,3.9986,5.1359,1.1373},
{20500,368.84,0.001833,0.006614,0.004781,1823.11,2649.32,826.21,1785.54,2513.74,728.19,4.0302,5.1147,1.0845},
{21000,370.96,0.001852,0.006353,0.004501,1842.53,2644.14,801.61,1803.63,2510.73,707.10,4.0620,5.0938,1.0318},
{21500,373.05,0.001872,0.006104,0.004232,1862.01,2638.96,776.95,1821.77,2507.73,685.96,4.0941,5.0730,0.9790},
{22000,375.09,0.001892,0.005867,0.003975,1881.57,2633.79,752.21,1839.96,2504.72,664.76,4.1264,5.0525,0.9261}
};
float c[80][13]=
{
{0.02,17.5,86.08,97.63,109.2,120.7,132.2,143.8,155.3,178.4,201.5,224.6,247.6},
{0.04,29.0,43.03,48.81,54.58,60.35,66.12,71.89,77.66,89.20,100.7,112.3,123.8},
{0.06,36.2,28.68,3253,36.38,40.23,44.08,47.93,51.77,59.47,67.16,74.85,82.54},
{0.08,41.5,21.50,24.40,27.28,30.17,33.06,35.94,38.83,44.60,50.37,56.14,61.91},
{0.10,45.8,17.20,19.51,21.83,24.14,26.45,28.75,31.06,35.68,40.30,44.91,49.53},
{0.15,54.0,11.51,13.06,14.61,16.16,17.71,19.25,20.80,23.89,26.98,30.07,33.16},
{0.20,60.1,8.585,9.748,10.91,12.07,13.22,14.37,15.53,17.84,20.15,22.45,24.76},
{0.25,65.0,6.874,7.808,8.737,9.665,10.59,11.52,12.44,14.29,16.14,17.99,19.84},
{0.30,69.1,5.714,6.493,7.268,8.040,8.811,9.581,10.35,11.89,13.43,14.70,16.51},
{0.35,72.7,4.898,5.568,6.233,6.896,7.557,8.218,8.879,10.20,11.52,12.84,14.16},
{0.40,75.9,4.279,4.866,5.448,6.028,6.607,7.185,7.763,8.918,10.07,11.23,12.38},
{0.45,78.7,3.803,4.325,4.844,5.360,5.875,6.389,6.903,7.930,8.957,9.984,10.99},
{0.50,81.3,3.418,3.889,4.356,4.821,5.284,5.747,6.209,7.134,8.057,8.981,9.904},
{0.60,86.0,2.844,3.238,3.628,4.016,4.402,4.788,5.174,5.944,6.714,7.484,8.254},
{0.70,90.0,2.434,2.773,3.108,3.441,3.772,4.103,4.434,5.095,5.755,6.415,7.074},
{0.80,93.5,2.126,2.425,2.718,3.010,3.300,3.590,3.879,4.457,5.035,5.613,6.190},
{0.90,96.7,1.887,2.153,2.415,2.674,2.933,3.190,3.448,3.962,4.475,4.989,5.502},
{1.00,99.6,1.696,1.936,2.172,2.406,2.639,2.871,3.103,3.565,4.028,4.490,4.952},
{1.50,111.4,0,1.285,1.444,1.601,1.757,1.912,2.067,2376,2.685,2.993,3.301},
{2.00,120.2,0,0.9595,1.080,1.199,1.316,1.433,1.549,1.781,2.013,2.244,2.475},
{2.5,127.4,0,0.7641,0.8620,0.9574,1.052,1.145,1.239,1.424,1.610,1.795,1.980},
{3.0,133.5,0,0.6337,0.7164,0.7964,0.8753,0.9535,1.031,1.187,1.341,1.496,1.650},
{3.5,138.9,0,0.5406,0.6123,0.6814,0.7493,0.8166,0.8835,1.017,1.149,1.282,1.414},
{4.0,143.6,0,0.4707,0.5343,0.5952,0.6549,0.7139,0.7725,0.8892,1.005,1.121,1.237},
{4.5,147.9,0,0.4165,0.4733,0.5234,0.5817,0.6343,0.6865,0.7905,0.8939,0.9971,1.100},
{5.0,151.8,0,0.4250,0.4744,0.5226,0.5701,0.6172,0.7108,0.8040,0.8969,0.9896},
{6.0,158.8,0,0,0.3520,0.3939,0.4344,0.4742,0.5136,0.5918,0.6696,0.7471,0.8245},
{7.0,165.0,0,0,0.2999,0.3364,0.3714,0.4057,0.4396,0.5069,0.5737,0.6402,0.7066},
{8.0,170.4,0,0,0.2608,0.2932,0.3241,0.3543,0.3842,0.4432,0.5017,0.5600,0.6181},
{9.0,175.4,0,0,0.2303,0.2596,0.2874,0.3144,0.3410,0.3936,0.4458,0.4976,0.5493},
{10.0,179.9,0,0,0.2059,0.2328,0.2580,0.2824,0.3065,0.3540,0.4010,0.4477,0.4943},
{11.0,184.1,0,0,0.1859,0.2108,0.2339,0.2563,0.2782,0.3215,0.3644,0.4069,0.4492},
{12.0,188.0,0,0,0.1692,0.1924,0.2139,0.2345,0.2547,0.2945,0.3338,0.3729,0.4118},
{13.0,191.6,0,0,0.1551,0.1769,0.1969,0.2161,0.2348,0.2716,0.3080,0.3441,0.3800},
{14.0,195.0,0,0,0.1429,0.1636,0.1823,0.2002,0.2177,0.2520,0.2859,0.3194,0.3528},
{15.0,198.3,0,0,0.1324,0.1520,0.1697,0.1865,0.2029,0.2350,0.2667,0.2980,0.3292},
{16.0,201.4,0,0,0,0.1419,0.1587,0.1745,0.1900,0.2202,0.2499,0.2793,0.3086},
{17.0,204.3,0,0,0,0.1329,0.1489,0.1640,0.1786,0.2070,0.2351,0.2628,0.2904},
{18.0,207.1,0,0,0,0.1250,0.1402,0.1546,0.1684,0.1954,0.2219,0.2481,0.2742},
```

```
{19.0,209.8,0,0,0,0.1179,0.1325,0.1461,0.1593,0.1849,0.2101,0.2350,0.2597},
{20.0,212.4,0,0,0,0.1115,0.1255,0.1386,0.1511,0.1756,0.1995,0.2232,0.2467},
{22.0,217.2,0,0,0,0.1004,0.1134,0.1255,0.1370,0.1593,0.1812,0.2028,0.2242},
{24.0,221.8,0,0,0,0.09108,0.1034,0.1146,0.1252,0.1458,0.1659,0.1858,0.2054},
{26.0,226.0,0,0,0,0.08321,0.09483,0.1053,0.1153,0.1344,0.1530,0.1714,0.1895},
{28.0,230.0,0,0,0,0.07644,0.08751,0.09740,0.1067,0.1246,0.1419,0.1590,0.1759},
{30.0,233.8,0,0,0,0.07055,0.08116,0.09053,0.09931,0.1161,0.1323,0.1483,0.1641},
{32.0,237.4,0,0,0,0.06538,0.07559,0.08451,0.09283,0.1087,0.1239,0.1390,0.1538},
{34.0,240.9,0,0,0,0.06080,0.07068,0.07920,0.08711,0.1021,0.1165,0.1307,0.1447},
{36.0,244.2,0,0,0,0.05670,0.06630,0.07448,0.08202,0.09626,0.1100,0.1234,0.1366},
{38.0,247.3,0,0,0,0.05302,0.06237,0.07025,0.07747,0.09104,0.1041,0.1168,0.1294},
{40.0,250.3,0,0,0,0,0.05883,0.06645,0.07338,0.08634,0.09876,0.1109,0.1229},
{40.0,250.3,0,0,0,0,0.05883,0.06645,0.07338,0.08634,0.09876,0.11090,0.1229},
{42.0,253.2,0,0,0,0,0.05563,0.06300,0.06967,0.08209,0.09397,0.10560,0.1170},
{44.0,256.0,0,0,0,0,0.05270,0.05986,0.06630,0.07823,0.08961,0.10070,0.1116},
{46.0,258.8,0,0,0,0,0.05003,0.05699,0.06322,0.07470,0.08562,0.09626,0.1067},
{48.0,261.4,0,0,0,0,0.04757,0.05436,0.06039,0.07147,0.08197,0.09219,0.1022},
{50.0,263.9,0,0,0,0,0.04530,0.05194,0.05779,0.06849,0.07862,0.08845,0.09809},
{55.0,269.9,0,0,0,0,0.04343,0.04666,0.05213,0.06202,0.07131,0.08031,0.08912},
{60.0,275.6,0,0,0,0,0.03615,0.04222,0.04738,0.05659,0.06518,0.07348,0.08159},
{65.0,280.8,0,0,0,0,0.03258,0.03848,0.04338,0.05203,0.06003,0.06774,0.07526},
{70.0,285.8,0,0,0,0,0.02946,0.03523,0.03992,0.04809,0.05559,0.06279,0.06980},
{75.0,290.5,0,0,0,0,0.02672,0.03243,0.03694,0.04469,0.05176,0.05852,0.06509},
{80.0,295.0,0,0,0,0,0.02426,0.02995,0.03431,0.04170,0.04839,0.05477,0.06096},
{85.0,299.2,0,0,0,0,0.02191,0.02776,0.03200,0.03908,0.04544,0.05148,0.05732},
{90.0,303.3,0,0,0,0,0,0.02579,0.02993,0.03674,0.04280,0.04853,0.05408},
{95.0,307.2,0,0,0,0,0,0.02403,0.02808,0.03465,0.04045,0.04591,0.05119},
{100.0,311.0,0,0,0,0,0,0.02242,0.02641,0.03276,0.03832,0.04355,0.04858},
{110.0,318.0,0,0,0,0,0,0.01961,0.02351,0.02950,0.03466,0.03947,0.04408},
{120.0,324.6,0,0,0,0,0,0.01721,0.02108,0.02679,0.03160,0.03607,0.04033},
{130.0,330.8,0,0,0,0,0,0.01510,0.01902,0.02449,0.02902,0.03319,0.03716},
{140.0,336.6,0,0,0,0,0,0.01321,0.01723,0.02251,0.02680,0.03072,0.03444},
{150.0,342.1,0,0,0,0,0,0.01146,0.01566,0.02080,0.02488,0.02859,0.03209},
{160.0,347.3,0,0,0,0,0,0.00976,0.01428,0.01929,0.02320,0.02672,0.03003},
{170.0,352.30,0,0,0,0,0,0,0.01303,0.01797,0.02172,0.02507,0.02821},
{180.0,357.00,0,0,0,0,0,0,0.01191,0.01679,0.02040,0.02300,0.02659},
{190.0,361.40,0,0,0,0,0,0,0.01089,0.01573,0.01922,0.02229,0.02515},
{200.0,365.70,0,0,0,0,0,0,0.00995,0.01477,0.01816,0.02111,0.02385},
{210.0,369.80,0,0,0,0,0,0,0.00907,0.01391,0.01720,0.02004,0.02267},
{220.0,373.70,0,0,0,0,0,0,0.00825,0.01312,0.01633,0.01907,0.02160},
{221.2,374.15,0,0,0,0,0,0,0.00816,0.01303,0.01622,0.01895,0.02135},
};
float d[79][13]=
{
{0.02,17.5,2688.5,2783.7,2880.0,2977.7,3076.8,3177.5,3279.7,3489.2,3705.6,3928.8,4158.7},
{0.04,29.0,2688.3,2783.5,2879.9,2977.6,3076.8,3177.4,3279.7,3489.2,3705.6,3928.8,4158.7},
{0.06,36.2,2688.0,2783.4,2879.8,2977.6,3076.7,3177.4,3279.6,3489.2,3705.6,3928.8,4158.7},
{0.08,41.5,2687.8,2783.2,2879.7,2977.5,3076.7,3177.3,3279.6,3489.1,3705.5,3928.8,4158.7},
{0.10,45.8,2687.5,2783.1,2879.6,2977.4,3076.6,3177.3,3279.6,3489.1,3705.5,3928.8,4158.7},
{0.15,54.0,2686.9,2782.4,2879.5,2977.3,3076.5,3177.7,3279.5,3489.1,3705.5,3928.7,4158.7},
{0.20,60.1,2686.3,2782.3,2879.2,2977.1,3076.4,3177.1,3279.4,3489.0,3705.4,3928.7,4158.7},
{0.25,65.0,2685.7,2782.0,2879.0,2977.0,3076.3,3177.0,3279.3,3489.0,3705.4,3928.7,4158.6},
{0.30,69.1,2685.1,2781.6,2878.7,2976.8,3076.1,3176.9,3279.3,3488.9,3705.4,3928.7,4158.6},
{0.35,72.7,2684.5,2781.2,2878.5,2976.7,3076.0,3176.8,3279.2,3488.9,3705.3,3928.7,4158.6},
{0.40,75.9,2683.8,2780.9,2878.2,2976.5,3075.9,3176.8,3279.1,3488.8,3705.3,3928.6,4158.6},
{0.45,78.7,2683.2,2780.5,2878.0,2976.3,3075.8,3176.7,3279.1,3488.8,3705.2,3928.6,4158.5},
```

{0.50,81.3,2682.6,2780.1,2877.7,2976.1,3075.7,3176.6,3279.0,3488.7,3705.2,3928.6,4158.5},
{0.60,86.0,2681.3,2779.4,2877.3,2975.8,3075.4,3176.4,3278.8,3488.6,3705.1,3928.5,4158.5},
{0.70,90.0,2680.0,2778.6,2876.8,2975.5,3075.2,3176.2,3278.7,3488.5,3705.0,3928.4,4158.4},
{0.80,93.5,2678.8,2777.8,2876.3,2975.2,3075.0,3176.0,3278.5,3488.4,3705.0,3928.4,4158.4},
{0.90,96.7,2677.5,2777.1,2875.8,2974.8,3074.7,3175.8,3278.4,3488.3,3704.9,3928.3,4158.3},
{1.00,99.6,2676.2,2776.3,2875.4,2974.5,3074.5,3175.6,3278.2,3488.1,3704.8,3928.2,4158.3},
{1.50,111.4,0,2772.5,2872.9,2972.9,3073.3,3174.7,3277.5,3487.6,3704.4,3927.9,4158.0},
{2.00,120.2,0,2768.5,2870.5,2971.2,3072.1,3173.8,3276.7,3487.0,3704.0,3927.6,4157.8},
{2.50,127.4,0,2764.5,2868.0,2969.6,3070.9,3172.8,3275.9,3486.5,3703.6,3927.3,4157.6},
{3.00,133.5,0,2760.4,2865.5,2967.9,3069.7,3171.9,3275.2,3486.0,3703.2,3927.0,4157.3},
{3.50,138.9,0,2756.3,2863.0,2966.2,3068.4,3170.9,3274.4,3485.4,3702.7,3926.7,4157.1},
{4.00,143.6,0,2752.0,2860.4,2964.5,3067.2,3170.0,3273.6,3484.9,3702.3,3926.4,4156.9},
{4.50,147.9,0,2746.7,2857.8,2962.8,3066.0,3169.1,3272.9,3484.3,3701.9,3926.1,4156.7},
{5.00,151.8,0,0,2855.1,2961.1,3064.8,3168.1,3272.1,3483.8,3701.5,3925.8,4156.4},
{6.00,158.8,0,0,2849.7,2957.6,3062.3,3166.2,3270.6,3482.7,3700.7,3925.1,4155.9},
{7.00,165.0,0,0,2844.2,2954.0,3059.8,3164.3,3269.0,3481.6,3699.9,3924.5,4155.5},
{8.00,170.4,0,0,2838.6,2950.4,3057.3,3162.4,3267.5,3480.5,3699.1,3923.9,4155.0},
{9.00,175.4,0,0,2832.7,2946.8,3054.7,3160.5,3266.0,3479.4,3698.2,3923.3,4154.5},
{10.0,179.9,0,0,2826.8,2943.0,3052.1,3158.5,3264.4,3478.3,3697.4,3922.7,4154.1},
{11.0,184.1,0,0,2820.7,2939.3,3049.6,3156.6,3262.9,3477.2,3696.6,3922.0,4153.6},
{12.0,188.0,0,0,2814.4,2935.4,3046.9,3154.6,3261.3,3476.1,3695.8,3921.4,4153.1},
{13.0,191.6,0,0,2808.0,2931.5,3044.3,3152.7,3259.7,3475.0,3695.0,3920.8,4152.7},
{14.0,195.0,0,0,2801.4,2927.6,3041.6,3150.7,3258.2,3473.9,3694.1,3920.2,4152.2},
{15.0,198.3,0,0,2794.7,2923.5,3038.9,3148.7,3256.6,3472.8,3693.3,3919.6,4151.7},
{16.0,201.4,0,0,0,2919.4,3036.2,3146.7,3255.0,3471.7,3692.5,3918.9,4151.3},
{17.0,204.3,0,0,0,2915.3,3033.5,3144.7,3253.5,3470.6,3691.7,3918.3,4150.8},
{18.0,207.1,0,0,0,2911.0,3030.7,3142.7,3251.9,3469.5,3690.9,3917.7,4150.3},
{19.0,209.3,0,0,0,2906.7,3027.9,3140.7,3250.3,3468.4,3690.0,3917.1,4149.8},
{20.0,212.4,0,0,0,2902.4,3025.0,3138.6,3248.7,3467.3,3689.2,3916.5,4149.4},
{22.0,217.2,0,0,0,2893.4,3019.3,3134.5,3245.5,3465.1,3687.6,3915.2,4148.4},
{24.0,221.8,0,0,0,2884.2,3013.4,3130.4,3242.3,3462.9,3685.9,3914.0,4147.5},
{26.0,226.0,0,0,0,2874.7,3007.4,3126.1,3239.0,3460.6,3684.3,3912.7,4146.6},
{28.0,230.0,0,0,0,2864.9,3001.3,3121.9,3235.8,3458.4,3682.6,3911.5,4145.6},
{30.0,233.8,0,0,0,2854.8,2995.1,3117.5,3232.5,3456.2,3681.0,3910.3,4144.7},
{32.0,237.4,0,0,0,2844.4,2988.7,3113.2,3229.2,3454.0,3679.3,3909.0,4143.8},
{34.0,240.9,0,0,0,2833.6,2982.2,3108.7,3225.9,3451.7,3677.7,3907.8,4142.8},
{36.0,244.2,0,0,0,2822.5,2975.6,3104.2,3222.5,3449.5,3676.1,3906.5,4141.9},
{38.0,247.3,0,0,0,2811.0,2968.9,3099.7,3219.1,3447.2,3674.4,3905.3,4141.0},
{40.0,250.3,0,0,0,0,2962.0,3095.1,3215.7,3445.0,3672.8,3904.1,4140.0},
{42.0,253.2,0,0,0,0,2955.0,3090.4,3212.3,3442.7,3671.1,3902.8,4139.1},
{44.0,256.0,0,0,0,0,2947.8,3085.7,3208.8,3440.5,3669.5,3901.6,4138.2},
{46.0,258.8,0,0,0,0,2940.5,3080.9,3205.3,3438.2,3667.8,3900.3,4137.2},
{48.0,261.4,0,0,0,0,2933.1,3076.1,3201.8,3435.9,3666.2,3899.1,4136.3},
{50.0,263.9,0,0,0,0,2925.5,3071.2,3198.3,3433.7,3664.5,3897.9,4135.3},
{55.0,269.9,0,0,0,0,2905.8,3058.7,3189.3,3427.9,3660.4,3894.8,4133.0},
{60.0,275.6,0,0,0,0,2885.0,3045.8,3180.1,3422.2,3656.2,3891.7,4130.7},
{65.0,280.8,0,0,0,0,2863.0,3032.4,3170.8,3416.4,3652.1,3888.6,4128.8},
{70.0,285.8,0,0,0,0,2839.0,3018.7,3161.2,3410.6,3647.9,3885.4,4126.0},
{75.0,290.5,0,0,0,0,2814.1,3004.5,3151.6,3404.7,3643.7,3882.4,4123.7},
{80.0,295.0,0,0,0,0,2786.6,2989.9,3141.6,3398.8,3639.5,3879.2,4121.3},
{85.0,299.2,0,0,0,0,2757.1,2974.7,3131.5,3392.8,3635.4,3876.1,4119.0},
{90.0,303.3,0,0,0,0,0,2959.0,3121.2,3386.8,3631.1,3873.0,4116.7},
{95.0,307.2,0,0,0,0,0,2942.7,3110.7,3380.7,3627.0,3869.9,4114.4},
{100.0,311.0,0,0,0,0,0,2925.8,3099.9,3374.6,3622.7,3866.8,4112.0},
{110.0,318.0,0,0,0,0,0,2889.6,3077.8,3362.2,3614.2,3860.5,4107.3},
{120.0,324.6,0,0,0,0,0,2849.7,3054.8,3349.6,3605.7,3854.3,4102.7},

```
{130.0,330.8,0,0,0,0,0,2805.0,3030.7,3336.8,3597.1,3848.0,4098.0},
{140.0,336.6,0,0,0,0,0,2754.2,3005.6,3323.8,3588.5,3841.7,4093.3},
{150.0,342.1,0,0,0,0,0,2694.8,2979.1,3310.6,3579.8,3835.4,4088.6},
{160.0,347.3,0,0,0,0,0,2620.8,2951.3,3297.1,3571.0,3829.1,4084.0},
{170.0,352.30,0,0,0,0,0,0,2921.7,3283.5,3562.2,3822.8,4079.3},
{180.0,357.00,0,0,0,0,0,0,2890.3,3269.6,3553.4,3816.5,4074.6},
{190.0,361.40,0,0,0,0,0,0,2856.7,3255.4,3544.5,3810.2,4070.0},
{200.0,365.70,0,0,0,0,0,0,2820.5,3241.1,3535.5,3803.8,4065.3},
{210.0,369.80,0,0,0,0,0,0,2781.3,3226.5,3526.5,3797.5,4060.6},
{220.0,373.70,0,0,0,0,0,0,2738.8,3211.7,3517.4,3791.1,4055.9},
{221.2,374.15,0,0,0,0,0,0,2734.5,3210.7,3516.4,3789.1,4054.7},
};
float e[79][13]=
{
{0.02,17.5,9.193,9.433,9.648,9.844,10.025,10.193,10.351,10.641,10.904,11.146,11.371},
{0.04,29.0,8.873,9.113,9.328,9.524,9.705,9.874,10.031,10.321,10.585,10.827,11.051},
{0.06,36.2,8.685,8.925,9.141,9.337,9.518,9.686,9.844,10.134,10.397,10.639,10.864},
{0.08,41.5,8.552,8.792,9.008,9.204,9.385,9.554,9.711,10.001,10.265,10.507,10.731},
{0.10,45.8,8.449,8.689,8.905,9.101,9.282,9.450,9.608,9.898,10.162,10.404,10.628},
{0.15,54.0,8.261,8.502,8.718,8.915,9.096,9.264,9.422,9.712,9.975,10.217,10.442},
{0.20,60.1,8.126,8.368,8.584,8.781,8.962,9.130,9.288,9.578,9.842,10.084,10.309},
{0.25,65.0,8.022,8.264,8.481,8.678,8.859,9.028,9.186,9.476,9.739,9.981,10.206},
{0.30,69.1,7.936,8.179,8.396,8.593,8.774,8.943,9.101,9.391,9.654,9.897,10.121},
{0.35,72.7,7.864,8.107,8.325,8.522,8.703,8.872,9.030,9.320,9.583,9.826,10.050},
{0.40,75.9,7.801,8.045,8.263,8.460,8.641,8.810,8.968,9.258,9.522,9.764,9.989},
{0.45,78.7,7.745,7.990,8.208,8.405,8.587,8.755,8.914,9.204,9.467,9.709,9.934},
{0.50,81.3,7.695,7.941,8.159,8.356,8.538,8.707,8.865,9.155,9.419,9.661,9.886},
{0.60,86.0,7.609,7.855,8.074,8.272,8.454,8.622,8.781,9.071,9.334,9.576,9.801},
{0.70,90.0,7.535,7.783,8.002,8.200,8.382,8.551,8.709,9.000,9.263,9.505,9.730},
{0.80,93.5,7.470,7.720,7.940,8.138,8.320,8.489,8.648,8.938,9.214,9.444,9.669},
{0.90,96.7,7.413,7.664,7.884,8.083,8.266,8.435,8.593,8.884,9.147,9.389,9.614},
{1.00,99.6,7.362,7.614,7.835,8.034,8.217,8.386,8.544,8.835,9.098,9.341,9.565},
{1.50,111.4,0,7.419,7.644,7.845,8.028,8.198,8.356,8.647,8.911,9.153,9.378},
{2.00,120.2,0,7.279,7.507,7.710,7.894,8.064,8.223,8.514,8.778,9.020,9.245},
{2.5,127.4,0,7.169,7.400,7.604,7.789,7.960,8.119,8.410,8.674,8.917,9.142},
{3.0,133.5,0,7.077,7.312,7.518,7.703,7.874,8.034,8.326,8.590,8.833,9.058},
{3.5,138.5,0,6.998,7.237,7.444,7.631,7.802,7.962,8.254,8.518,8.761,8.986},
{4.0,143.6,0,6.929,7.171,7.380,7.568,7.740,7.899,8.192,8.456,8.699,8.925},
{4.5,147.9,0,6.866,7.112,7.323,7.512,7.684,7.844,8.137,8.402,8.645,8.870},
{5.0,151.8,0,0,7.059,7.272,7.461,7.634,7.795,8.088,8.353,8.596,8.821},
{6.0,158.8,0,0,6.966,7.183,7.374,7.548,7.709,8.003,8.268,8.511,8.737},
{7.0,165.0,0,0,6.886,7.107,7.300,7.475,7.636,7.931,8.196,8.440,8.665},
{8.0,170.4,0,0,6.815,7.040,7.235,7.411,7.573,7.868,8.134,8.377,8.603},
{9.0,175.4,0,0,6.751,6.980,7.177,7.354,7.517,7.812,8.079,8.323,8.549},
{10.0,179.9,0,0,6.692,6.926,7.125,7.303,7.467,7.763,8.029,8.273,8.500},
{11.0,184.1,0,0,6.638,6.876,7.078,7.257,7.421,7.718,7.985,8.229,8.455},
{12.0,188.0,0,0,6.587,6.831,7.034,7.214,7.379,7.677,7.944,8.188,8.415},
{13.0,191.6,0,0,6.539,6.788,6.994,7.175,7.340,7.639,7.906,8.151,8.378},
{14.0,195.0,0,0,6.494,6.748,6.956,7.139,7.305,7.603,7.871,8.116,8.343},
{15.0,198.3,0,0,6.451,6.710,6.921,7.104,7.271,7.570,7.839,8.084,8.311},
{16.0,201.4,0,0,0,6.674,6.887,7.072,7.239,7.540,7.808,8.054,8.281},
{17.0,204.3,0,0,0,6.640,6.856,7.042,7.210,7.511,7.779,8.025,8.252},
{18.0,207.1,0,0,0,6.607,6.826,7.013,7.182,7.483,7.752,7.998,8.226},
{19.0,209.3,0,0,0,6.576,6.797,7.986,7.155,7.457,7.727,7.973,8.200},
{20.0,212.4,0,0,0,6.545,6.770,6.960,7.130,7.432,7.702,7.949,8.176},
{22.0,217.2,0,0,0,6.488,6.718,6.911,7.082,7.386,7.657,7.904,8.132},
```

```
{24.0,221.8,0,0,0,6.434,6.670,6.866,7.038,7.344,7.615,7.862,8.091},
{26.0,226.0,0,0,0,6.382,6.625,6.824,6.998,7.305,7.577,7.825,8.053},
{28.0,230.0,0,0,0,6.333,6.582,6.784,6.960,7.269,7.541,7.789,8.018},
{30.0,233.8,0,0,0,6.286,6.542,6.747,6.925,7.235,7.508,7.756,7.986},
{32.0,237.4,0,0,0,6.240,6.504,6.712,6.891,7.203,7.477,7.726,7.955},
{34.0,240.9,0,0,0,6.195,6.467,6.679,6.860,7.172,7.447,7.697,7.927},
{36.0,244.2,0,0,0,6.151,6.432,6.647,6.829,7.144,7.420,7.669,7.900},
{38.0,247.3,0,0,0,6.109,6.397,6.616,6.801,7.117,7.393,7.643,7.874},
{40.0,250.3,0,0,0,0,6.364,6.587,6.773,7.091,7.368,7.619,7.850},
{42.0,253.2,0,0,0,0,6.332,6.559,6.747,7.066,7.344,7.595,7.826},
{44.0,256.0,0,0,0,0,6.301,6.532,6.722,7.043,7.321,7.573,7.804},
{46.0,258.8,0,0,0,0,6.270,6.505,6.697,7.020,7.299,7.551,7.783},
{48.0,261.4,0,0,0,0,6.240,6.479,6.674,6.998,7.278,7.531,7.763},
{50.0,263.9,0,0,0,0,6.211,6.455,6.651,6.977,7.258,7.511,7.743},
{55.0,269.9,0,0,0,0,6.139,6.395,6.597,6.928,7.210,7.464,7.697},
{60.0,275.6,0,0,0,0,6.069,6.339,6.546,6.882,7.166,7.422,7.655},
{65.0,280.8,0,0,0,0,6.001,6.285,6.499,6.839,7.126,7.382,7.617},
{70.0,285.8,0,0,0,0,5.933,6.233,6.454,6.799,7.088,7.346,7.581},
{75.0,290.5,0,0,0,0,5.864,6.184,6.411,6.762,7.053,7.311,7.547},
{80.0,295.0,0,0,0,0,5.794,6.135,6.369,6.726,7.019,7.279,7.516},
{85.0,299.2,0,0,0,0,5.744,6.088,6.330,6.692,6.987,7.249,7.486},
{90.0,303.3,0,0,0,0,0,6.041,6.292,6.660,6.957,7.220,7.458},
{95.0,307.2,0,0,0,0,0,5.995,6.254,6.629,6.929,7.192,7.431},
{100.0,311.0,0,0,0,0,0,5.949,6.218,6.599,6.901,7.166,7.406},
{110.0,318.0,0,0,0,0,0,5.857,6.148,6.543,6.850,7.117,7.358},
{120.0,324.6,0,0,0,0,0,5.764,6.081,6.491,6.802,7.072,7.315},
{130.0,330.8,0,0,0,0,0,5.666,6.016,6.441,6.758,7.030,7.274},
{140.0,336.6,0,0,0,0,0,5.562,5.951,6.394,6.716,6.991,7.237},
{150.0,342.1,0,0,0,0,0,5.447,5.888,6.349,6.676,6.954,7.201},
{160.0,347.3,0,0,0,0,0,5.311,5.824,6.305,6.639,6.919,7.168},
{170.0,352.3,0,0,0,0,0,0,5.760,6.264,6.603,6.886,7.137},
{180.0,357.0,0,0,0,0,0,0,5.695,6.223,6.569,6.854,7.107},
{190.0,361.4,0,0,0,0,0,0,5.628,6.184,6.536,6.824,7.078},
{200.0,365.7,0,0,0,0,0,0,5.559,6.146,6.504,6.795,7.051},
{210.0,369.8,0,0,0,0,0,0,5.486,6.108,6.474,6.768,7.025},
{200.0,373.7,0,0,0,0,0,0,5.410,6.072,6.444,6.741,7.000},
{221.2,374.15,0,0,0,0,0,0,5.399,6.068,6.441,6.738,6.994},
};
void interpol_superheat(float press,int temp_hp,int j1,float &hh,float &ss,float &tt);
void interpol_table_b(float press,float &hf,float &hg,float &hfg,float &sf,float &sg,float &sfg,float &vf,float
&t_sat);
int main()
{
double p[100];
double eff[100];
int i,ch,ch1,ch2,ch3,ch4,count=0,l=0,m=0;
float
prr,tp,tz,j1,hz,sz,hfz,hgz,hfgz,sfz,sgz,sfgz,vfz,t_satz,temp_hp,c_p,c_p1,c_p2,c_p3,c_p4,max_efficiency=0,pr,p3,
p4,p5,p6=0,x,bp,cp,hp,lp,ip,ip1,ip3,hfip3,hgip3,hfgip3,sfip3,sgip3,sfgip3,vfip3,hfcp,hgcp,sfcp,sgcp,hgip,hfgip,s
fip,vfip,hfip,sgip,sfgip,hfip1,hgip1,hfgip1,sfip1,sgip1,sfgip1,vfip1,ip2,hfip2,hgip2,hfgip2,sfip2,sgip2,sfgip2,vfip
2,x3=1,x5=1,h3,h2=0,x4=1,x6=1,x7=1,h4,h1,h5,h6,h7,h8,h9,h10,h11,h12,work,efficiency,sfgcp,hflp,hfglp,hglp,
sglp,sflp,sfglp,s2=0,s3=0,s4=0,s5=0,s6=0,s7=0;
float vfcp,hfgcp,vflp,efficiency1=0,efficiency2=0;
float t3=0,t4,t5=02,t6,t7=0,t_sat_lp,t_sat_ip,t_sat_ip1,t_sat_ip2,t_sat_ip3,t_sat_cp;
float step,step1,step2,step3,temp_cond;
```

```
printf("For finding out steam properties for a given temperature and pressure, Enter 1. \n \nFor optimization of Reheat and regenerative Rankine cycle, Enter 2 \n");
scanf("%d",&ch4);

if(ch4!=1 && ch4!=2)
printf("Invalid input. Please run program again. \n");

if(ch4==1)
{
printf("Enter pressure \n");
scanf("%f",&prr);
printf("If temperature is known,enter temperature,or else enter 0 \n");
scanf("%f",&tp);

if(tp==0)
{
interpol_table_b(prr,hfz,hgz,hfgz,sfz,sgz,sfgz,vfz,t_satz);
printf("Temperature not entered. Assumed: Steam is not superheated. \n");
printf("Value of hf is %f \n",hfz);
printf("Value of hg is %f \n",hgz);
printf("Value of hfg is %f \n",hfgz);
printf("Value of sf is %f \n",sfz);
printf("Value of sg is %f \n",sgz);
printf("Value of sfg is %f \n",sfgz);
printf("Value of vf is %f \n",vfz);
printf("Value of saturation temperatute is %f \n",t_satz);
}
if(tp!=0)
{
int j1;
switch((int)tp)
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
```

```
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (tp<100)
{
flag=1;
gl=100;
lg=150;
j=2;
j2=3;
break;
}
if(tp>100 && tp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j2=3;break;
}
if(tp>150 && tp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j2=4;break;
}
if(tp>200 && tp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j2=5;break;
}
if(tp>250 && tp<300)
{
gl=250;
lg=300;
```

```
flag=1;
j=5,j2=6;break;
}
if(tp>300 && tp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j2=7; break;
}
if(tp>350 && tp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j2=8;break;
}
if(tp>400 && tp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j2=9;break;
}
if(tp>500 && tp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j2=10;break;
}
if(tp>600 && tp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j2=11;break;
}
if(tp>700 && tp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(tp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(prr,(int)tp,j1,hz,sz,tz);
if(status==1)
```

```
{
printf("The given temperature is below the saturation temperature of steam at given pressure \n Steam is not superheated. \n");

interpol_table_b(prr,hfz,hgz,hfgz,sfz,sgz,sfgz,vfz,t_satz);

printf("Value of hf is %f \n",hfz);
printf("Value of hg is %f \n",hgz);
printf("Value of hfg is %f \n",hfgz);
printf("Value of sf is %f \n",sfz);
printf("Value of sg is %f \n",sgz);
printf("Value of sfg is %f \n",sfgz);
printf("Value of vf is %f \n",vfz);
printf("Value of saturation temperatute is %f \n",t_satz);
}
else
{
printf("Value of h is %f \n",hz);
printf("Value of s is %f \n",sz);
printf("Value of saturation temperatute is %f \n",tz);
}
}
}
if(ch4==2)
{
printf("Enter number of reheats \n");
scanf("%d",&ch);
printf("Enter number of feed water heaters \n");
scanf("%d",&ch1);
printf("Enter boiler pressure in bar \n");
scanf("%f",&bp);
printf("Enter condensor pressure in bar \n");
scanf("%f",&cp);
if(cp<0.006)
{
printf("Condenser pressure cannot be less than 0.006 bar. Please input correct value \n");
exit(0);
}
if(bp<=cp)
{
printf("Boiler pressure should be greater than condenser pressure. Please run program again. \n");
exit(0);
}
if(bp>200)
{
printf("Boiler pressure should not exceed 200 bar. Please run program again. \n");
exit(0);
}
if(ch==0 && ch1==0) // Simple Rankine
{
printf("Simple Rankine cycle selected \n");
hp=bp;
printf("Enter temperature of steam at exit of boiler in Celcius \n");
scanf("%f",&temp_hp);
int j1;
switch((int)temp_hp)
```

```
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
j=2;
j2=3;
break;
}
```

```
if(temp_hp>100 && temp_hp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j2=3;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j2=4;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j2=5;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j2=6;break;
}
if(temp_hp>300 && temp_hp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j2=7; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j2=8;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j2=9;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j2=10;break;
}
```

```
if(temp_hp>600 && temp_hp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j2=11;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
if(status==1)
{
printf("Temperature entered is below the saturation temperature. Please run program again");
exit(0);
}
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);

if(t_sat_cp<77)
c_p=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p=2.1645;
if(t_sat_cp>=577 && t_sat_cp<627)
c_p=2.1995;
```

```
s3=s2;
if(s3>sgcp)
{
t3=(pow(2.718,((s3-sgcp)/c_p))*(t_sat_cp+273))-273;
h3=c_p*(t3-t_sat_cp) + hgcp;
}
if(s3<sgcp)
{
x3=(s3-sfcp)/sfgcp;
h3=hfcp+x3*hfgcp;
}
if(s3==sgcp)
{
h3=hgcp;
}
h4=hfcp;
h1=h4+vfcp*(bp-cp)*100;
work=(h2-h3)- (h1-h4);
efficiency=work/(h2-h1);
printf("Net work done in the cycle \t %f \nEfficiency of simple rankine cycle is \t %f %%
\n",work,efficiency*100);
if(x3<1)
printf("At maximum efficiency x = %f \t",x3);
else
exit(0);
}
if(ch==1 && ch1==0) //1 reheat 0 regen
{
printf("Single reheat Rankine cycle selected \n");
printf("Enter temperature at exit of boiler in C \n");
scanf("%f",&temp_hp)
printf("Enter step size in bar \n");
scanf("%f",&step);
hp=bp;
int j1,i;
count=0;
switch((int)temp_hp)
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
```

```
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
j=2;
j2=3;
break;
}
if(temp_hp>100 && temp_hp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j2=3;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j2=4;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
```

```
j=4,j2=5;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j2=6;break;
}
if(temp_hp>300 && temp_hp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j2=7; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j2=8;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j2=9;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j2=10;break;
}
if(temp_hp>600 && temp_hp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j2=11;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
```

```
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);
if(status==1)
{
printf("Temperature given is below saturation temperature of steam at given pressure. Run program again \n");
exit(0);
}
for(lp=hp-step;lp>cp;lp=lp-step)
{
x5=1;
interpol_table_b(lp,hflp,hglp,hfglp,sflp,sglp,sfglp,vflp,t_sat_lp);
if(t_sat_lp<77)
c_p=1.871;
if(t_sat_lp>=77 && t_sat_lp<127)
c_p=1.890;
if(t_sat_lp>=127 && t_sat_lp<177)
c_p=1.9135;
if(t_sat_lp>=177 && t_sat_lp<227)
c_p=1.94;
if(t_sat_lp>=227 && t_sat_lp<277)
c_p=1.969;
if(t_sat_lp>=277 && t_sat_lp<327)
c_p=1.9995;
if(t_sat_lp>=327 && t_sat_lp<377)
c_p=2.031;
if(t_sat_lp>=377 && t_sat_lp<427)
c_p=2.0635;
if(t_sat_lp>=427 && t_sat_lp<477)
c_p=2.0965;
if(t_sat_lp>=477 && t_sat_lp<527)
c_p=2.13;
if(t_sat_lp>=527 && t_sat_lp<577)
c_p=2.1645;
if(t_sat_lp>=577 && t_sat_lp<627)
c_p=2.1995;

interpol_superheat(lp,temp_hp,j1,h4,s4,tz);
s3=s2;
if(s3>sglp)
{
t3=(pow(2.718,((s3-sglp)/c_p))*(t_sat_lp+273)) - 273;
h3=c_p*(t3-t_sat_lp) + hglp;
x3=1;
}
if(s3<sglp)
{
x3=(s3-sflp)/sfglp;
h3=hflp+x3*hfglp;
}
if(s3==sglp)
{
h3=hglp;
```

```
x3=1;
}
s5=s4;
if(s5>sgcp)
{
t5=(pow(2.718,((s5-sgcp)/c_p))*(t_sat_cp+273))-273;
h5=c_p*(t5-t_sat_cp) + hgcp;
x5=1;
}
if(s5<sgcp)
{
x5=(s5-sfcp)/sfgcp;
h5=hfcp+x5*hfgcp;
}
if(s5==sgcp)
{
h5=hgcp;
x5=1;
}
h6=hfcp;
h1=h6+vfcp*(lp-cp);
work=(h2-h3) + (h4-h5) - (h1-h6);
efficiency=work/(h2-h1+h4-h3);
printf("Reheat pressure \t %f \t h1 \t %f h2 \t %f h3 \t %f \t h4 \t %f \t h5 \t %f \t h6 \t %f s2 \t %f \t s3 \t %f \t s4 \t %f \t s5 \t %f \t x3 \t %f \t x5 \t %f t3 \t %f \t t5 \t %f \t work \t %f \t efficiency \t %f \t \n \n",lp,h1,h2,h3,h4,h5,h6,s2,s3,s4,s5,x3,x5,t3,t5,work,efficiency);

p[l++]=lp;
eff[m++]=efficiency*100;
count++;
if(efficiency>max_efficiency)
{
max_efficiency=efficiency;
x=x5;
pr=lp;
}
}
printf("Max efficiency of cycle is %f %%\nAt intermediate pressure = \t %f bar",max_efficiency*100,pr);
if(x5<1)
printf("\n At which x =\t %f",x5);
else
exit(0);

}
if(ch==0 && ch1==1) //0 reheat 1 regeneration
{
printf("Single regenerative Rankine cycle selected \n");
hp=bp;
printf("Enter temperature at exit of boiler in C \n");
scanf("%f",&temp_hp);
printf("Enter step size in bar \n");
scanf("%f",&step);
int j1;
switch((int)temp_hp)
{
case 100:
```

```
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
j=2;
j2=3;
break;
}
if(temp_hp>100 && temp_hp<150)
{
```

```
flag=1;
gl=100;
lg=150;
j=2,j2=3;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j2=4;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j2=5;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j2=6;break;
}
if(temp_hp>300 && temp_hp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j2=7; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j2=8;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j2=9;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j2=10;break;
}
if(temp_hp>600 && temp_hp<700)
{
```

```
gl=600;
lg=700;
flag=1;
j=10,j2=11;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);

if(status==1)
{
printf("Temperature given is below saturation temperature of steam at given pressure. Please run program again
\n");
exit(0);
}
for(ip=hp-step; ip>cp; ip=ip-step)
{
interpol_table_b(ip,hfip,hgip,hfgip,sfip,sgip,sfgip,vfip,t_sat_ip);

if(t_sat_ip>=-23 && t_sat_ip<27)
c_p1=1.859;
if(t_sat_ip>=27 && t_sat_ip<77)
c_p1=1.871;
if(t_sat_ip>=77 && t_sat_ip<127)
c_p1=1.890;
if(t_sat_ip>=127 && t_sat_ip<177)
c_p1=1.9135;
if(t_sat_ip>=177 && t_sat_ip<227)
c_p1=1.94;
if(t_sat_ip>=227 && t_sat_ip<277)
c_p1=1.969;
if(t_sat_ip>=277 && t_sat_ip<327)
c_p1=1.9995;
if(t_sat_ip>=327 && t_sat_ip<377)
c_p1=2.031;
if(t_sat_ip>=377 && t_sat_ip<427)
c_p1=2.0635;
if(t_sat_ip>=427 && t_sat_ip<477)
c_p1=2.0965;
if(t_sat_ip>=477 && t_sat_ip<527)
c_p1=2.13;
```

```
if(t_sat_ip>=527 && t_sat_ip<577)
c_p1=2.1645;
if(t_sat_ip>=577 && t_sat_ip<627)
c_p1=2.1995;

if(t_sat_cp>=-23 && t_sat_cp<27)
c_p2=1.859;
if(t_sat_cp>=27 && t_sat_cp<77)
c_p2=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p2=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p2=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p2=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p2=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p2=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p2=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p2=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p2=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p2=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p2=2.1645;
if(t_sat_cp>=577 && t_sat_cp<627)
c_p2=2.1995;

s6=s2;
s3=s2;
if(s6>sgip)
{
t6=(pow(2.718,((s6-sgip)/c_p1))*(t_sat_ip+273)) - 273;

h6=c_p1*(t6-t_sat_ip) + hgip;
}
if(s6<sgip)
{
x6=(s6-sfip)/sfgip;
h6=hfip+x6*hfgip;
}
if(s6==sgip)
{
h6=hgip;
}
if(s3>sgcp)
{
t3=(pow(2.718,((s6-sgcp)/c_p2))*(t_sat_cp+273)) - 273;
h3=c_p2*(t3-t_sat_cp) + hgcp;
}
if(s3<sgcp)
{
```

```
x3=(s3-sfcp)/sfgcp;
h3=hfcp+x3*hfgcp;
}
if(s3==sgcp)
{
h3=hgcp;
}
h7=hfip;
h4=hfcp;
h5=h4+vfcp*(ip-cp)*100;
h1=h7+vfip*(hp-ip)*100;
float m1=(h7-h5)/(h6-h5);
work=(h2-h6)+ (1-m1)*(h6-h3) - (1-m1)*(h5-h4) - (h1-h7);
efficiency= work/(h2-h1);
p[l++]=ip;
eff[m++]=efficiency*100;
count++;
printf("\n Regeneration pressure \t %f \t efficiency \t %f \n",ip,efficiency);
if(x3<1)
printf("x:\t %f",x3);
else
exit(0);
if(efficiency>max_efficiency)
{
max_efficiency=efficiency;
x=x3;
pr=ip;
}
}
printf("\n Max efficiency of cycle is %f %% \nAt intermediate pressure = \t %f bar",max_efficiency*100,pr);
if(x3<1)
printf("\n At which x = \t %f \t",x3);
else
exit(0);
plotdata x(p,count);
plotdata y(eff,count);
axesBotLeft(x, y, 0, 20);
axesTopRight(x, y, 150, 50);
for(i=0;i<count;i++)
{
addMark(x, y, p[i], eff[i]);
}
plot(x,y);
}
if(ch==0 && ch1==2) // 0 reheat double regeneration
{
printf("Double regenerative Rankine cycle selected \n");
printf("Enter temperature at the exit of boiler in C \n");
scanf("%f",&temp_hp);
printf("Enter step size for first regeneration pressure in bar \n");
scanf("%f",&step1);
printf("Enter step size for second regeneration pressure in bar \n");
scanf("%f",&step2);
hp=bp;
int j1;
switch((int)temp_hp)
```

```
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
j=2;
j2=3;
break;
}
```

```
if(temp_hp>100 && temp_hp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j2=3;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j2=4;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j2=5;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j2=6;break;
}
if(temp_hp>300 && temp_hp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j2=7; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j2=8;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j2=9;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j2=10;break;
}
```

```
if(temp_hp>600 && temp_hp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j2=11;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);
if(status==1)
{
printf("Temperature given is below saturation temperature of steam at given pressure. Run program again \n");
exit(0);
}
for(ip1=hp-step1; ip1>cp ; ip1=ip1-step1)
{
for(ip2=ip1-step2; ip2>cp; ip2=ip2-step2)
{
interpol_table_b(ip1,hfip1,hgip1,hfgip1,sfip1,sgip1,sfgip1,vfip1,t_sat_ip1);
interpol_table_b(ip2,hfip2,hgip2,hfgip2,sfip2,sgip2,sfgip2,vfip2,t_sat_ip2);
if(t_sat_ip1<77)
c_p1=1.871;
if(t_sat_ip1>=77 && t_sat_ip1<127)
c_p1=1.890;
if(t_sat_ip1>=127 && t_sat_ip1<177)
c_p1=1.9135;
if(t_sat_ip1>=177 && t_sat_ip1<227)
c_p1=1.94;
if(t_sat_ip1>=227 && t_sat_ip1<277)
c_p1=1.969;
if(t_sat_ip1>=277 && t_sat_ip1<327)
c_p1=1.9995;
if(t_sat_ip1>=327 && t_sat_ip1<377)
c_p1=2.031;
if(t_sat_ip1>=377 && t_sat_ip1<427)
c_p1=2.0635;
if(t_sat_ip1>=427 && t_sat_ip1<477)
c_p1=2.0965;
if(t_sat_ip1>=477 && t_sat_ip1<527)
c_p1=2.13;
```

```
if(t_sat_ip1>=527 && t_sat_ip1<577)
c_p1=2.1645;
if(t_sat_ip1>=577 && t_sat_ip1<627)
c_p1=2.1995;


if(t_sat_ip2<77)
c_p2=1.871;
if(t_sat_ip2>=77 && t_sat_ip2<127)
c_p2=1.890;
if(t_sat_ip2>=127 && t_sat_ip2<177)
c_p2=1.9135;
if(t_sat_ip2>=177 && t_sat_ip2<227)
c_p2=1.94;
if(t_sat_ip2>=227 && t_sat_ip2<277)
c_p2=1.969;
if(t_sat_ip2>=277 && t_sat_ip2<327)
c_p2=1.9995;
if(t_sat_ip2>=327 && t_sat_ip2<377)
c_p2=2.031;
if(t_sat_ip2>=377 && t_sat_ip2<427)
c_p2=2.0635;
if(t_sat_ip2>=427 && t_sat_ip2<477)
c_p2=2.0965;
if(t_sat_ip2>=477 && t_sat_ip2<527)
c_p2=2.13;
if(t_sat_ip2>=527 && t_sat_ip2<577)
c_p2=2.1645;
if(t_sat_ip2>=577 && t_sat_ip2<627)
c_p2=2.1995;

if(t_sat_cp<77)
c_p3=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p3=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p3=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p3=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p3=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p3=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p3=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p3=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p3=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p3=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p3=2.1645;
if(t_sat_cp>=577 && t_sat_cp<627)
c_p3=2.1995;
s3=s2;
```

```
if(s3>sgip1)
{
t3=(pow(2.718,((s3-sgip1)/c_p1))*(t_sat_ip1+273)) -273;
h3=c_p1*(t3-t_sat_ip1) + hgip1;
}
if(s3<sgip1)
{
x3=(s3-sfip1)/sfgip1;
h3=hfip1+x3*hfgip1;
}
if(s3==sgip1)
{
h3=hgip1;
}
s4=s2;
if(s4>sgip2)
{
t4=(pow(2.718,((s4-sgip2)/c_p2))*(t_sat_ip2+273)) -273;
h4=c_p2*(t4-t_sat_ip2) + hgip2;
}
if(s4<sgip2)
{
x4=(s4-sfip2)/sfgip2;
h4=hfip2+x4*hfgip2;
}
if(s4==sgip2)
{
h4=hgip2;
}
s5=s2;
if(s5>sgcp)
{
t5=(pow(2.718,((s5-sgcp)/c_p3))*(t_sat_cp+273)) - 273;
h5=c_p3*(t5-t_sat_cp) + hgcp;
}
if(s5<sgcp)
{
x5=(s5-sfcp)/sfgcp;
h5=hfcp+x5*hfgcp;
}
if(s5==sgcp)
{
h5=hgcp;
}
h6=hfcp;
h8=hfip2;
h10=hfip1;
h7=h6+vfcp*(ip2-cp)*100;
h9=h8+vfip2*(ip1-ip2)*100;
h1=h10+vfip1*(hp-ip1)*100;
float m1=(h10-h9)/(h3-h9);
float m2=(h8-m1*h8-h7+m1*h7)/(h4-h7);
efficiency1=((h2-h3) +(1-m1)*(h3-h4) + (1-m1-m2)*(h4-h5) - (1-m1-m2)*(h7-h6) - (1-m1)*(h9-h8) - (h1-
h10))/(h2-h1);
if(efficiency1>max_efficiency)
{
```

```
max_efficiency=efficiency1;
p4=ip1;
x=x5;
}
}
}
ip1=p4;
printf("For first regeneration pressure optimum %f \t \n",ip1);
for(ip2=ip1-step2; ip2>cp; ip2=ip2-step2)
{
interpol_table_b(ip1,hfip1,hgip1,hfgip1,sfip1,sgip1,sfgip1,vfip1,t_sat_ip1);
interpol_table_b(ip2,hfip2,hgip2,hfgip2,sfip2,sgip2,sfgip2,vfip2,t_sat_ip2);
if(t_sat_ip1<77)
c_p1=1.871;
if(t_sat_ip1>=77 && t_sat_ip1<127)
c_p1=1.890;
if(t_sat_ip1>=127 && t_sat_ip1<177)
c_p1=1.9135;
if(t_sat_ip1>=177 && t_sat_ip1<227)
c_p1=1.94;
if(t_sat_ip1>=227 && t_sat_ip1<277)
c_p1=1.969;
if(t_sat_ip1>=277 && t_sat_ip1<327)
c_p1=1.9995;
if(t_sat_ip1>=327 && t_sat_ip1<377)
c_p1=2.031;
if(t_sat_ip1>=377 && t_sat_ip1<427)
c_p1=2.0635;
if(t_sat_ip1>=427 && t_sat_ip1<477)
c_p1=2.0965;
if(t_sat_ip1>=477 && t_sat_ip1<527)
c_p1=2.13;
if(t_sat_ip1>=527 && t_sat_ip1<577)
c_p1=2.1645;
if(t_sat_ip1>=577 && t_sat_ip1<627)
c_p1=2.1995;


if(t_sat_ip2<77)
c_p2=1.871;
if(t_sat_ip2>=77 && t_sat_ip2<127)
c_p2=1.890;
if(t_sat_ip2>=127 && t_sat_ip2<177)
c_p2=1.9135;
if(t_sat_ip2>=177 && t_sat_ip2<227)
c_p2=1.94;
if(t_sat_ip2>=227 && t_sat_ip2<277)
c_p2=1.969;
if(t_sat_ip2>=277 && t_sat_ip2<327)
c_p2=1.9995;
if(t_sat_ip2>=327 && t_sat_ip2<377)
c_p2=2.031;
if(t_sat_ip2>=377 && t_sat_ip2<427)
c_p2=2.0635;
if(t_sat_ip2>=427 && t_sat_ip2<477)
c_p2=2.0965;
```

```
if(t_sat_ip2>=477 && t_sat_ip2<527)
c_p2=2.13;
if(t_sat_ip2>=527 && t_sat_ip2<577)
c_p2=2.1645;
if(t_sat_ip2>=577 && t_sat_ip2<627)
c_p2=2.1995;

if(t_sat_cp<77)
c_p3=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p3=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p3=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p3=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p3=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p3=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p3=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p3=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p3=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p3=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p3=2.1645;
if(t_sat_cp>=577 && t_sat_cp<627)
c_p3=2.1995;
s3=s2;
if(s3>sgip1)
{
t3=(pow(2.718,((s3-sgip1)/c_p1))*(t_sat_ip1+273)) -273;
h3=c_p1*(t3-t_sat_ip1) + hgip1;
}
if(s3<sgip1)
{
x3=(s3-sfip1)/sfgip1;
h3=hfip1+x3*hfgip1;
}
if(s3==sgip1)
{
h3=hgip1;
}
s4=s2;
if(s4>sgip2)
{
t4=(pow(2.718,((s4-sgip2)/c_p2))*(t_sat_ip2+273)) -273;
h4=c_p2*(t4-t_sat_ip2) + hgip2;
}
if(s4<sgip2)
{
x4=(s4-sfip2)/sfgip2;
h4=hfip2+x4*hfgip2;
```

```
}
if(s4==sgip2)
{
h4=hgip2;
}
s5=s2;
if(s5>sgcp)
{
t5=(pow(2.718,((s5-sgcp)/c_p3))*(t_sat_cp+273)) - 273;
h5=c_p3*(t5-t_sat_cp) + hgcp;
}
if(s5<sgcp)
{
x5=(s5-sfcp)/sfgcp;
h5=hfcp+x5*hfgcp;
}
if(s5==sgcp)
{
h5=hgcp;
}
h6=hfcp;
h8=hfip2;
h10=hfip1;
h7=h6+vfcp*(ip2-cp)*100;
h9=h8+vfip2*(ip1-ip2)*100;
h1=h10+vfip1*(hp-ip1)*100;
float m1=(h10-h9)/(h3-h9);
float m2=(h8-m1*h8-h7+m1*h7)/(h4-h7);
efficiency2=((h2-h3) +(1-m1)*(h3-h4) + (1-m1-m2)*(h4-h5) - (1-m1-m2)*(h7-h6) - (1-m1)*(h9-h8) - (h1-
h10))/(h2-h1);
p[l++]=ip2;
eff[m++]=efficiency2*100;
count++;
printf("Second regeneration pressure \t %f \t Efficiency \t %f \t x: \t %f \t \n",ip2,efficiency2,x5);
if(efficiency2>=max_efficiency)
{
max_efficiency=efficiency2;
p5=ip2;
x=x5;
}
}
printf("\nMaximum efficiency of the cycle is \t %f \nAt first Regeneration pressure \t %f \nand second
regeneration pressure \t %f \t \n",max_efficiency,p4,p5);
if(x5<1)
printf("At which x:\t %f \n",x5);
else
exit(0);
}
if(ch==1 && ch1==1) // Single reheat single regen open
{
printf("Single reheat single regenerative Rankine cycle selected \n");
hp=bp;
printf("Enter temperature at exit of boiler in C \n");
scanf("%f",&temp_hp);
printf("Enter step size for reheat pressure \n");
scanf("%f",&step1);
```

```
printf("Enter step size for feed water heater pressure \n");
scanf("%f",&step2);
int j1;
switch((int)temp_hp)
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
```

```
j=2,j1=3,j2=4; break;
}
if(temp_hp>100 && temp_hp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j1=3,j2=4;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j1=4,j2=5;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j1=5,j2=6;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j1=6,j2=7;break;
}
if(temp_hp>300 && temp_hp<350)
{
gl=300;
lg=350;
flag=1;
j=6,j1=7,j2=8; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j1=8,j2=9;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j1=9,j2=10;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
```

```
j=9,j1=10,j2=11;break;
}
if(temp_hp>600 && temp_hp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j1=11,j2=12;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);
if(status==1)
{
printf("Temperature given is below saturation temperature of steam at given pressure. Run program again \n");
exit(0);
}
for(ip1=hp-step1; ip1>cp ; ip1=ip1-step1)
{
for(ip2=ip1-step2; ip2>cp; ip2=ip2-step2)
{
interpol_table_b(ip1,hfip1,hgip1,hfgip1,sfip1,sgip1,sfgip1,vfip1,t_sat_ip1);
interpol_table_b(ip2,hfip2,hgip2,hfgip2,sfip2,sgip2,sfgip2,vfip2,t_sat_ip2);
s3=s2;
if(t_sat_ip1<77)
c_p1=1.871;
if(t_sat_ip1>=77 && t_sat_ip1<127)
c_p1=1.890;
if(t_sat_ip1>=127 && t_sat_ip1<177)
c_p1=1.9135;
if(t_sat_ip1>=177 && t_sat_ip1<227)
c_p1=1.94;
if(t_sat_ip1>=227 && t_sat_ip1<277)
c_p1=1.969;
if(t_sat_ip1>=277 && t_sat_ip1<327)
c_p1=1.9995;
if(t_sat_ip1>=327 && t_sat_ip1<377)
c_p1=2.031;
if(t_sat_ip1>=377 && t_sat_ip1<427)
c_p1=2.0635;
if(t_sat_ip1>=427 && t_sat_ip1<477)
```

```
c_p1=2.0965;
if(t_sat_ip1>=477 && t_sat_ip1<527)
c_p1=2.13;
if(t_sat_ip1>=527 && t_sat_ip1<577)
c_p1=2.1645;
if(t_sat_ip1>=577 && t_sat_ip1<627)
c_p1=2.1995;


if(t_sat_ip2<77)
c_p2=1.871;
if(t_sat_ip2>=77 && t_sat_ip2<127)
c_p2=1.890;
if(t_sat_ip2>=127 && t_sat_ip2<177)
c_p2=1.9135;
if(t_sat_ip2>=177 && t_sat_ip2<227)
c_p2=1.94;
if(t_sat_ip2>=227 && t_sat_ip2<277)
c_p2=1.969;
if(t_sat_ip2>=277 && t_sat_ip2<327)
c_p2=1.9995;
if(t_sat_ip2>=327 && t_sat_ip2<377)
c_p2=2.031;
if(t_sat_ip2>=377 && t_sat_ip2<427)
c_p2=2.0635;
if(t_sat_ip2>=427 && t_sat_ip2<477)
c_p2=2.0965;
if(t_sat_ip2>=477 && t_sat_ip2<527)
c_p2=2.13;
if(t_sat_ip2>=527 && t_sat_ip2<577)
c_p2=2.1645;
if(t_sat_ip2>=577 && t_sat_ip2<627)
c_p2=2.1995;

if(t_sat_cp<77)
c_p3=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p3=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p3=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p3=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p3=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p3=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p3=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p3=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p3=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p3=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p3=2.1645;
```

```
if(t_sat_cp>=577 && t_sat_cp<627)
c_p3=2.1995;

if(s3>sgip1)
{
t3=(pow(2.718,((s3-sgip1)/c_p1))*(t_sat_ip1+273))-273;
h3=c_p1*(t3-t_sat_ip1) + hgip1;
}
if(s3<sgip1)
{
x3=(s3-sfip1)/sfgip1;
h3=hfip1+x3*hfgip1;
}
if(s3==sgip1)
{
h3=hgip1;
}
interpol_superheat(ip1,temp_hp,j1,h4,s4,tz);
s5=s4;
if(s5>sgip2)
{
t5=(pow(2.718,((s5-sgip2)/c_p2))*(t_sat_ip2+273))-273;
h5=c_p2*(t5-t_sat_ip2) + hgip2;
}
if(s5<sgip2)
{
x5=(s5-sfip2)/sfgip2;
h5=hfip2+x5*hfgip2;
}
if(s5==sgip2)
{
h5=hgip2;
}
s6=s4;
if(s6>sgcp)
{
t6=(pow(2.718,((s6-sgcp)/c_p3))*(t_sat_cp+273));
h6=c_p3*(t6-t_sat_cp) + hgcp;
}
if(s6<sgcp)
{
x6=(s6-sfcp)/sfgcp;
h6=hfcp+x6*hfgcp;
}
if(s6==sgcp)
{
h6=hgcp;
}
h7=hfcp;
h9=hfip2;
h8=h7+vfcp*(ip2-cp)*100;
h1=h9+vfip2*(hp-ip2)*100;
float m1=(h9-h8)/(h5-h8);
efficiency1=((h2-h3) +(h4-h5)+(1-m1)*(h5-h6)-(1-m1)*(h8-h7)-(h1-h9))/((h2-h1) + (h4-h3));
if(efficiency1>max_efficiency)
{
```

```
max_efficiency=efficiency1;
p4=ip1;
p5=ip2;
x=x6;
}
}
}
printf("\nMaximum efficiency of cycle is \t %f \nAt which Reheat pressure is \t %f \nRegeneration pressure is \t
%f \n",max_efficiency,p4,p5);
if(x6<1)
printf("At which x is \t %f \n",x6);
else
exit(0);
}
if(ch==1 && ch1==2) // Single reheat open-open
{
printf("Single reheat double regenerative Rankine cycle selected \n");
hp=bp;
printf("Enter temperature at exit of boiler in C \n");
scanf("%f",&temp_hp);
printf("Enter step size for first feed water heater pressure in bar\n");
scanf("%f",&step1);
printf("Enter step size for reheat pressure in bar \n");
scanf("%f",&step2);
printf("Enter step size for second feed water heater pressure in bar \n");
scanf("%f",&step3);
int j1;
switch((int)temp_hp)
{
case 100:
{
j1 =2; break;
}
case 150:
{
j1=3; break;
}
case 200:
{
j1=4;break;
}
case 250:
{
j1=5;break;
}
case 300:
{
j1=6; break;
}
case 350:
{
j1=7; break;
}
case 400:
{
j1=8; break;
```

```
}
case 500:
{
j1=9; break;
}
case 600:
{
j1=10; break;
}
case 700:
{
j1=11; break;
}
case 800:
{
j1=12; break;
}
default:
{
if (temp_hp<100)
{
flag=1;
gl=100;
lg=150;
j=2,j1=3,j2=4; break;
}
if(temp_hp>100 && temp_hp<150)
{
flag=1;
gl=100;
lg=150;
j=2,j1=3,j2=4;break;
}
if(temp_hp>150 && temp_hp<200)
{
gl=150;
lg=200;
flag=1;
j=3,j1=4,j2=5;break;
}
if(temp_hp>200 && temp_hp<250)
{
gl=200;
lg=250;
flag=1;
j=4,j1=5,j2=6;break;
}
if(temp_hp>250 && temp_hp<300)
{
gl=250;
lg=300;
flag=1;
j=5,j1=6,j2=7;break;
}
if(temp_hp>300 && temp_hp<350)
{
```

```
gl=300;
lg=350;
flag=1;
j=6,j1=7,j2=8; break;
}
if(temp_hp>350 && temp_hp<400)
{
gl=350;
lg=400;
flag=1;
j=7,j1=8,j2=9;break;
}
if(temp_hp>400 && temp_hp<500)
{
gl=400;
lg=500;
flag=1;
j=8,j1=9,j2=10;break;
}
if(temp_hp>500 && temp_hp<600)
{
gl=500;
lg=600;
flag=1;
j=9,j1=10,j2=11;break;
}
if(temp_hp>600 && temp_hp<700)
{
gl=600;
lg=700;
flag=1;
j=10,j1=11,j2=12;break;
}
if(temp_hp>700 && temp_hp<800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
if(temp_hp>800)
{
gl=700;
lg=800;
flag=1;
j=11; j2=12; break;
}
}
}
interpol_superheat(hp,temp_hp,j1,h2,s2,tz);
interpol_table_b(cp,hfcp,hgcp,hfgcp,sfcp,sgcp,sfgcp,vfcp,t_sat_cp);
if(status==1)
{
printf("Temperature given is below saturation temperature of steam at given pressure. Run program again \n");
exit(0);
}
```

```
for(ip1=hp-step1; ip1>cp ; ip1=ip1-step1)
{
for(ip2=hp-step2; ip2>ip1; ip2=ip2-step2)
{
for(ip3=ip1-step3; ip3>cp; ip3=ip3-step3)
{
interpol_table_b(ip1,hfip1,hgip1,hfgip1,sfip1,sgip1,sfgip1,vfip1,t_sat_ip1);
interpol_table_b(ip2,hfip2,hgip2,hfgip2,sfip2,sgip2,sfgip2,vfip2,t_sat_ip2);
interpol_table_b(ip3,hfip3,hgip3,hfgip3,sfip3,sgip3,sfgip3,vfip3,t_sat_ip3);

if(t_sat_ip1<77)
c_p1=1.871;
if(t_sat_ip1>=77 && t_sat_ip1<127)
c_p1=1.890;
if(t_sat_ip1>=127 && t_sat_ip1<177)
c_p1=1.9135;
if(t_sat_ip1>=177 && t_sat_ip1<227)
c_p1=1.94;
if(t_sat_ip1>=227 && t_sat_ip1<277)
c_p1=1.969;
if(t_sat_ip1>=277 && t_sat_ip1<327)
c_p1=1.9995;
if(t_sat_ip1>=327 && t_sat_ip1<377)
c_p1=2.031;
if(t_sat_ip1>=377 && t_sat_ip1<427)
c_p1=2.0635;
if(t_sat_ip1>=427 && t_sat_ip1<477)
c_p1=2.0965;
if(t_sat_ip1>=477 && t_sat_ip1<527)
c_p1=2.13;
if(t_sat_ip1>=527 && t_sat_ip1<577)
c_p1=2.1645;
if(t_sat_ip1>=577 && t_sat_ip1<627)
c_p1=2.1995;


if(t_sat_ip2<77)
c_p2=1.871;
if(t_sat_ip2>=77 && t_sat_ip2<127)
c_p2=1.890;
if(t_sat_ip2>=127 && t_sat_ip2<177)
c_p2=1.9135;
if(t_sat_ip2>=177 && t_sat_ip2<227)
c_p2=1.94;
if(t_sat_ip2>=227 && t_sat_ip2<277)
c_p2=1.969;
if(t_sat_ip2>=277 && t_sat_ip2<327)
c_p2=1.9995;
if(t_sat_ip2>=327 && t_sat_ip2<377)
c_p2=2.031;
if(t_sat_ip2>=377 && t_sat_ip2<427)
c_p2=2.0635;
if(t_sat_ip2>=427 && t_sat_ip2<477)
c_p2=2.0965;
if(t_sat_ip2>=477 && t_sat_ip2<527)
c_p2=2.13;
```

```
if(t_sat_ip2>=527 && t_sat_ip2<577)
c_p2=2.1645;
if(t_sat_ip2>=577 && t_sat_ip2<627)
c_p2=2.1995;

if(t_sat_ip3<77)
c_p3=1.871;
if(t_sat_ip3>=77 && t_sat_ip3<127)
c_p3=1.890;
if(t_sat_ip3>=127 && t_sat_ip3<177)
c_p3=1.9135;
if(t_sat_ip3>=177 && t_sat_ip3<227)
c_p3=1.94;
if(t_sat_ip3>=227 && t_sat_ip3<277)
c_p3=1.969;
if(t_sat_ip3>=277 && t_sat_ip3<327)
c_p3=1.9995;
if(t_sat_ip3>=327 && t_sat_ip3<377)
c_p3=2.031;
if(t_sat_ip3>=377 && t_sat_ip3<427)
c_p3=2.0635;
if(t_sat_ip3>=427 && t_sat_ip3<477)
c_p3=2.0965;
if(t_sat_ip3>=477 && t_sat_ip3<527)
c_p3=2.13;
if(t_sat_ip3>=527 && t_sat_ip3<577)
c_p3=2.1645;
if(t_sat_ip3>=577 && t_sat_ip3<627)
c_p3=2.1995;

if(t_sat_cp<77)
c_p4=1.871;
if(t_sat_cp>=77 && t_sat_cp<127)
c_p4=1.890;
if(t_sat_cp>=127 && t_sat_cp<177)
c_p4=1.9135;
if(t_sat_cp>=177 && t_sat_cp<227)
c_p4=1.94;
if(t_sat_cp>=227 && t_sat_cp<277)
c_p4=1.969;
if(t_sat_cp>=277 && t_sat_cp<327)
c_p4=1.9995;
if(t_sat_cp>=327 && t_sat_cp<377)
c_p4=2.031;
if(t_sat_cp>=377 && t_sat_cp<427)
c_p4=2.0635;
if(t_sat_cp>=427 && t_sat_cp<477)
c_p4=2.0965;
if(t_sat_cp>=477 && t_sat_cp<527)
c_p4=2.13;
if(t_sat_cp>=527 && t_sat_cp<577)
c_p4=2.1645;
if(t_sat_cp>=577 && t_sat_cp<627)
c_p4=2.1995;
s3=s2;
if(s3>sgip2)
```

```
{
t3=(pow(2.718,((s3-sgip2)/c_p2))*(t_sat_ip2+273))-273;
h3=c_p2*(t3-t_sat_ip2) + hgip2;
}
if(s3<sgip2)
{
x3=(s3-sfip2)/sfgip2;
h3=hfip2+x3*hfgip2;
}
if(s3==sgip2)
{
h3=hgip2;
}
s4=s2;
if(s4>sgip1)
{
t4=(pow(2.718,((s4-sgip1)/c_p1))*(t_sat_ip1+273))-273;
h4=c_p1*(t4-t_sat_ip1) + hgip1;
}
if(s4<sgip1)
{
x4=(s4-sfip1)/sfgip1;
h4=hfip1+x4*hfgip1;
}
if(s4==sgip1)
{
h4=hgip1;
}
interpol_superheat(ip1,temp_hp,j1,h5,s5,tz);
s6=s5;
if(s6>sgip3)
{
t6=(pow(2.718,((s6-sgip3)/c_p3))*(t_sat_ip3+273))-273;
h6=c_p3*(t6-t_sat_ip3) + hgip3;
}
if(s6<sgip3)
{
x6=(s6-sfip3)/sfgip3;
h6=hfip3+x6*hfgip3;
}
if(s6==sgip3)
{
h6=hgip3;
}
s7=s5;
if(s7>sgcp)
{
t7=(pow(2.718,((s7-sgcp)/c_p4))*(t_sat_cp+273));
h7=c_p4*(t7-t_sat_cp) + hgcp;
}
if(s7<sgcp)
{
x7=(s7-sfcp)/sfgcp;
h7=hfcp+x7*hfgcp;
}
if(s7==sgcp)
```

```
{
h7=hgcp;
}
h8=hfcp;
h9=h8+vfcp*(ip3-cp);
h10=hfip3;
h11=h10+vfip3*(ip1-ip3);
h12=hfip2;
h1=h12+vfip2*(hp-ip2);
float m1=(h12-h11)/(h3-h11);
float m2=((1-m1)*(h10-h9))/(h6-h9);
work=(h2-h3)+(1-m1)*(h3-h4)+(1-m1)*(h5-h6)+(1-m1-m2)*(h6-h7)-(1-m1-m2)*(h9-h8)-(1-m1)*(h11-h10)-
(h1-h12);
efficiency1=work/(h2-h1+h5-h4);
if(efficiency1>max_efficiency)
{
max_efficiency=efficiency1;
p3=ip1;
p4=ip2;
p5=ip3;
x=x7;
}
}
}
}
printf("Maximum efficiency \t %f \t \nAt reheat pressure \t %f \t \nFirst regeneration pressure \t %f \t \nSecond
regeneration pressure \t %f \t \n",max_efficiency,p3,p4,p5);
}
}
}
void interpol_superheat(float press,int temp_hp,int j1,float &hh,float &ss,float &tt)
{
int i,pt1,found1=0,pos1,pos2,p1,p2,ptt1,f1=0;
float greatest_less=0,least_great=0,tt1,tt2,lt_gt,gt_ls=0;
float hh1,hh2,ss1,ss2,hh11,hh22,ss11,ss22,hh_old1,hh_old2,ss_old1,ss_old2,hha,hhb,ssa,ssb;
for(i=0;i<124;i++)
{
if(b[i][0]==press*100)
{
ptt1=i;
f1=1;
}
}

if(f1==1)
{
if(temp_hp<b[ptt1][1])
{
status=1;
}
}
if(f1==0)
{
lt_gt=b[123][0];
for(i=0;i<124;i++)
{
```

```
if(b[i][0]>=gt_ls && b[i][0]<press*100)
{
gt_ls=b[i][0];
p1=i;
}
}
for(i=123;i>=0;i--)
{
if(b[i][0]<=lt_gt && b[i][0]>press*100)
{
lt_gt=b[i][0];
p2=i;
}
}
tt1=b[p1][1];
tt2=b[p2][1];
tt=(((press*100-gt_ls)/(lt_gt-gt_ls))*(tt2-tt1))+tt1;
if(temp_hp<tt)
{
status=1;
}
}

for(i=0;i<79;i++)
{
if(d[i][0]==press)
{
pt1=i;
found1=1;
}
}
if(found1==1)
{
if(flag==0)
{
hh=d[pt1][j1];
ss=e[pt1][j1];
}
if(flag==1)
{
hha=d[pt1][j];
hhb=d[pt1][j2];
hh=(((temp_hp-gl)/(lg-gl))*(hhb-hha))+hha;
ssa=e[pt1][j];
ssb=e[pt1][j2];
ss=(((temp_hp-gl)/(lg-gl))*(ssb-ssa))+ssa;
}
}
if(found1==0 && flag==0)
{
least_great=d[78][0];
for(i=0;i<79;i++)
{
if(d[i][0]>=greatest_less && d[i][0]<press)
{
greatest_less=d[i][0];
```

```
pos1=i;
}
}
for(i=78;i>=0;i--)
{
if(d[i][0]<=least_great && d[i][0]>press)
{
least_great=d[i][0];
pos2=i;
}
}
hh1=d[pos1][j1];
hh2=d[pos2][j1];
hh=(((press-greatest_less)/(least_great-greatest_less))*(hh2-hh1))+hh1;
ss1=e[pos1][j1];
ss2=e[pos2][j1];
ss=(((press-greatest_less)/(least_great-greatest_less))*(ss2-ss1))+ss1;
}
if(found1==0 && flag==1)
{
least_great=d[78][0];
for(i=0;i<79;i++)
{
if(d[i][0]>=greatest_less && d[i][0]<press)
{
greatest_less=d[i][0];
pos1=i;
}
}
for(i=78;i>=0;i--)
{
if(d[i][0]<=least_great && d[i][0]>press)
{
least_great=d[i][0];
pos2=i;
}
}
hh1=d[pos1][j];
hh2=d[pos2][j];
hh_old1=(((press-greatest_less)/(least_great-greatest_less))*(hh2-hh1))+hh1;
hh11=d[pos1][j2];
hh22=d[pos2][j2];
hh_old2=(((press-greatest_less)/(least_great-greatest_less))*(hh22-hh11))+hh11;
hh=(((temp_hp-gl)/(lg-gl))*(hh_old2-hh_old1))+hh_old1;
ss1=e[pos1][j];
ss2=e[pos2][j];
ss_old1=(((press-greatest_less)/(least_great-greatest_less))*(ss2-ss1))+ss1;
ss11=e[pos1][j2];
ss22=e[pos2][j2];
ss_old2=(((press-greatest_less)/(least_great-greatest_less))*(ss22-ss11))+ss11;
ss=(((temp_hp-gl)/(lg-gl))*(ss_old2-ss_old1))+ss_old1;
}
}
void interpol_table_b(float press,float &hf,float &hg,float &hfg,float &sf,float &sg,float &sfg,float &vf,float
&t_sat)
{
```

```
int i,ppt1,found2=0,pos1,pos2;
float greatest_less=0,least_great=0;
float vf1,vf2,hf1,hf2,hg1,hg2,hfg1,hfg2,sf1,sf2,sg1,sg2,sfg1,sfg2,t_sat1,t_sat2;
for(i=0;i<124;i++)
{
if(b[i][0]==press*100)
{
ppt1=i;
found2=1;
}
}
if(found2==1)
{
sf=b[ppt1][11];
sg=b[ppt1][12];
sfg=b[ppt1][13];
hf=b[ppt1][5];
hg=b[ppt1][6];
hfg=b[ppt1][7];
vf=b[ppt1][2];
t_sat=b[ppt1][1];
}
if(found2==0)
{
least_great=b[123][0];
for(i=0;i<124;i++)
{
if(b[i][0]>=greatest_less && b[i][0]<press*100)
{
greatest_less=b[i][0];
pos1=i;
}
}
for(i=123;i>=0;i--)
{
if(b[i][0]<=least_great && b[i][0]>press*100)
{
least_great=b[i][0];
pos2=i;
}
}
vf1=b[pos1][2];
vf2=b[pos2][2];
vf=(((press*100-greatest_less)/(least_great-greatest_less))*(vf2-vf1))+vf1;
hf1=b[pos1][5];
hf2=b[pos2][5];
hf=(((press*100-greatest_less)/(least_great-greatest_less))*(hf2-hf1))+hf1;
hg1=b[pos1][6];
hg2=b[pos2][6];
hg=(((press*100-greatest_less)/(least_great-greatest_less))*(hg2-hg1))+hg1;
hfg1=b[pos1][7];
hfg2=b[pos2][7];
hfg=(((press*100-greatest_less)/(least_great-greatest_less))*(hfg2-hfg1))+hfg1;
sf1=b[pos1][11];
sf2=b[pos2][11];
sf=(((press*100-greatest_less)/(least_great-greatest_less))*(sf2-sf1))+sf1;
```

```
sfg1=b[pos1][13];
sfg2=b[pos2][13];
sfg=(((press*100-greatest_less)/(least_great-greatest_less))*(sfg2-sfg1))+sfg1;
sg1=b[pos1][12];
sg2=b[pos2][12];
sg=(((press*100-greatest_less)/(least_great-greatest_less))*(sg2-sg1))+sg1;
t_sat1=b[pos1][1];
t_sat2=b[pos2][1];
t_sat=(((press*100-greatest_less)/(least_great-greatest_less))*(t_sat2-t_sat1))+t_sat1;
}
}
```