Convection in Fluids

1 Objective

To study the convection in fluids using the Lorentz system

2 Introduction

Convection is the concerted, collective movement of ensembles of molecules within fluids. Convective heat transfer is one of the major modes of heat transfer and convection is also a major mode of mass transfer in fluids. The broader sense is in fluid mechanics, where convection refers to fluid motion regardless of cause. However, in thermodynamics, convection specifically refers to heat transfer by convection, often. Additionally, convection includes fluid movement, both by bulk motion (advection) and by the motion of individual particles (diffusion). However, in some cases, convection is taken to mean only advective phenomena. For instance, in the transport equation, which describes several different transport phenomena, terms are separated into convective and diffusive effects, with convective meaning purely advective in context.
A two-dimensional fluid cell is warmed from below and cooled from above, which results in a convective motion. Convection cells occur when a fluid is heated from below if the temperature difference between the top and the bottom is sufficiently large. Atmospheric circulation is the large-scale movement of air and how thermal energy is distributed on the surface of the earth, together with the much slower (lagged) ocean circulation system. More localized phenomena than global atmospheric movement are also due to convection, including wind and some of the hydrologic cycle. Heat is the movement of energy from an object with high thermal energy to one with lower thermal energy. Convection is the process by which heat is transferred by the bulk moment of a fluid (gas or liquid).
Natural convection is the up-and-down movement of fluid due to temperature differences. Forced convection is when an external device, such as a fan, causes the circulation of warm or cool air. In the convection cycle, heat is absorbed from the source by a portion of the fluid that is being heated. The heated particles move to the colder region of the fluid. The heated particles share their heat with the colder particles. Colder regions fill the space vacated by this movement. In this way, the cycle is repeated.
A **Lorentz system** is a celebrated nonlinear dynamical dissipative system that Lorentz originally derived to study chaos in weather patterns.

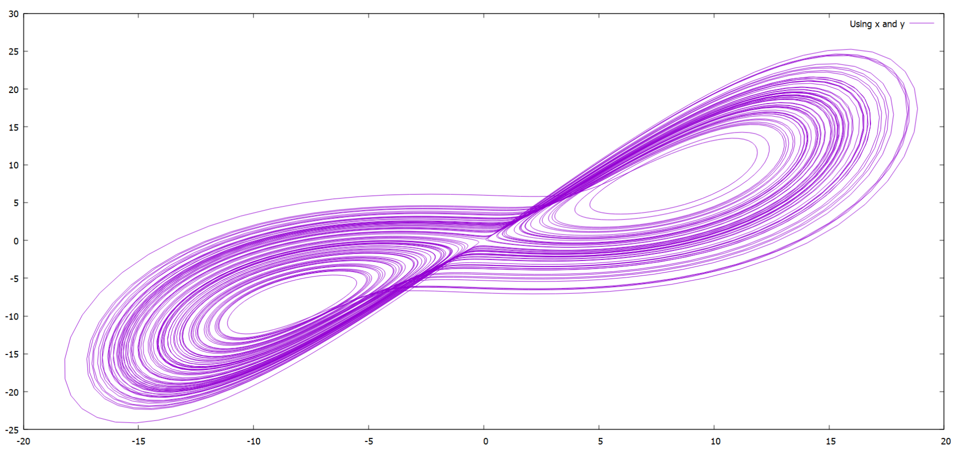

Figure 1: Lorentz Attractor

3 Important Definitions

- **Prandtl number**: It is defined as the ratio of viscosity to thermal diffusivity. It is a dimensionless number.
- **Rayleigh Number**: In fluid mechanics, the Rayleigh number (named after Lord Rayleigh) for a fluid is a dimensionless number associated with buoyancy-driven flow, also known as free (or natural) convection. It gives the temperature difference between the top and bottom of the slice (of atmosphere under consideration).

4 Parameters Required for the C++ Code

To simulate the convection in fluids using C++, we need the following parameters:

Parameters Input by User	**Symbol**
Prandtl Number	sig
Rayleigh Number	rho
Ratio of physical dimensions of layer	beta
Initial value of convective flow	x
Initial variation in horizontal temperature value	y
Initial variation in vertical temperature value	z
Step size	h
Number of steps	n

Table 1

5 Test Values

This problem is highly input-specific, so the values given as input by the user can't be random, as there is a set of certain values for which the chaotic behavior of the system can be properly seen.
The following is a set of **test values** for all parameters:

Parameter	**Test value**
sig	10
rho	28
beta	8/3
x	1
y	1
z	1
h	0.01
n	10,000

Table 2

The values listed above are not fixed, and the user is advised to change them to analyze how this change affects the system's behavior. For different test values, refer to [1].

6 Governing Equations: The Lorenz Equations

Imagine a rectangular slice of air heated from below and cooled from above by edges kept at constant temperatures. This is our atmosphere in its simplest description. The bottom is heated by the earth and the top is cooled by the void of outer space. Within this slice, warm air rises, and cool air sinks. The state of the atmosphere in this model can be described by three time-evolving variables:

$$\frac{dx}{dt} = \sigma(y - x) \tag{1}$$

$$\frac{dy}{dx} = x(\rho - z) - y \tag{2}$$

$$\frac{dz}{dt} = xy - \beta z \tag{3}$$

where
x = the convective flow
y = the horizontal temperature distribution
z = the vertical temperature distribution
σ = Prandtl number
ρ = Rayleigh number
β = the width-to-height ratio of the slice These equations also arise in studies of instability in planetary atmospheres, models of lasers and dynamos, etc.

7 Algorithm

1. **Include all the required header files.**
2. **Define a class 'lorenz'** which has all the above-listed parameters in Table 1.
3. **Declare public member functions** in the class that can be accessed from outside the class using an object or instance of the class defined in the main function (the compiler starts the execution of any C++ program from the main() function).
4. **Definition of Member Function 1:** void input ( ){//}
   The function initializes the parameters; Prandtl number, Rayleigh Number, ratio of physical dimensions of layer, initial values of the flow, variation in horizontal temperature, variation in temperature value, step size & number of steps.
   - Return-type: `void`
   - The function doesn't return any parameters.
   - The function doesn't accept any parameters.
5. **Definition of Member Function 2:** double function1(double ,double) {//}
   This function studies the evolution of the convection rate using eq. (1).
   - Return type: `double`
     The function returns the computed value of RHS of eq. (1).
   - The function accepts two parameters:
     - `x`: Initial value of convective flow.
     - `y`: Initial variation in horizontal temperature value.
6. **Definition of Member Function 3:** double function2 (double, double, double) {//}
   This function studies the evolution in horizontal temperature value using eq. (2)
   - Return type: `double`
     The function returns the computed value of RHS of eq. (2).
   - The function accepts three parameters:
     - `x`: Initial value of convective flow.
     - `y`: Initial variation in horizontal temperature value.
     - `z`: Initial variation in vertical temperature value.
7. **Definition of Member Function 4:** double function3 (double, double, double) {//}
   This function studies the evolution in vertical temperature value using eq. (3).
   - Return type: `double`
     The function returns the computed value of RHS of eq. (3).
   - The function accepts three parameters:
     - `x`: Initial value of convective flow.
     - `y`: Initial variation in horizontal temperature value.
     - `z`: Initial variation in vertical temperature value.

8. **Definition of Member Function 4:** void calc (void) {//}
   This function computes the convective flow, the horizontal temperature distribution, and the vertical temperature distribution over a period of time via the Euler Method (for more details, check out the documentation for the same) by solving the ordinary differential equations given in eq. (1), (2) and (3).
   - Return data-type: void This function doesn't return any value.
   - The function doesn't accept any parameter.
   - The function performs the following important computations required for the final result:
     - A **for-loop** is used to implement the Euler method for calculating the values of x, y, and z at different time steps. The loop gets terminated depending on the value of the total number of iterations.
       The function also makes the function calls to Member function 2, Member function 3, & Member function 4 and they return the required parameters to calc().
     - The function finally creates a 'lorenz.dat' data file and enters the data computed to this file, which includes values of x, y, and z.
9. **Define the main function:** `int main()`
   - Creates an instance/object of the class 'lorenz' and makes function calls to all necessary functions.
   - Return 0 to indicate successful execution.
10. The final step in the algorithm involves a crucial step i.e., plotting the data saved in the file by the calc() function using **Gnuplot software**.
    Gnuplot is a graphing utility for Linux, OS/2, MS Windows, OSX, VMS, and many other platforms. Its source code is freely distributed and is extensively used for data visualization.
    Follow the following steps to plot data on Gnuplot:
    (a) Download Gnuplot.
    (b) Go to → Change directory.
    Select the folder where the 'lorenz.dat' data file is stored.
    (c) Write the following command in Gnuplot:

```
plot "lorenz.dat" u 1:2 w l \\or
plot "lorenz.dat" u 2:3 w l
\* This command asks the Gnuplot to plot Column 2 with respect to Column 1
    with a line-type graph or plot Column 4 with respect to Column 3 with a
    line-type graph. There are many other variations to this command, you can
    check them out at the official website of Gnuplot or watch some free
    tutorials on YouTube
```

References

[1] @misc enwiki:1240615776, author = "Wikipedia contributors", title = "Lorenz system — Wikipedia, The Free Encyclopedia", year = "2024", "https://en.wikipedia.org/w/index.php?title=Lorenz_system&oldid=1240615776", note = "[Online; accessed 25-August-2024]"

[2] @misc enwiki:1237005729, author = "Wikipedia contributors", title = "Prandtl number — Wikipedia, The Free Encyclopedia", year = "2024", "https://en.wikipedia.org/w/index.php?title=Prandtl_number&oldid=1237005729", note = "[Online; accessed 25-August-2024]"

[3] @misc enwiki:1233266181, author = "Wikipedia contributors", title = "Rayleigh number — Wikipedia, The Free Encyclopedia", year = "2024", "https://en.wikipedia.org/w/index.php?title=Rayleigh_number&oldid=1233266181", note = "[Online; accessed 25-August-2024]"