TÉCNICO LISBOA

Aerodynamics III

Masters in Aerospace Engineering

Converging-Diverging Nozzle followed by an Adiabatic Pipe with variable Friction

---

Authors:
Eduardo De Almeida Helena Nº102793
Rafael Coimbra Azeiteiro Nº102478
Elena Francesca Cipriano Nº112514
João Pedro Gonçalves da Cruz Coelho de Almeida Nº103026
Professor:
José Manuel Da Silva Chaves Ribeiro Pereira

Group 3

Contents

1 Introduction

Once upon a time there was flow. This wasn't any particular flow, it was quite stationary, actually. Until it happened: the flow met a convergent-divergent nozzle. Our flow would never be the same again. It was taken from its cozy reservoir into a perilous adventure, through an adiabatic pipe with friction. What will be the fate of this flow? You decide! Presenting the newest Quasi 1D flow simulator. Will you accelerate the flow to supersonic speeds, or will you keep it calm and subsonic? Or will you force it to go through intense and dangerous shock waves? You control the action with the new Quasi 1D flow simulator!

This Matlab application was developed for the Aerodynamics III class and aims to analyze the behavior of a Quasi-1D flow in a convergent-divergent nozzle followed by an adiabatic pipe with friction.

The first approach was to create an iterative solver for subsonic-subsonic, subsonic-super sonic and shock cases and an analytical steady-state solver to compare the results. The file system of the differente solvers will be explicited in chapter 4.3.

2 Literature

All the equations used in this solver are derived from Modern Compressible Flow: With Historical Perspective by John D. Anderson (3rd Edition, McGraw-Hill, 2004) [1], specifically from Chapters 3, 4, 5, and 6. When referencing these equations in our explanation, we will explicitly cite their corresponding equation numbers from the book to maintain clarity and accuracy.

Some equations were also sourced from the Aerodynamics III class notes by José Manuel Da Silva Chaves Ribeiro Pereira [4] and [2] for friction relations. It will be explicitly present them where applicable.

3 First Solvers

3.1 Adiabatic Frictionless Nozzle Solver

First, an Adiabatic Frictionless Nozzle Solver was developed in order to understand the phenomena of the Quasi-1D compressible flow. In this solver are taken into account shocks and expansion fans which allowed to understand more deeply why a shock exists and where does it form.

Every case is taken into account: subsonic-subsonic, subsonic-subsonic chocked at the throat, subsonic-supersonic, normal shocks, oblique shocks and expansion fans. We filter the cases based on the user's inputs and get the graphs for: Nozzle geometry, pressure, Mach and temperature along the nozzle as seen in Figure 1.

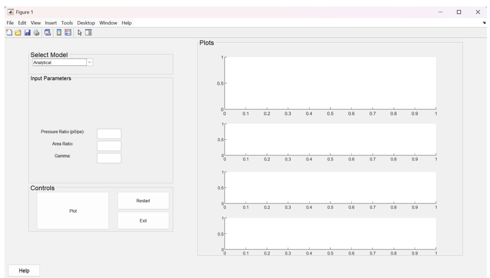


Figure 1: First Solvers GUI

Figure 2 shows the presence of oblique shocks, requiring by the solver the calculation of the shock angle β using Eq. 4.9 [1] and the deflection angle θ using Eq. 4.17 [1]. With this information, the oblique shock waves can be plotted outside of the nozzle.

In Figure 3, the green lines indicate expansion fans, specifically marking the limits. The black line adjacent to the green ones represents the flow direction after the expansion waves. To determine the precise angles, the Mach Angle was computed using Eq. 4.1 [1] for the flow before and after the expansion, while θ was obtained using the Prandtl-Meyer equations, Eqs. 4.44 and 4.45 [1].

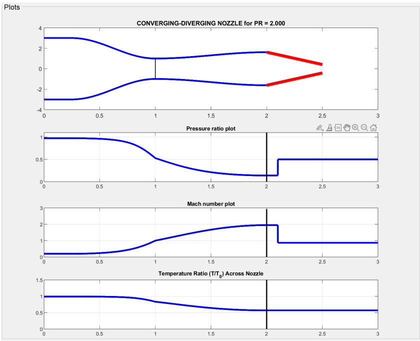


Figure 2: Oblique Shock

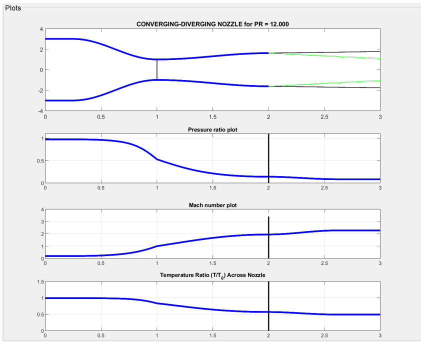


Figure 3: Expansion Fans

Developing this solver provided a deeper understanding of the flow evolution over the converging and diverging sections, as well as the formation, location, and reasoning behind shocks and expansion waves. Additionally, establishing a solid foundation in nozzle design was essential, particularly through the development of the nozzle drawing function, as will be discussed in Chapter 4.2.

3.2 Fanno Flow Nozzle

A limited solver was developed for a Fanno flow adiabatic pipe with variable friction. This analysis provided insights into the behavior of the flow inside a pipe and the effects of friction.

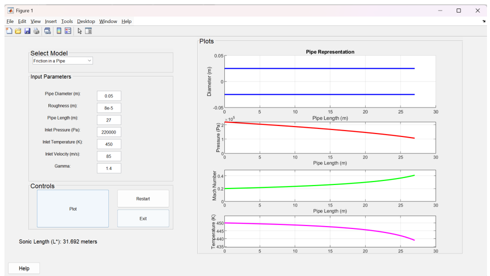


Figure 4: Pipe with Friction GUI

Values from an example in [2] were used as input, generating the plots shown in Figure 4. As illustrated in the same figure, the **sonic length** (L^*) represents the pipe length required for the flow to accelerate to Mach=1 while maintaining all other parameters constant. This parameter will play a crucial role in later analyses and decision on the main solver.

Exceeding the sonic length (L^*) leads to a constraint known as double choking, where the flow becomes constrained both at the throat and downstream. Resolving double choking requires the introduction of a new Fanno curve, allowing the flow to readjust to a modified mass flow rate and consequently altering the upstream conditions.

Another important concept explored in this solver is the distinction between a variable and a constant friction factor. The friction factor is used to calculate the pressure drop due to fluid flow in a pipe, representing the interaction between the fluid and the pipe walls.

There are different methods to determine the friction factor. One approach is graphical, using the Moody diagram 10A, but this method is impractical for automated calculations. Instead, empirical correlations are required. The correlation used in this solver is the Churchill Equation [2], which relates the Darcy friction factor (f) to the Reynolds number, diameter, and pipe roughness ($4Cf = f$).

As stated, the dependence on the Reynolds number means that friction is not constant at every point in the pipe. In this solver, variable friction is accounted by computing the Reynolds number at each discrete point along the nozzle, leading to key insights that influence major main solver decisions.

1. Computing variable friction requires higher computational power.
2. In low velocity flow variations, the effect of friction variation can be neglected (see Figure 10A).
3. In high-flow velocity variations, the effect of friction variation can make a relatively difference, as will be proven later in this report.

4 Main Solver - CD Nozzle followed by an Adiabatic Pipe with variable Friction

The `GUI_Main.m`, figure 5, function serves as the graphical user interface (GUI) for visualizing the problem in hands. The primary purpose of this GUI is to allow users to input relevant parameters, validate their entries, and visualize the results through graphical plots.

The GUI consists of several key components:

- **Input Panel:** Users can enter values for critical parameters, that will be best defined in chapter 4.1.
- **Control Panel:** This panel contains buttons to execute the solver (`Plot`), restart the GUI (`Restart`), and exit the program (`Exit`). A help button is also included to open a reference document.
- **Plot Panel:** The GUI provides visualizations of the results, displaying Mach number distribution, pressure variation, and temperature changes along the nozzle and pipe.

Each user input is handled through dedicated callback functions that assign values to global variables. The function also includes input validation to prevent unphysical or incorrect values:

- The stagnation pressure (p_0), stagnation temperature (T_0), and exit pressure (p_e) must be positive.
- The area ratio (A_e/A_t) must be greater than 1 to ensure proper nozzle operation.
- The exit pressure must be less than the stagnation pressure ($p_e < p_0$) to ensure that there is a flow and not in the wrong direction.

Once validated, the `Plot` function calls the main solver function, `GUI_Fanno_Flow_Nozzle2.m`, which performs the necessary calculations and updates the plots 4.1.

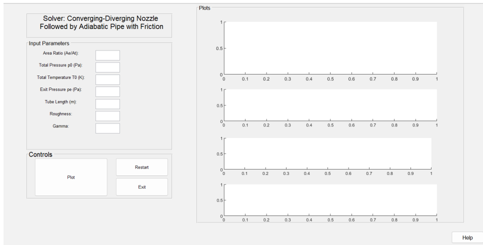


Figure 5: Main Solver GUI

4.1 Code Structure

The `GUI_Fanno_Flow_Nozzle2` function is responsible for solving the proposed problem, as stated in chapter 4. It integrates numerical computations and flow condition checks based on the required inputs.

The key inputs include:

- **Geometrical Parameters:** Area ratio (A_e/A_t), tube length (L).
- **Flow Properties:** Stagnation pressure (p_0), stagnation temperature (T_0), exit pressure (p_e), pipe roughness (ϵ), and heat capacity ratio (γ).
- **Graphical Handles:** Axes for plotting

The function initially determines the geometric characteristics of the nozzle based on the user-defined Area Ratio (AR) and Tube Length inputs. The tube diameter is implicitly defined by the specified Area Ratio, a concept that will be explored in greater detail in Chapter 4.2, as well as the details for the other functions mentioned.

Next, it determines the critical pressures to classify the type of flow 6. This is done using:

- critical_1: Computes the highest allowable pressure so that the flow is sonic at the throat and subsonic afterwards through the divergent and pipe sections.
- critical_2: Determines the outside pressure for a shock wave to occur at the pipe inlet, so that the flow is subsonic afterwards in the pipe.

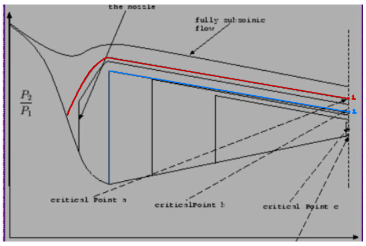


Figure 6: Critical 1 vs Critical 2

Based on these conditions, the function handles multiple flow scenarios, let's start with the ones that deal with subsonic flow in the pipe:

1. **Subsonic Flow:** If $p_e \geq$ critical 1, the function uses the `Subsonic.m` solver to compute the Mach number at the inlet of the pipe (exit of the nozzle) and at the exit of the pipe.
2. **Shock in the Nozzle:** If critical_2 $\leq p_e <$ critical_1, a normal shock occurs in the nozzle, and the function determines the shock position using `Shock_Div_Nozzle.m`.

Both these conditions deal when the user inputs the actual critical pressure.

Before addressing the cases where the pipe inlet is supersonic, two special conditions must be considered. As detailed later in the function explanations chapter 4.2, both the `Subsonic` and `Shock_Div_Nozzle` functions also compute the Mach number distribution along the pipe.

If the pipe length exceeds the critical length ($L > L_{\max}$ Eq.3.107 [1]) , the flow would reach sonic conditions before the outlet, requiring an adjustment to a new Fanno curve. This change would be impossible without new upstream conditions, such as a modified mass flow rate. However, to maintain consistency with the specified user inputs, such scenarios are disregarded in this solver.

Thus, we still compute critical_2 and critical_1, but under the assumption that Mach 1 (M = 1) occurs at the tube outlet (a flag is activated in the code). If the user inputs an exit pressure lower than this assumption, the system would experience double choking, rendering a **invalid solution**. In such cases, the user is asked to provide new input values.

Jumping now to the solutions where $p_e <$ critical_1 - in these cases the Tube Length is a major factor in the solver.

First to differentiate subsequent cases the code determines the inlet supersonic Mach number (M_{inlet}) based on the given area ratio (AR) Eq 5.20 [1], and then calculates the maximum allowable pipe length ($L_{\max}$) that allows the flow to reach Mach 1 ($M = 1$) at the pipe exit Eq 3.107 [1].

1. **Short tube:** If $L_{input} < L_{\max}$, the function calls the `L_shock.m` (finds shock location and Mach number before the shock), to analyze the supersonic flow evolution inside the tube. This can result in:

   - $p_e >$ critical_3: A normal shock forming within the pipe, leading to a transition from supersonic to subsonic flow.
   - critical_4 $< p_e <$ critical_3: Overexpanded flow with oblique shock outside.
   - $p_e =$ critical_4: No shock, flow is perfectly expanded - Design Operation Conditions.
   - $p_e <$ critical_4: Underexpanded flow with expansion waves outside.

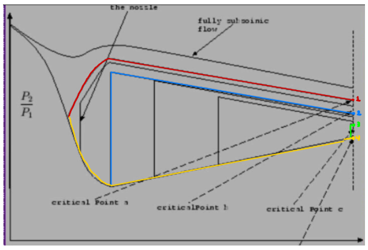


Figure 7: Critical cases with Supersonic Inlet in the Pipe

2. **Long Tube:** If $L_{\text{input}} > L_{\max}$, the function calls `L_sup.m` (finds shock location and Mach number before the shock). Since the flow at the pipe inlet is supersonic in this scenario, a shock wave must form within the pipe to prevent double choking. Thus, the function calculates a `critical_5` but under the assumptions that Mach = 1 perfectly occurs at the tube outlet, the flow decelerates from supersonic perfectly.

   The follow scenarios can happen:

   - $p_e >=$ `critical_5`: There is a normal shock inside the pipe leading to a transition and the exit flow is subsonic, or in the limit, the flow exits the pipe at M=1.
   - $p_e <$ `critical_5` If the downstream pressure is too low, the model predicts an **invalid solution** due to double choking, meaning a new Fanno line would be required to achieve a physically possible result. In this case, the function returns an error message prompting the user to adjust input parameters.

Once the type of solution is determined, the code computes the Mach number, temperature, and pressure along the nozzle and pipe using isentropic relations, Fanno flow properties, normal shock equations.

These values are iteratively calculated for each discretized nozzle position along the flow domain and are then assigned to the respective plot parameters to visualize the nozzle shape, Mach number distribution, temperature profile, and pressure variations. At each discretized point of the nozzle, the Churchill.m equation (referenced in 4.2) ensures the calculation of friction as a function of the Reynolds number and density. This is crucial for determining the Fanno flow properties - method for accounting variable friction.

4.2 Functions

- In the `NozzleDraw.m` function, the variable "Parameter", that varies between 0 or 1 determines whether the software draws a nozzle only (0) or a nozzle with a pipe (1).

  The first 1/8 of the nozzle is part of the Reservoir, and therefore is just a straight tube. The nozzle is then defined by two cosine functions that meet at the throat, set at 1/2 of the length of the nozzle. The radius of the throat is set to 1/3 of the radius of the Reservoir.

  If the case of a Nozzle only, the radius of the Reservoir is set to 10. In the case of Nozzle plus pipe, the Reservoir radius is set to 3, so when the user chooses the area ratio, they are in fact choosing the pipe radius. The length of the pipe is also defined by the user.

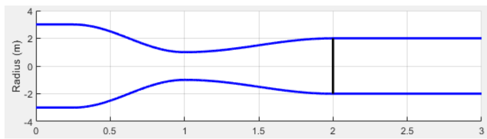


Figure 8: Nozzle with an AR of 2 and a pipe with a length of $1m$

- In the `Churchill.m` function calculates the friction at a selected point given the pipe roughness and diameter and pressure, temperature and Mach number at that point. It starts by computing the dynamic viscosity using the Sutherland equation, the density with the perfect gas equation and the speed of sound. Then it solves the following equations [2], [3]:

$$Re = \frac{\rho ARV}{\mu} \quad (1)$$

$$A = (-2.457log[(\frac{7}{Re})^{0.9} + 0.27\frac{\epsilon}{AR}]^{16} \quad (2)$$

$$B = (\frac{37530}{Re})^{16} \quad (3)$$

$$f = 8[(\frac{8}{Re})^{12} + (A + B)^{-1.5}]^{\frac{1}{12}} \quad (4)$$

- The `critical_1.m` function calculates the critical exit pressure (pc_1), assuming sonic conditions at the nozzle throat, and evaluates whether the flow at the pipe exit is subsonic or choked. Based on the area ratio (AR), it determines the Mach number (M_1) at the nozzle exit (pipe inlet) using Eq. 5.20 [1]. Then with Eq 3.28 and 3.29 [1], it calculates the temperature and pressure at this point. This way, it can call the `Churchill.m` function to calculate the friction factor at this point. With that, it can use Eq 3.107 [1] to calculate Mach number at the exit. Then, applying the Fanno and isentropic relations (Eq. 3.98, 3.102, 3.30 [1]) it determines pc_1 . Note that, since the pipe length is defined by the user, doubly choked flow may occur. If $Me > 1$, choked flow is assumed, and pc_1 is calculated considering sonic conditions at the pipe exit, as the goal is to find the minimum allowable static pressure at the exit that keeps the flow subsonic (and sonic) throughout the nozzle and pipe.

- The `critical_2.m` function calculates the critical external pressure (pc_2) that would cause a shock wave to form at the pipe inlet. It assumes sonic flow at the nozzle throat and, based on the specified area ratio (AR), computes the supersonic Mach number before the shock (Eq. 5.20 [1]). From this, the Mach number after the shock (M_1, i.e., at the pipe inlet) is determined (Eq. 3.51 [1]). Using Fanno flow equations [4], the function then evaluates the evolution of the flow along the pipe and computes the Mach number at the pipe exit (Me). It is important to mention that the friction factor used is calculated for the beginning of the pipe. If the flow remains subsonic in the pipe—or becomes exactly sonic at the exit—pc_2 is calculated using the corresponding Fanno relations (Eq. 3.64, 3.98, 3.102, 3.30 [1]). However, if the flow becomes doubly choked (i.e., $Me > 1$), $flag_2$ is activated, and pc_2 is instead computed assuming a choked exit ($Me = 1$), providing a lower bound for the critical pressure.

- The `Subsonic.m` function computes M_1, the Mach number at the inlet of the pipe and M_e, the Mach number at the exit of the pipe for a subsonic flow through a converging-diverging nozzle followed by a pipe with friction, solving isentropic and Fanno relations. It starts by using a guess for the values of M_e and M_1 and attributing the value of p_0 that was given to p_{0e}. Then, the MATLAB's fsolve function is used to solve the system of nonlinear equations defined in the `SubsonicEquations` function. The first equation, relates the stagnation pressure ratio (p_{0e}/p_0) to the Mach numbers (Eq. 3.102 [1]), the second equation relates the exit stagnation-to-static pressure ratio to the exit Mach number (Eq. 3.30 [1]), and the third equation applies the Fanno flow relation for pipe flow with friction (Eq. 3.107 [1]), the friction being calculated with the `Churchill.m` function for both the inlet and outlet of the pipe. The `Subsonic` function then checks if the `SubsonicEquations` function converges, asking for different inputs, if it doesn't. Then, after extracting M_1 and M_e, it checks if the flow gets chocked, returning 1 if the Area ratio is higher than the critical area and 0 if it isn't. It is important to keep in mind that, since the throat area is 1, the Area ratio has the same value as the exit area.

- The `Shock_Div_Nozzle.m` function determines the position of a normal shock wave inside a converging-diverging nozzle, specifically when the exit pressure (p_e) lies between two critical values, pc_1 and pc_2. In this condition, the flow becomes supersonic in the divergent section of the nozzle but must transition to subsonic via a shock wave within the nozzle to match the downstream pressure. The function starts by computing the critical mass flow rate (Eq. 5.21 [1]), assuming sonic flow at the throat. This value, constant throughout the system, serves as a constraint for subsequent calculations. It then estimates the exit Mach number (M_e) by solving an equation that matches the critical mass flow rate with the flow conditions at the pipe exit (Eq. 2.1 in [1]).

  Once Me is found, the corresponding stagnation pressure at the pipe exit is calculated using isentropic relations (Eq. 3.30 [1]). With this, all the values necessary to calculate the friction factor at the exit have been obtained. Then the Fanno flow equations ([4]) can be used to compute the Mach number at the pipe inlet (M_1) and the stagnation pressure before the shock (p_{0_1}).

  Given the stagnation pressures before (p_0) and after (p_{0_1}) the shock, the Mach number before the shock (M_A) is obtained using relations from the course slides, followed by the Mach number after the shock (M_B) using normal shock equations (Eq. 3.51 [1]). Finally, the function determines the shock position by calculating the area required to sustain Mach M_A and locating where this area matches the actual nozzle geometry.

- The function `L_shock.m` determines the location of a shock, either inside the pipe or at the exit. It starts by computing the critical mass flow rate, with Eq 5.21 [1]. It then uses the Matlab function fzero to solve Eq 5.20, 3.107 and 2.1 [1] for supersonic Mach number at tube inlet (M_1), for Mach number before the shock (M_A) and to compute static pressure before the shock (p_A), respectively. To solve Eq 3.107 [1] and calculate M_A, the friction factor at the tube inlet. It then uses Eq 3.57 [1] to compute the critical exit pressure for a shock at the outlet ($p_{e_{sk}}$). This way it can compare the exit pressure, p_e to it: if the exit pressure is lower or equal to the critical pressure, then it knows that the shock occurs at the exit of the pipe.

  If this is true, it can determine the type of shock at the exit: a normal shock at the outlet, an overexpanded flow with an oblique shock, a perfectly expanded flow with no shock, or finally an underexpanded flow with expansion waves. All these cases are properly identified with the "flag" variable, ranging from 1 to 4, respectively, and the process to diferentiate between cases has been outlined in section 4.1.

  If the exit pressure is higher than the critical pressure, it knows that the shock occurs inside the tube. It then computes the Mach number at the outlet of the tube with Eq 2.1 [1] and calculates the shock location and the Mach number after the shock. It does this by using fsolve to solve (no way) the system of equations outlined in the function **M_shock**. This function contains the Fanno flow equation up to the shock location and after the shock, Eq 3.107 [1], and the normal shock relation, Eq 3.51 [1]. It also calls the `Churchill.m` function to calculate the friction factor at the tube inlet and outlet to solve Eq 3.107. With these equations solved it then calculates the shock position and the Mach number after the shock.

- The `L_sup.m` function calculates the shock wave location and post-shock Mach number for supersonic flow through a long tube ($L_{input} > L_{max}$), assuming the inlet flow is supersonic. First, it computes the critical mass flow rate (Eq. 5.21 [1]), which is the maximum mass flow rate possible for the given inlet conditions. The inlet Mach number ($M1$) is found iteratively by solving Eq. 5.20 [1] within a Mach number range above 1. The function then calculates the stagnation pressure and the outlet pressure for sonic flow at the exit of the pipe (Eq. 5.21 and 3.30 [1]). This provides a reference to compare with the given exit pressure (p_e). If $pe_{crit_{lmax}} <= p_e$, the flow is subsonic or sonic at the exit, and the function solves Eq. 2.1 [1] for the outlet Mach number (Me), followed by an iterative solution of the `M_shock` function to find the shock location, as explained in `L_shock.m`. If $p_e < pe_{crit_{lmax}}$, no solution exists, as this would indicate a doubly choked flow.

4.3 User Manual

Upon delivery, the user will find two directories: `First Solvers` and `Main Solver`. The `First Solvers` directory contains solvers discussed in Chapter 3, while the `Main Solver` directory includes the solver detailed in Chapter 4 - CD Nozzle followed by an adiabatic pipe with variable friction.

In both directories, the function `GUI_Main` serves as the entry point for execution. Running this function launches the graphical user interface (GUI), allowing users to interactively input parameters and execute the corresponding solvers.

Usage Instructions:

- Enter the required input parameters in the provided text fields.
- Click the **Plot** button to compute and display results.
- Click **Restart** to reset inputs and restart the GUI.
- Click **Exit** to close the program.
- Click **Help** to open the help file.

If an error occurs related to fzero() or fsolve(), ensure that the MATLAB Optimization Toolbox is installed and activated, as these functions require it. It is possible to check the install via Command Window: $ver('optim')$.

If not installed, go to MATLAB Add-Ons - Get Add-Ons - Optimization Toolbox, or use: $matlab.addons.install('Optimization\_Toolbox')$.

5 Case Studies

To validate the code, a set of representative test cases were selected, aiming to cover all the main flow scenarios, and is presented in this report. The simulations were carried out by varying two key parameters: the exit pressure and the pipe length. As discussed in 4.1, varying the $TubeLength$ parameter allowed the classification of cases into Short Tubes ($L_{\text{input}} < L_{\text{max}}$) and Long Tubes ($L_{\text{input}} > L_{\text{max}}$). For each of these two configurations, multiple exit pressure conditions were analyzed.
Note that all plots showing the distribution of pressure, Mach number, and temperature along the nozzle and pipe, together with the respective input parameters, are provided in B.

Short Pipe - $L_{input} \leq L_{max}$

- **Case 1** ($p_e \geq$ `critical_1`) : The flow remains subsonic throughout the nozzle and the pipe, or at most becomes choked at the nozzle throat when $p\_e =$ `critical_1`. In the pipe, the Fanno flow progresses toward sonic conditions, which are reached only when $L_{input} = L_{\text{max}}$. It is worth noting that the variation of the Mach number along the pipe for this case is almost negligible. This is because, as seen in Fig. 11 , the required L^* for a subsonic flow is very large, unlike the case with a supersonic inlet, where much shorter pipe lengths are needed to reach sonic conditions.

Friction Study - Case 1

Fig. 18b shows the variation of the friction factor along the pipe. As expected, in the case of a fully subsonic flow, the friction factor gradually decreases along the pipe. In this particular scenario, the decrease of friction factor between the inlet and the outlet of the pipe is on the order of 10^{-11}. Additionally, the friction factor is also plotted as a function of the Reynolds number (Fig. 18a), which increases along the pipe, to generate a graph resembling the Moody diagram. When compared with the actual Moody diagram, the expected behavior is confirmed: for Reynolds numbers around 10^8, the friction factor remains nearly constant. For a roughness ratio of $\varepsilon/D = 0.05$, the friction factor value is approximately 0.07, as anticipated.

- **Case 2** (`critical_2` $< p_e <$ `critical_1`): With sonic conditions fixed at the throat, a normal shock occurs in the divergent section of the nozzle, shifting toward the nozzle exit as p_e decreases within the given range. Downstream of the shock, the flow becomes subsonic and remains subsonic along the pipe, potentially reaching sonic conditions only at the outlet.

- **Case 2.1** ($p_e =$ `critical_2`): As a more specific case within the pressure range of Case 2, this p_e shows, as expected, the shock wave positioned at the nozzle exit.

- **Case 3** (`critical_3` $\leq p_e <$ `critical_2`): From this point onward, the flow entering the pipe is supersonic. As the external pressure decreases, a normal shock wave forms inside the pipe and progressively moves toward its outlet. When the outlet pressure reaches $p_e =$ `critical_3`, the shock is positioned precisely at the pipe exit. In all these cases, the flow exits the pipe in a subsonic state, or sonic at most.

- **Case 4** (`critical_4` $< p_e <$ `critical_3`): within this range of p_e, the flow remains supersonic throughout the pipe and exits in an overexpanded condition. To adjust to the external pressure, oblique shock waves form at the pipe outlet.

- **Case 5** ($p_e =$ `critical_4`): the exit pressure perfectly matches the pipe outlet pressure, allowing the flow to exit without the formation of a shock.

Friction Study - Case 3

As in Case 1, Fig.19b shows how the friction factor changes along the pipe. Because of the shock wave, there is a sudden jump in the friction factor, with a small increase of about 0.00028% exactly at the shock location. Before the shock, the flow is supersonic and slows down, causing the friction factor to increase. After the shock, the flow becomes subsonic, and the friction factor starts to decrease again as the flow moves toward sonic conditions, but the change is more gradual. This behavior is attributed to the fact that frictional effects are significantly stronger at higher Mach numbers. In Fig.19a, for a roughness ratio of $\varepsilon/D = 0.05$ and Reynold number around 10^8 the friction factor remains nearly constant.
A detailed view of this plot is shown in Fig.19c where a small loop can be seen. This happens because the shock causes the Reynolds number to have the same value in two different parts of the pipe—in correspondence of the shock and after the shock—creating this curved shape in the graph.

Friction Study - Case 5

In Fig. 20b, the variation of the friction factor along the pipe follows the expected trend. As the Mach number decreases from supersonic toward sonic conditions, the friction factor increases accordingly. The overall increase in the friction factor along the pipe is approximately 0.000014%, which, although lower than in the case involving a shock wave, is still greater than that observed in fully subsonic flow.

Additionally, in the friction factor versus Reynolds number plot (Fig. 20a), for a fixed roughness ratio and Reynolds numbers on the order of 10^8, the friction factor remains nearly constant, consistent with the behavior described by the Moody diagram.

- **Case 6** ($p_e <$ `critical_4`): Within this pressure range, the flow follows the $p_e =$ `critical_4` curve until the outlet of the pipe, where the underexpanded flow adjusts to the external conditions through expansion waves.

Long Pipe - $L_{input} > L_{max}$

When $L_{input} = L_{max}$, the flow is choked, meaning it's not possible to increase the pipe length any further without changing the inlet conditions. If the flow enters subsonically, a longer pipe would require a lower inlet Mach number, which means shifting the inlet conditions to the left on the Fanno curve. As a result, for this situation, the equivalent of Case 1 and Case 2 (including Case 2.1) does not provide any meaningful outcome.
On the other hand, if the inlet flow is supersonic and L_{input} exceeds the length required to reach Mach 1 at the exit, a normal shock will form inside the pipe, after which the flow becomes subsonic and act differently according to the outside pressure:

- **Case 7** ($p_e \geq$ `critical_5`): after the shockwave the flow stays subsonic throughout the remaining length, or at most, sonic at the outlet when $p_e =$ `critical_5`
- **Case 8** ($p_e <$ `critical_5`): based on the definition of `critical_5`, this scenario results in a doubly choked flow, which the solver cannot process. It suggests adjusting the inputs (ex. lowering the pipe length or increasing its diameter).

For each case, the computational results have matched the expectations. The shockwaves and expansion waves appear exactly when expected.

Figure 9: Time to take a break [4]

References

[1] John D. Anderson. *Modern Compressible Flow: With Historical Perspective*. McGraw-Hill, New York, third edition, 2004.

[2] John M. Cimbala. Fanno flow – compressible duct flow with friction, 2024. Accessed: 2025-03-11.

[3] COMSOL Multiphysics. High mach number flow modeling, 2020. COMSOL Multiphysics Documentation, Version 5.5.

[4] José Manuel Da Silva Chaves Ribeiro Pereira. "Class Notes on Aerodynamics III". Lecture notes, Instituto Superior Técnico, Universidade de Lisboa, 2024.

Appendices

A First appendix

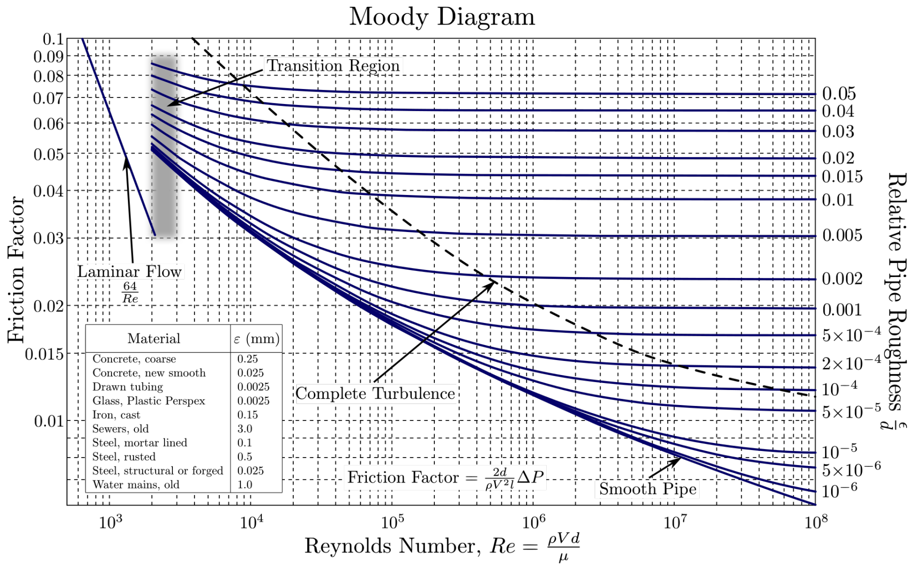


Figure 10: Moody Diagram

B Second appendix

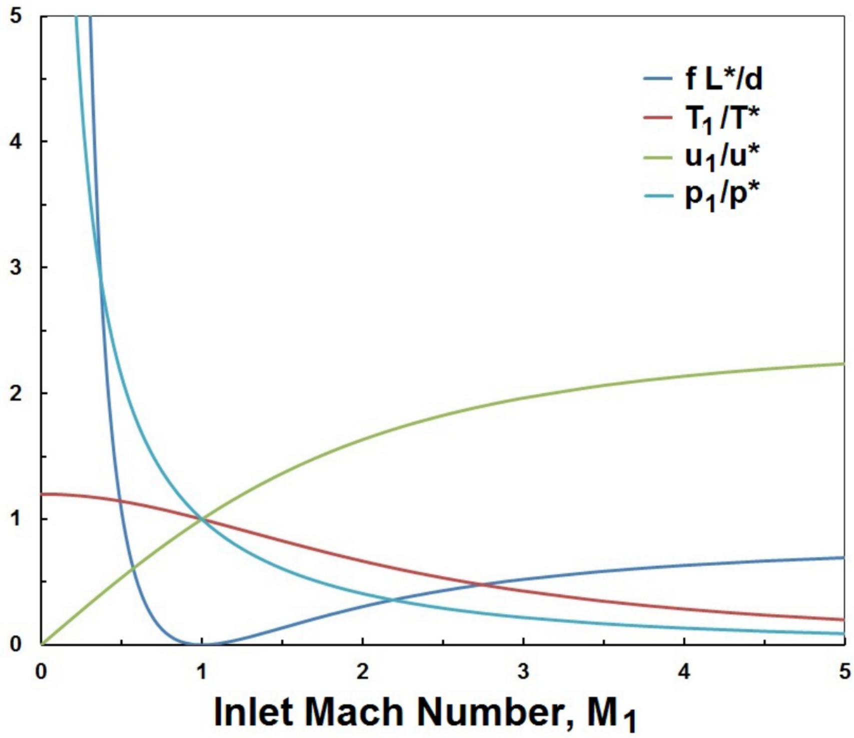
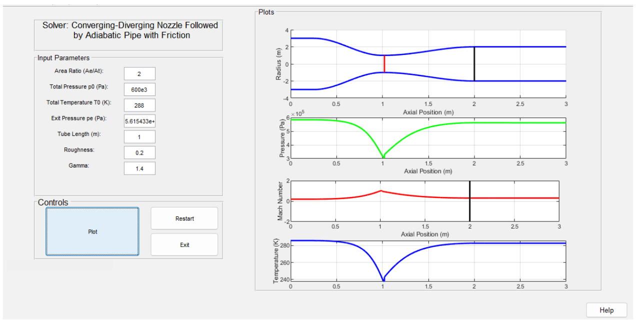
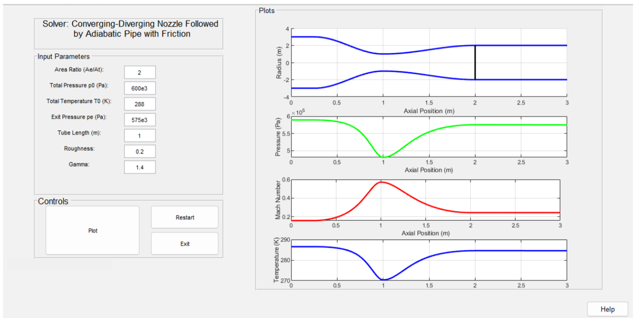


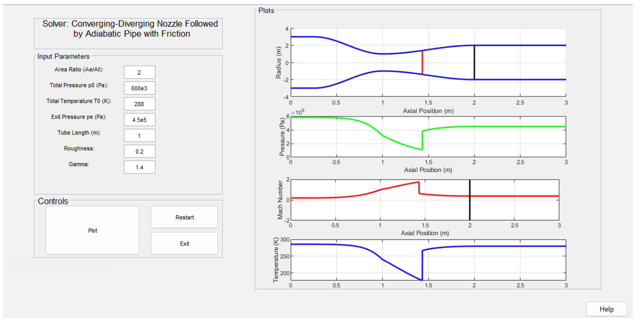
Figure 11: Graphs of $fL_{max}/AR, T_e/T^*, u_e/u^*, p_e/p^*$



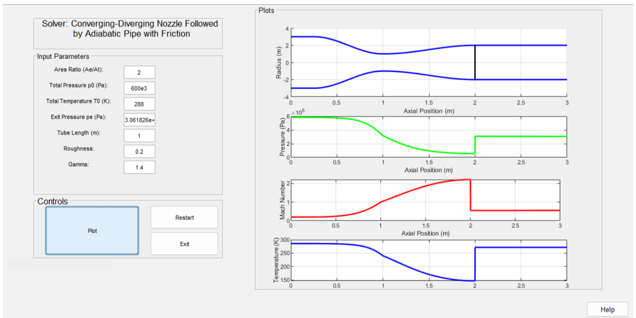
(a) Case 1 ($p_e = p_{c1}$)



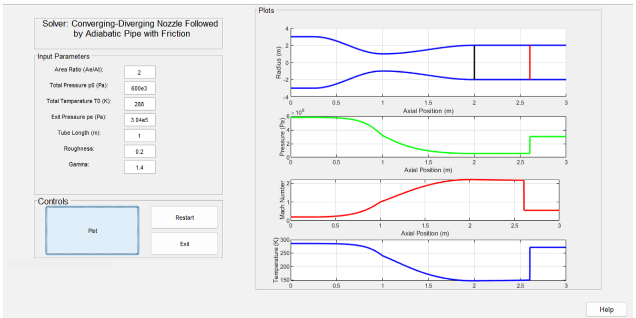
(b) Case 1 (Subsonic - Subsonic)



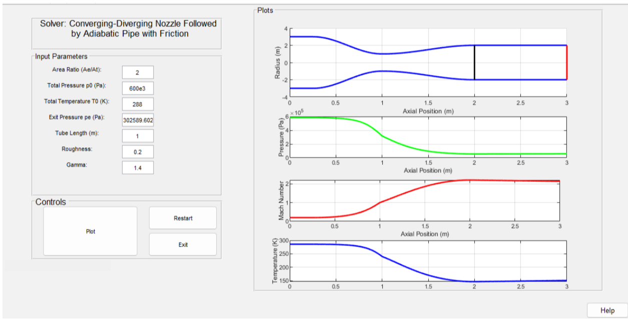
(a) Case 2 (Shock in the Nozzle)



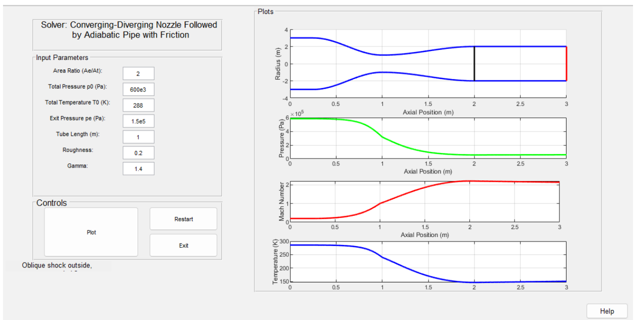
(b) Case 2.1



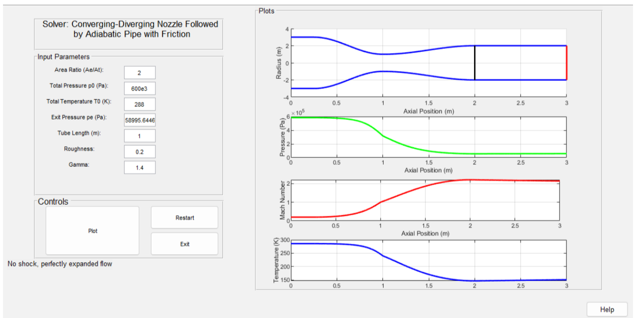
(a) Case 3 ($p_{c3} < p_e < p_{c2}$) (Pipe Shock)



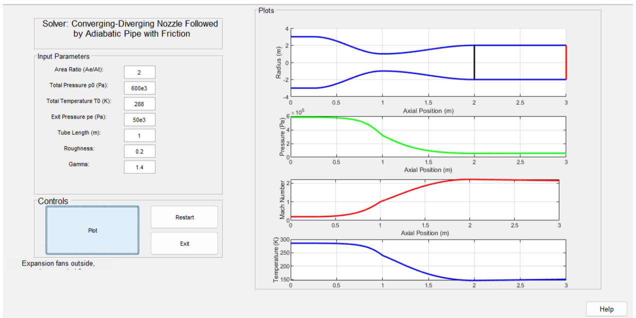
(b) Case 3 ($p_e = p_{c3}$)



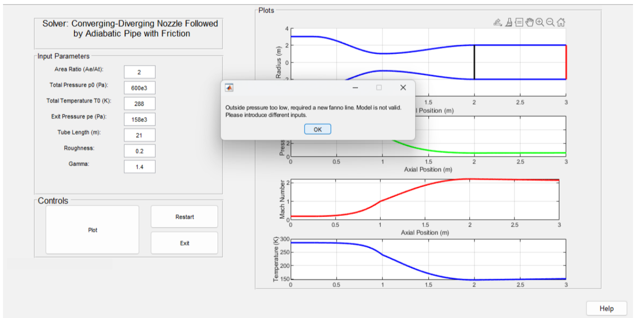
(a) Case 4 $p_{c4} < p < p_{c3}$



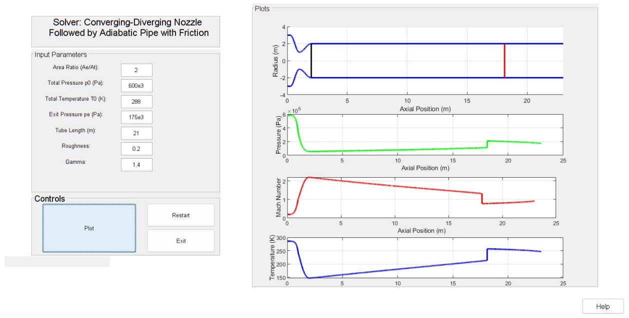
(b) Case 5 ($p = p_{c4}$)



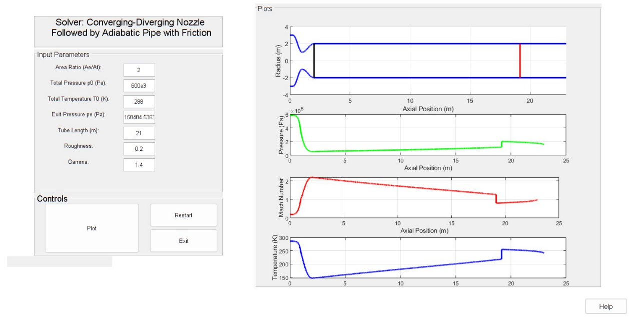
(a) Case 6



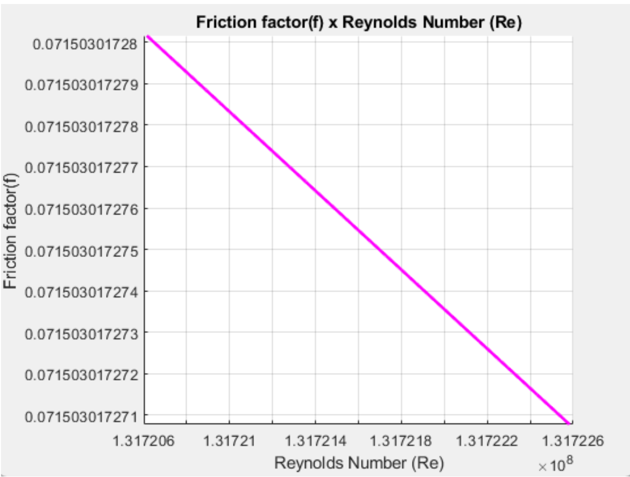
(b) Case 8



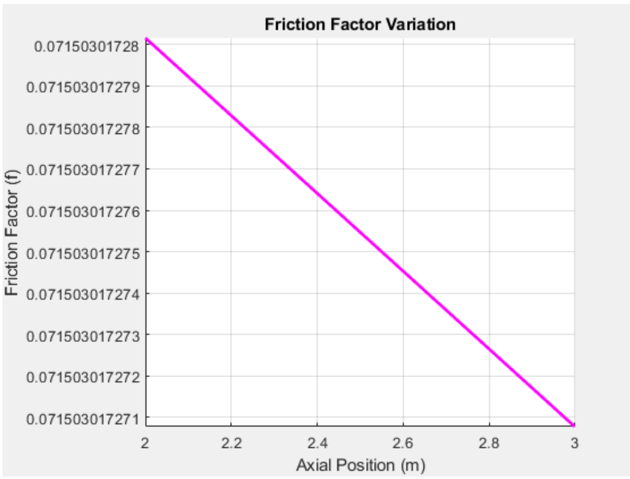
(a) Case 7 ($p_e > p_{c5}$)



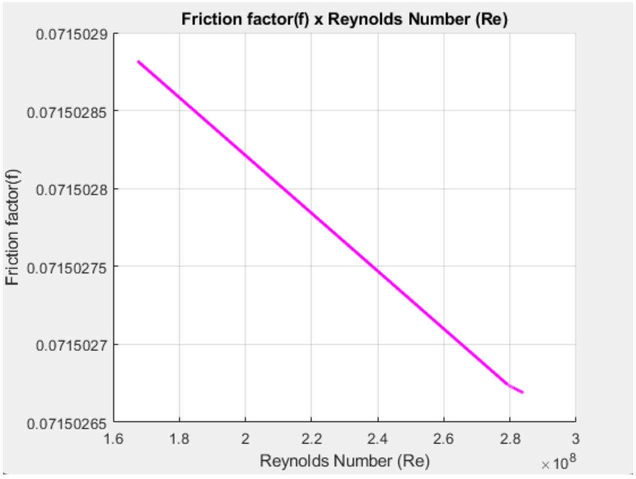
(b) Case 7 ($p_e = p_{c5}$)



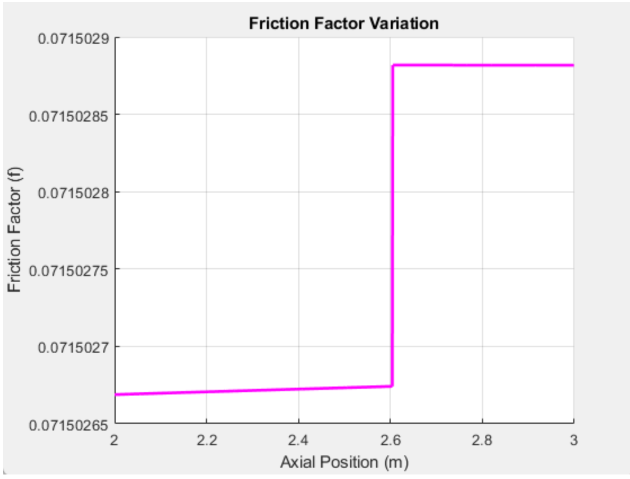
(a) Case 1 Friction vs Reynolds (Sub-Sub)



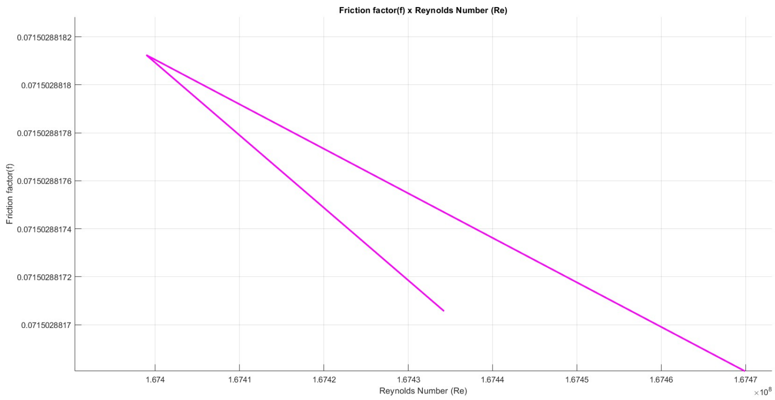
(b) Case 1 Friction along the pipe (Sub-Sub)



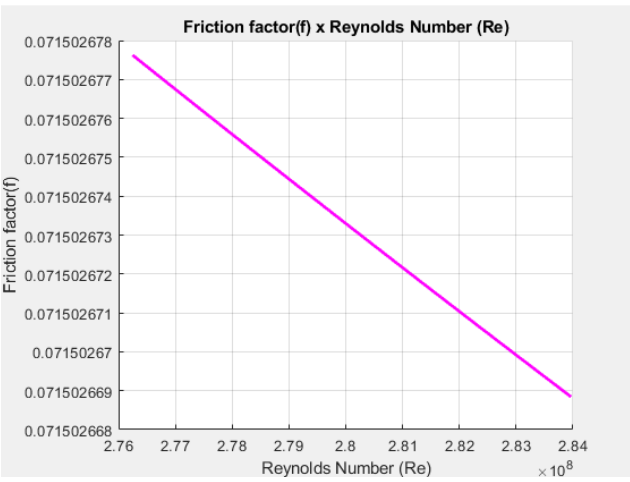
(a) Case 3 Friction vs Reynolds (Shock)



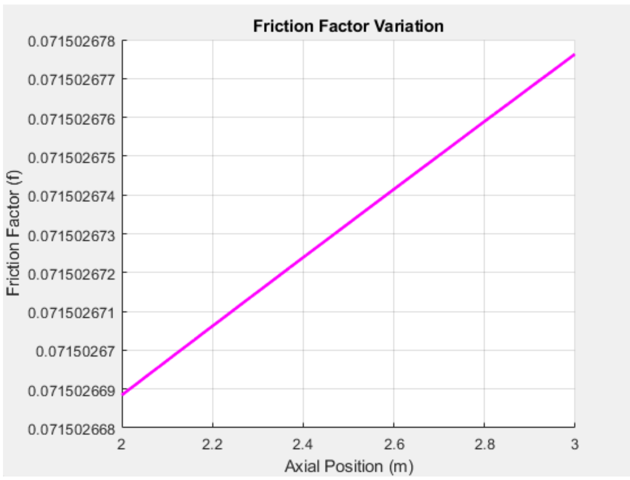
(b) Case 3 Friction along the pipe (Shock)



(c) Case 3 close-up



(a) Case 5 Friction vs Reynolds



(b) Case 5 Friction along the pipe