University of Cambridge

Department of Engineering

Progress Report 0

Ekrem Ekici

PhD in Engineering

Supervisor: Prof Matthew Juniper

January 4, 2021

Contents

1 Motivation

Over the last five decades the requirement for supersonic and hypersonic propulsion has grown considerably. Prediction of thermoacoustic oscillations gained importance in order to run rocket engines and gas turbines properly. This motivation requires to being capable of solving complicated mathematical models by utilizing adjoint methods.

2 Mathematical Model

1D thermoacoustic model has been considered. Mean flow has been set to zero. As can be seen in Figure 1, open ended elongated tube from 0 to X containing gas at following parameters;

- Uniform density, $\overline{\rho}$
- Uniform pressure, $\overline{p}$
- Ratio of specific heats, γ
- Velocity, u
- Pressure, p

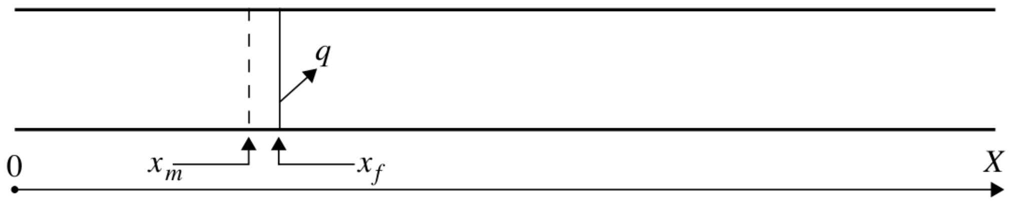

Figure 1: Diagram of the open-ended tube extending from 0 (upstream) to X (downstream). The heat release, q, occurs at position x f and is a function of the acoustic velocity at x m . The mean flow, which is defined as positive in the positive x-direction, is set to zero.

2.1 Governing Equations

The 1D acoustic energy and momentum equations are;

$$\bar{\rho}\frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} = 0 \tag{2.1}$$

$$\frac{\partial p}{\partial t} + \gamma\bar{p}\frac{\partial u}{\partial x} = (\gamma - 1)q\delta_D(x - x_f) \tag{2.2}$$

Heat release rate $q(t) = nu(x_m, t_\tau)[W/m^2]$ is placed at $x = x_f$, where $n[J/m^3]$ is a real constant, τ is time delay, x_m is the measurement position of u. Mean density drop across the heat source, viscous and thermal dissipations have been neglected.

δ is the Dirac delta. Reference length, reference speed and reference pressure are defined as $L_{ref} = X$, $U_{ref} = \bar{c}$ and $P_{ref} = \bar{p}$, respectively.Dimensionless acoustic energy and momentum equations are stated as follows;

$$\gamma\frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} = 0 \tag{2.3}$$

$$\frac{\partial p}{\partial t} + \gamma\frac{\partial u}{\partial x} = (\gamma - 1)q\delta_D(x - x_f) \tag{2.4}$$

3 Numerical Methods

Solution of fluid flow domain has a complex physical and mathematical examinations. 4 methods are commonly preferred to approximate exact solutions of governing equations. These methods are;

- Traveling Wave Method
- Helmholtz Finite Difference Method
- Helmholtz Finite Element Method
- Galerkin Method

Each method has been used to solve thermoacoustic governing equations mentioned in previous section. Equation 2.3 and 2.4 are determined by applying these methods.

4 Codework

In order to solve the thermoacoustic governing equations, these 4 method could be programmed to get parametric and quick results.

JULIA programming language is preferred to perform codeworks.

4.1 Why Julia Programming Language?

Julia programming language is developed in order to perform matrix calculations reasonably fast and easy. It is completely open source programming language which enables to work without getting licence. Additionally, having high-level programming syntax makes it easy to learn and its computing performance is blazing fast comparing to programming languages like MATLAB and Python. Furthermore, dynamic type syntax is not requiring to define parameter type(*int, float, double, string, char*) such as fast programming languages C++ and Fortran.

4.2 Programming Methodology

During the code development, mathematical model is created based on object oriented programming technique. Followed methodology is illustrated in Figure 2.

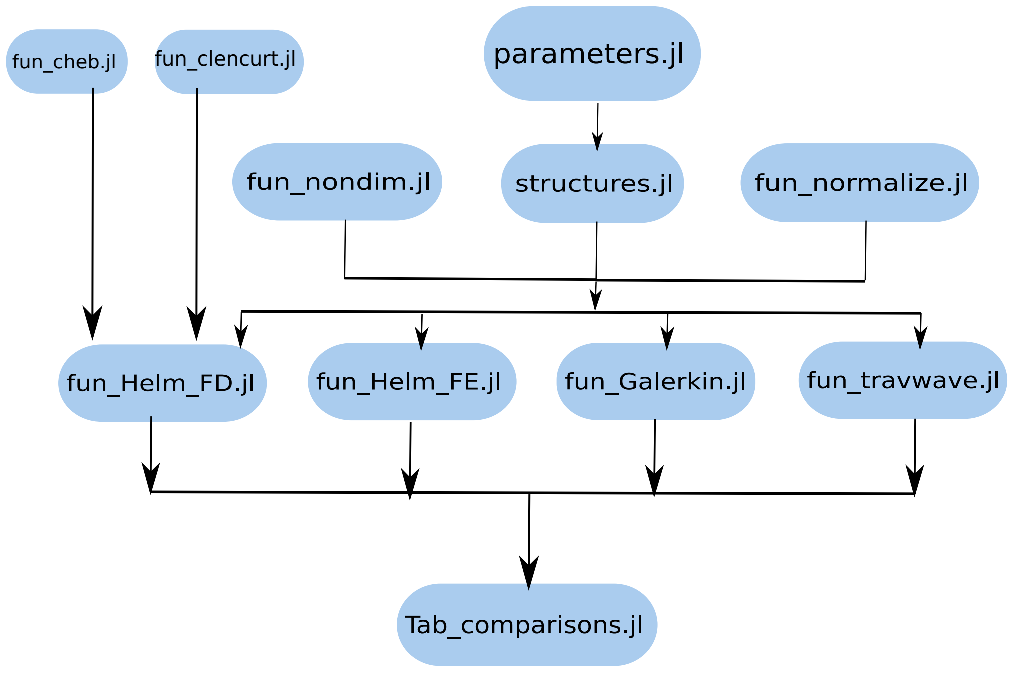

Figure 2: Codework layout and proposed flowchart

Figure 2 shows that each Julia file is connected by following systematic approach. By using these codes, all the methods' calculations can be performed and obtained results and table can be generated in file *Tab_ comparisons.jl*. Script *structures.jl* includes structures of **emode** and **ds**. These 2 structures are determined and returned by 4 main scripts;*fun_ Helm_ FD.jl*, *fun_ Helm_ FE.jl*, *fun_ Galerkin.jl and fun_ travwave.jl* Specified model parameters are inserted into *parameters.jl* file. Moreover, normalization and non-dimensionalization are performed by *fun_ normalize.jl* and *fun_ nondim.jl* files respectively.

5 Results

Table 1 shows the eigenvalues, s, determined according to four different methods for N=400. Figure 3 shows the corresponding P (x) and U (x) eigenvectors for the highest resolution case of each method. The eigenvectors differ only around the heat release zone, at $x = x_f = 0.25$. Main difference coming out in Galerkin method, rest of the three method showed good agreement.

Table 1: Eigenvalues, s, calculated with four different methods and, where relevant, up to five different resolutions, N , for the same thermoacoustic system. The growth rate is s r and the frequency is s i . The solutions approach each other as the resolution increases. (Tab comparisons.jl)

N	fun_travwave	fun_Helm_FD	fun_Helm_FE	fun_Galerkin
1	0.121875308592578883 + 3.2227814041928715im	-	-	-
10	-	0.12301472009722814 + 3.2222551623591285im	1.2649666277747205e-10 + 3.141592631034653im	3.42049450595866e-21 + 3.1545273778453238im
40	-	0.12363155192096682 + 3.221964689943754im	0.06990252109480179 + 3.184618956384646im	0.1045690701269305 + 3.2100253033952892im
100	-	0.12115229273097906 + 3.2231040481839406im	0.12302841468171331 + 3.2235632383934383im	0.12181433836201444 + 3.2228598981240886im
400	-	0.12205739886178892 + 3.2226989128574868im	0.12181867950887351 + 3.2227425705073545im	0.12181840527018106 + 3.2227499047678436im

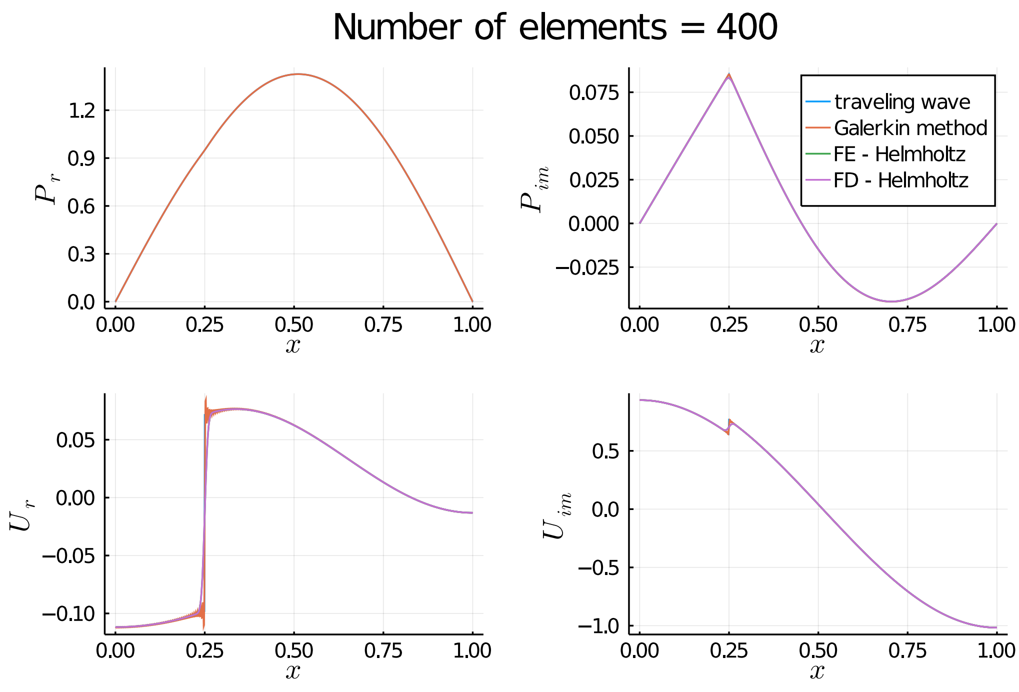

Figure 3: Real and imaginary components of the pressure, P (x), and velocity, U (x), eigenfunctions calculated for the highest resolution cases of the methods shown in table 1. (Tab_comparisons.jl)

Table 2 tabulates the calculated base sensitivities of each method.

Table 2: Base state sensitivities calculated by means of four methods with N = 400 (Tab comparisons.jl)

	$\partial s/\partial n$	$\partial s/\partial \tau$	$\partial s/\partial x_m$	$\partial s/\partial x_f$
fun_travwave.jl	0.10298315656272834 + 0.08269441085108045im	0.2539549055356268 - 0.34197060875680085im	-0.2664856896104548 - 0.17766984877202488im	0.43129016791308405 + 0.21134136333692508im
fun_Helm_FD.jl	0.1029381172495191 + 0.08265540766612185im	0.2538373154537853 - 0.341812025204056im	-0.2663486625117436 - 0.17759310185260918im	0.4310431667740005 + 0.2112708145173602im
fun_Helm_FE.jl	0.10293804348656137 + 0.08265471643133453im	0.2538357312085307 - 0.34181243558702773im	-0.2663469127817901 - 0.17759196026029755im	0.4310398326117469 + 0.21126935565722768im
fun_Galerkin.jl	0.1033169131009024 + 0.08257175330986924im	0.253493305952927 - 0.3430377969585296im	-0.26685205730784367 - 0.17749521966955883im	0.398800858731003 + 0.2259008693655126im