$\text{OSCILOS}_{\text{lite}}$ User Guide

Renaud Gaudron[1], Alexander MacLaren, Aimee S. Morgans

Department of Mechanical Engineering, Imperial College London, UK

Version 2.0, 18/01/2021

$\text{OSCILOS}_{\text{lite}}$ is an acoustic network model tool that can be used to predict the longitudinal modes appearing in reactive and non-reactive geometries. It can thus be used to predict the thermoacoustic stability of combustors. $\text{OSCILOS}_{\text{lite}}$ was developed at Imperial College London by Renaud Gaudron, Alexander MacLaren, and Aimee S. Morgans. It is based on $\text{OSCILOS}_{\text{long}}$, also developed at Imperial College London, but has no Graphical User Interface. This feature makes $\text{OSCILOS}_{\text{lite}}$ faster and more adapted to parametric studies. Conversely, the equations used by the solver are the same as those of $\text{OSCILOS}_{\text{long}}$ [1,2]. The code is open source, written in MATLAB®, and distributed under an open source license. It is highly modular, easy to use, and freely available via `https://www.oscilos.com/`.

Contents

[1] r.gaudron@imperial.ac.uk

Acknowledgement

This work was supported by the European Research Council (ERC) Consolidator Grant AFIRMATIVE (2018–2023).

How to Cite OSCILOS$_{\text{lite}}$

When using OSCILOS$_{\text{lite}}$, please cite:

- Li, J., Yang, D., Luzzato, C. and Morgans, A. S. (2017). OSCILOS Report.
- Gaudron, R., MacLaren, A. and Morgans, A. S. (2021). OSCILOS_lite User Guide.

Open Source License

1 Getting Started

OSCILOS$_{\text{lite}}$ is written in MATLAB®, and is available from GitHub or as a zip file from the OSCILOS website. An active installation of MATLAB®, with the Optimisation toolbox installed, is required to run OSCILOS$_{\text{lite}}$. The code was developed on version R2020a. A technical description of the equations and algorithms included OSCILOS$_{\text{lite}}$ can be found in [2].

1.1 Structure of OSCILOS$_{\text{lite}}$

The directory tree as well as the main files used or generated by OSCILOS$_{\text{lite}}$ are represented in Fig. 1. The main script, `./OSCILOS_lite.m`, is located in the root directory. A new calculation can be launched by running this script, which will call 6 subfunctions in the order listed below. These are located in `./SubFunctions/`.

1. `Init_subfc` - reads and initialises configuration parameters.
2. `Geometry_subfc` - reads and initialises the geometry.
3. `Mean_flow_subfc` - calculates the mean flow properties throughout the domain.
4. `Flame_subfc` - reads and initialises the flame model (if applicable).
5. `BC_subfc` - reads and initialises the boundary conditions.
6. `Solver_subfc` - solves the equations and outputs the various results.

Several files are required as **inputs**, and must be located in `./Inputs` - see Sec. 2.

- The **initialisation** file (`Init.txt`) specifies the initialisation options.
- The **geometry** file (`Geometry.txt`) defines the geometry of interest.
- The **mean flow** file (`Mean_flow.txt`) defines the mean flow parameters at the inlet.
- The **flame model** file (`Flame.txt`) defines the flame model parameters.
- The **inlet** file (`Inlet.txt`) defines the acoustic boundary condition at the inlet.
- The **outlet** file (`Outlet.txt`) defines the acoustic boundary condition at the outlet.
- The **scan range** file (`Scan_range.txt`) defines the parameters to be fed to the solver.

Several **output** files may also be written to `./Outputs` depending on the initialisation options, including:

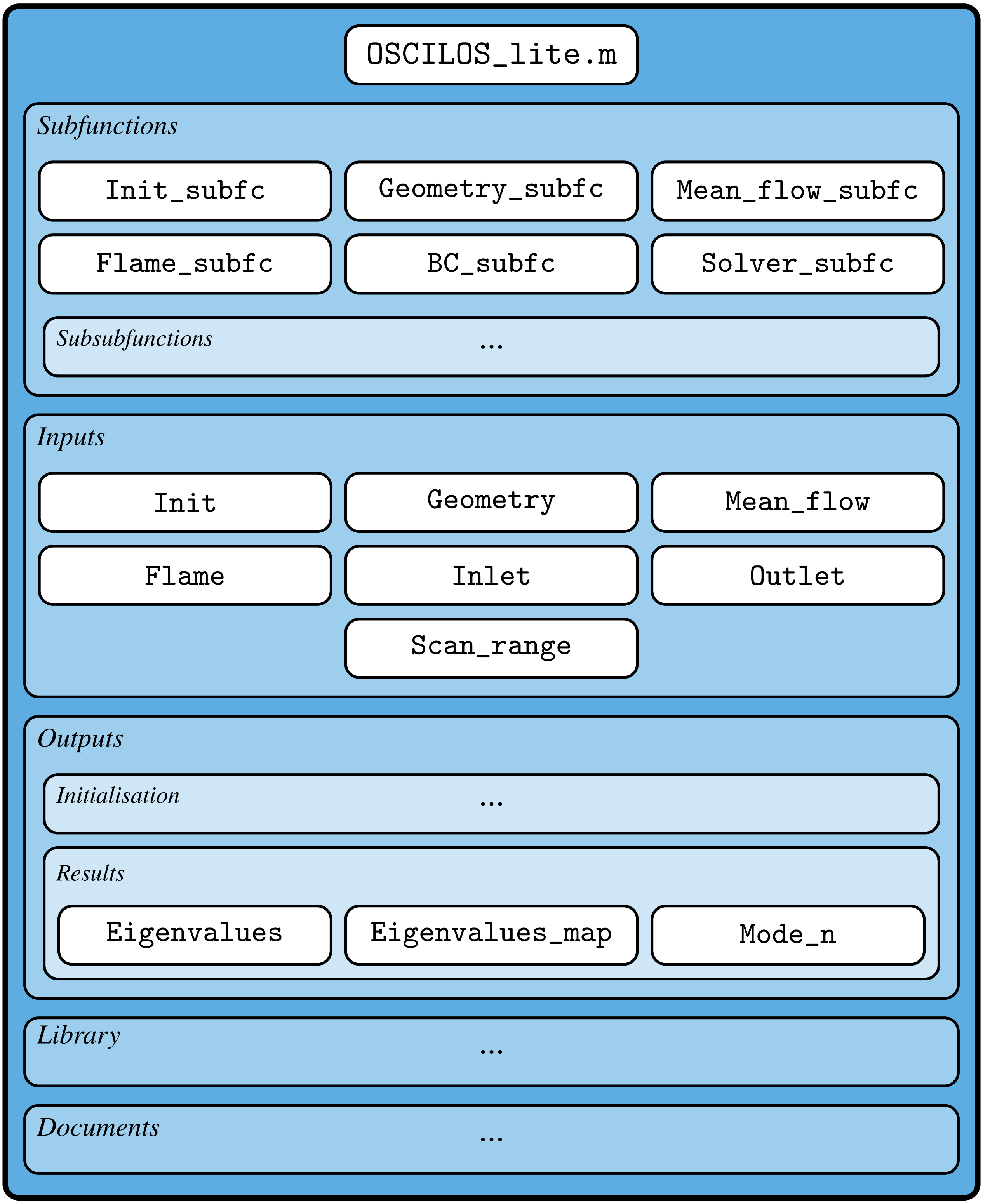

Figure 1: Directory tree and main files used by $\mathrm{OSCILOS}_{\mathrm{lite}}$.

- The **Eigenvalues list** file (`./Results/Eigenvalues.txt`) containing the frequency and growth rates of all modes in the range of interest.
- The **Eigenvalues Contour Map** file (`./Results/Eigenvalues_map.*`) depicting the residual of the characteristic equation as a function of the frequency and growth rate, thus visualising the eigenvalues. The symbol $*$ represents the `fig` or `pdf` format.
- The **Mode shape** files (`Mode_1.*`, `Mode_2.*`, ..., `Mode_n.*`) depicts the modulus of the acoustic velocity and acoustic pressure inside the geometry for the first n modes. The symbol $*$ represents the `fig` or `pdf` format.

Additional **output** files may also be written to `./Outputs` - see Sec. 3 for details. Finally, two additional folders, `./Library` and `./Documents`, contain the input files for a variety of reference cases and a selection of relevant documents respectively.

1.2 Running OSCILOS$_{\text{lite}}$

The default case is a simple duct sustaining a negligible mean flow. The duct is closed at the inlet and opened at the outlet. This case runs out-of-the-box by simply running the MATLAB script `./OSCILOS_lite.m` in the root directory. OSCILOS$_{\text{lite}}$ will then compute the mean flow everywhere in the domain, find the eigenvalues and eigenmodes, and plot a number of figures (optional). New cases may then be set up by tweaking the input files corresponding to that or other reference cases (located in the library folder `./Library`).

1.3 Assumptions

OSCILOS$_{\text{lite}}$ is based on the following assumptions:

1. 1D analysis (longitudinal modes).
2. Linear solver.
3. No entropy waves, heat exchangers, branched Helmholtz resonators or liners.
4. A single heat source (or no heat source at all).

2 Inputs

OSCILOS$_{\text{lite}}$ takes as input a number of human-readable files located in the `./Inputs/` subfolder. Each input file is whitespace-separated (e.g. tab-separated).

2.1 Initialisation

The initialisation parameters are read from the file `./Inputs/Init.txt`. The values of these parameters define whether figures are plotted and whether they are saved to `./Outputs/` among other options. The `Init.txt` file should always take the form:

Disp_figs	Small_plots	Save_pdfs	Save_figs	Save_eig	Plot_modes
{0;1}	{0;1}	{0;1}	{0;1}	{0;1}	$\mathbb{N}^+$

The different options available in the `Init.txt` file are now described:

- Figures are displayed on screen (`Disp_figs` = 1) or not (`Disp_figs` = 0).
- Small (`Small_plots` = 1) or large (`Small_plots` = 0) figures are generated.
- Figures are saved as pdf files (`Save_pdfs` = 1) or not (`Save_pdfs` = 0).
- Figures are saved as fig files (`Save_figs` = 1) or not (`Save_figs` = 0).
- The eigenvalues found by the solver are saved to a .txt file (`Save_eig` = 1) or not (`Save_eig` = 0).
- The first n modes appearing inside the geometry are represented (`Plot_modes` = n). n can be set to any whole number, including zero.

For example, the following `Init.txt` file can be used to display and save the mode shapes corresponding to the first five modes:

Disp_figs	Small_plots	Save_pdfs	Save_figs	Save_eig	Plot_modes
1	0	1	1	1	5

2.2 Geometry

2.2.1 Basics

The geometry file is located at `./Inputs/Geometry.txt`. OSCILOS$_{\mathrm{lite}}$ models the geometry using cylindrical finite elements. The elements are specified by their circular bounding faces, termed 'sections'. Each section must be located axially, and its radius specified. Each cylindrical element retains the radius of its upstream section, and extends axially to the next section. The inlet is located at the smallest axial co-ordinate, and the outlet at the largest. The sequence of sections in the file is not significant - the sections are sorted such that their axial co-ordinates are monotonically increasing, before the cylinders are constructed.

The geometry file should be tab-separated with four columns. The geometry file must begin with a line of column headers, one for each column. A blank line is interpreted as the end of the file. The file should therefore be laid out as follows:

`x[m]`	`r[m]`	`SectionIndex`	`TubeIndex`
$\mathbb{R}^+$	$\mathbb{R}^+$	`{0;10;11}`	`{0;1}`
`...`			

Each line of the file represents a section. The role of each column is outlined below:

1. `x[m]` - Axial position of the section in meters
2. `r[m]` - Section radius in meters
3. `SectionIndex` - Section type (introduced later)
4. `TubeIndex` - Tube shape (introduced later)

For example, a simple cylindrical duct is obtained with the following geometry file:

`x[m]`	`r[m]`	`SectionIndex`	`TubeIndex`
0	0.05	0	0
0.25	0.05	0	0
0.5	0.05	0	0

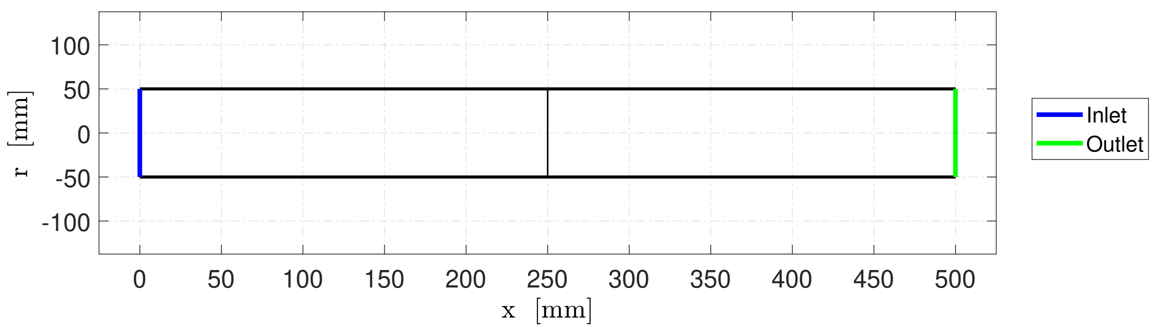

Figure 2: Cylindrical pipe with 3 sections.

OSCILOS$_{\text{lite}}$ requires at least 3 sections (2 tubes) to run - if only 2 sections are given, and the intervening tube is cylindrical (initial section `TubeIndex` = 0, no interpolation), a warning is thrown and a third section is created at the midpoint between the two, with radius the average of the others, and all other parameters equivalent to those of the inlet.

2.2.2 `TubeIndex`

If two consecutive sections have different radii, a discontinuity arises at the downstream section (`TubeIndex`=0) or is distributed over the whole tube length by dividing it axially into many shorter tubes (`TubeIndex`=1). This later approach allows a 'staircase' approximation to shapes other than cylinders. OSCILOS performs this operation internally by interpolating to a conical profile. An example is given in Fig. 3, using the geometry file:

```
x[m]      r[m]      SectionIndex      TubeIndex
0         0.05      0                 0
0.25      0.05      0                 1
0.5       0.1       0                 0
```

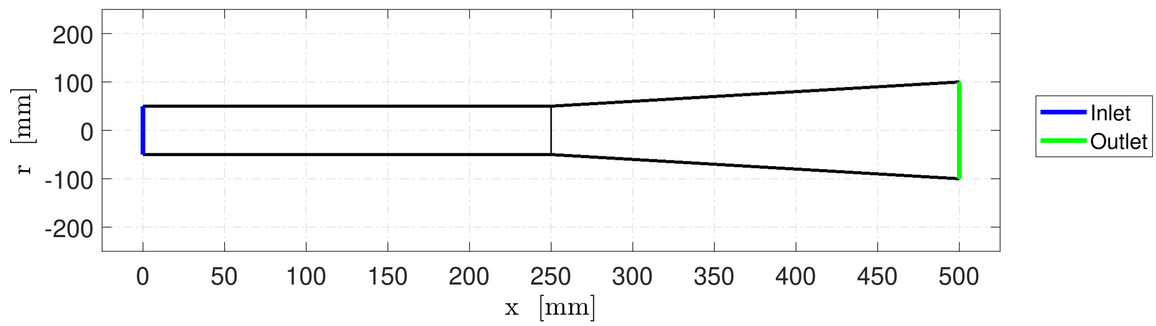

Figure 3: Cylindrical tube ending in conical horn.

The line in the file with greatest axial co-ordinate should have `TubeIndex` = 0, because there is no tube after the last section. A warning is thrown if this is not the case.

2.2.3 `SectionIndex`

In OSCILOS$_{\text{lite}}$, a given section may contain a steady heat source (`SectionIndex` = 10), a fluctuating heat source (`SectionIndex` = 11), or no heat source at all (`SectionIndex` = 0). A maximum of one steady or fluctuating heat source per geometry is currently allowed in OSCILOS$_{\text{lite}}$. An example of a steady heat source placed in the middle of a cylindrical duct is given in Fig. 4, using the geometry file:

x[m]	r[m]	SectionIndex	TubeIndex
0	0.05	0	0
0.25	0.05	10	0
0.5	0.05	0	0

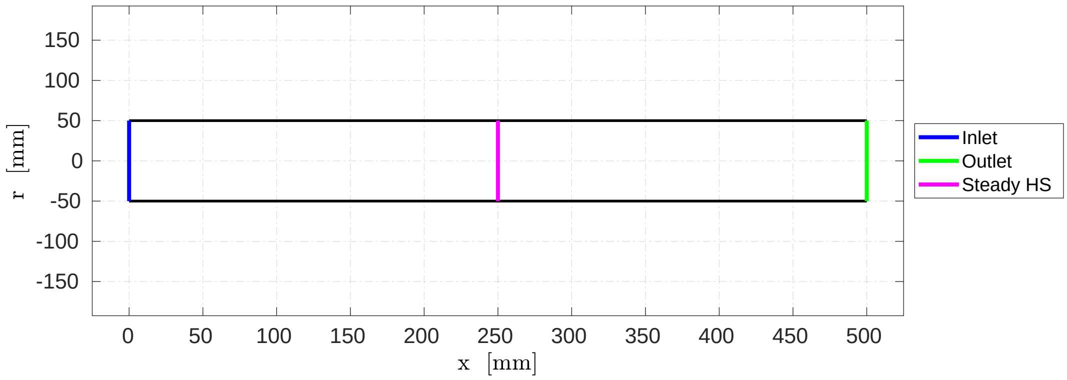

Figure 4: Steady heat source in the middle of a cylindrical tube.

2.3 Mean Flow

The mean flow parameters are read from the file `./Inputs/Mean_flow.txt`. These parameters define the mean flow at the inlet of the geometry (except for `Delta_T_HS`). OSCILOS$_{\text{lite}}$ will then compute the mean flow parameters throughout the geometry based on these inlet parameters. The `Mean_flow.txt` file should always take the form:

P1[Pa]	T1[K]	choice_M1_u1	M1_u1	Choice_gamma	Delta_T_HS
$\mathbb{R}^+$	$\mathbb{R}^+$	{1;2}	$\mathbb{R}^+$	1	$\mathbb{R}^+$

Each parameter in the `Mean_flow.txt` file is now described:

- `P1[Pa]`: Inlet Pressure in Pa.
- `T1[K]`: Inlet Temperature in K.
- `choice_M1_u1`: Choice between specifying the inlet Mach number (`choice_M1_u1` = 1) or inlet velocity (`choice_M1_u1` = 2).
- `M1_u1`: Value of the inlet Mach number or inlet velocity in m/s.
- `choice_gamma` = 1: Constant adiabatic index (other options not supported yet)

- `Delta_T_HS`: Temperature ratio across the heat source (if applicable).

Below is an example of a `Mean_flow.txt` file corresponding to a low Mach number flow at the inlet and no heat source within the domain:

P1[Pa]	T1[K]	choice_M1_u1	M1_u1	Choice_gamma	Delta_T_HS
101325	293	1	0.001	1	1

OSCILOS$_{\text{lite}}$ requires a nonzero mean flow to avoid singularity of the system of flow equations. If the magnitude of the inlet Mach number (or inlet velocity) is small enough to risk singularity or ill-conditioning of the flow equations, it will be increased to a negligible, but finite, value (currently `1e-5`).

2.4 Scan Range

The OSCILOS solver uses an eigensolver which performs a search for the complex eigenvalues representing the modes of the system. The scope of the search, in terms of frequency and growth rate, is determined from the scan range configuration parameters read from the file `./Inputs/Scan_range.txt`. This file should always take the form:

min_freq	max_freq	number_freq	min_GR	max_GR	number_GR
$\mathbb{R}^+$	$\mathbb{R}^+$	$\mathbb{N}^+$	$\mathbb{R}^+$	$\mathbb{R}^+$	$\mathbb{N}^+$

The eigenvalue search is conducted by MATLAB's `fsolve()` from seed points spaced linearly throughout frequency and growth rate ranges specified - the number of seed points is configurable using `number_freq` and `number_GR`. If the number of modes is close to or greater than `number_freq`, the solver may miss eigenvalues, and this will be evident on the contour plot, see Sec. 3. Each parameter in the `Scan_range.txt` file is now described:

- `min_freq`: Frequency scan lower bound in Hz.
- `max_freq`: Frequency scan upper bound in Hz.
- `number_freq`: Number of frequency starting points.
- `min_GR`: Growth rate scan lower bound in 1/s.
- `max_GR`: Growth rate scan upper bound in 1/s.

- `number_GR`: Number of growth rate starting points.

Below is an example of a `Scan_range.txt` file:

min_freq	max_freq	number_freq	min_GR	max_GR	number_GR
0	1000	20	-200	200	20

2.5 Flame Model

If a fluctuating flame is present inside the geometry, i.e. if there is a section such that `SectionIndex` $= 11$, then a flame model needs to be specified using the file `./Inputs/Flame.txt`. This file should always take the form:

Type	Param_1	Param_2
1	$\mathbb{R}^+$	$\mathbb{R}^+$

The only available flame model as of v2.0 is the $n - \tau$ model, corresponding to `Type` $= 1$. In that case, `Param_1` corresponds to the gain (i.e. n) and `Param_2` corresponds to the time delay (i.e. τ) in ms. Below is an example of a `Flame.txt` file corresponding to a unity gain and a time delay of 3 ms.

Type	Param_1	Param_2
1	1	3

2.6 Boundary Conditions

OSCILOS$_{\text{lite}}$ supports several different boundary condition (BC) types, which are specified at the inlet and outlet using the files `./Inputs/Inlet.txt` and `./Inputs/Outlet.txt` respectively. Each of these files should always take the form:

Type	Param_1	Param_2	Param_3
$\mathbb{N}^+$	$\mathbb{R}^+$	$\mathbb{R}^+$	$\mathbb{R}^+$

where

- `Type`: Specifies the boundary condition type.

- `Param_1`: First parameter (if required).
- `Param_2`: Second parameter (if required).
- `Param_3`: Third parameter (if required).

The number of parameters required and interpretation of their values depends on the boundary condition type selected. Supported values of the `Type` parameters are given in Table 1.

3 Outputs

The outputs generated by $\text{OSCILOS}_{\text{lite}}$ fall into two categories: initialisation outputs, saved to `./Outputs/Initialisation/`, and results outputs, saved to `./Outputs/Results/`. The initialisation outputs are produced before the solution begins, and serve to visualise and record the simulation setup. The results outputs visualise and record the various results yielded by the solver.

3.1 Initialisation

Initialisation outputs consist of a set of 3 or 4 figures representing the geometry, the mean flow throughout the domain, the acoustic boundary conditions at the inlet/outlet, and the flame model (if applicable). These are plotted if the `Disp_figs` initialisation option is set to 1 and saved to `./Outputs/Initialisation/` if `Save_figs` and/or `Save_pdfs` are set to 1.

The geometry plot shows the geometry after any interpolation is complete. The boundaries of the geometry are shown as thick black lines, and the tubes are oriented so that the flow direction is left → right. Inlet and outlet are shown as thick vertical coloured lines. The internal sections provided in the original geometry file are marked with thin vertical black lines. The 'staircasing' created by interpolating over a tube with nonzero `TubeIndex` is left without vertical lines. Finally, steady (respectively fluctuating) heat sources are represented as thick vertical purple (respectively red) lines.

The mean flow plot visualises the mean axial velocity and mean temperature distributions throughout the geometry. Where a radius discontinuity exists in the geometry file, a gap appears in the mean velocity and temperature curves to reflect the jump condition. Interpolated tubes by contrast are 'staircased' in the mean flow plots, so that the effect of the discretisation level on mean flow is evident.

Table 1: Boundary Condition types

Type	Meaning
1	Open condition, $\tilde{\mathcal{R}} = -1$
2	Closed condition, $\tilde{\mathcal{R}} = 1$
3	Not implemented *Reserved for Choked condition*
4	Time lag condition, $\tilde{\mathcal{R}}(s) = Ae^{-s\tau}$ $A =$ `Param1` (amplitude) $\tau =$ `Param2`$/1000$ (time lag in ms)
5	Phase lag condition, $\tilde{\mathcal{R}} = Ae^{-i\varphi}$ $A =$ `Param1` (amplitude) $\varphi =$ `Param2` (phase) `Param3` $= 0$ for `Param2` in radians `Param3` $= 1$ for `Param2` in degrees
6	Not implemented *Reserved for Polynomial from Coefficients with Time Delay condition*
7	Not implemented *Reserved for Polynomial from User Data with Time Delay condition*
8	Not implemented *Reserved for Heat Exchanger condition*
9	Not implemented *Reserved for Gain and Phase Interpolated from User Data condition*
10	Not implemented *Reserved for User-defined Gain and Phase functions condition*
11	Unflanged Levine-Schwinger condition [3] using polynomial approximations for $\|R\|$ and l by Norris and Sheng [4] $\tilde{\mathcal{R}}(k) = -\|R\|e^{-2ikl} \qquad k(s) = \frac{\mathrm{Im}[s]}{c}$
12	Flanged Levine-Schwinger condition using polynomial approximations for $\|R\|$ and l by Norris and Sheng [4] $\tilde{\mathcal{R}}(k) = -\|R\|e^{-2ikl} \qquad k(s) = \frac{\mathrm{Im}[s]}{c}$

The boundary conditions plot shows the gain and phase of the complex reflectances at the inlet and outlet as functions of frequency over the frequency range specified by the scan range configuration parameters.

The flame model plot shows the gain and phase of the Flame Transfer Function (FTF) as functions of frequency over the frequency range specified by the scan range configuration parameters.

3.2 Results

Results outputs include: a text file containing the modes found, the eigenvalue contour map, and mode shape plots.

3.2.1 Eigenvalue List File

The eigenvalue list file contains a summary of the solver results in whitespace-separated, human-readable form. It is saved to `./Outputs/Results/Eigenvalues.txt` if `Save_eig` is set to 1. It comprises 3 columns, labelled:

```
Mode number   Frequency [Hz]   Growth rate [1/s]
```

The `Mode number` column contains a list of ascending integers describing the modes in order of ascending frequency. For more discussion of `Frequency` and `Growth rate`, see Sec. 3.2.2.

3.2.2 Eigenvalue Contour Map

The eigenvalue contour map visualises the residual in the transmission-line calculation. The eigenvalues of the system, and hence the acoustic modes, lie at the minima of this function. This figure is plotted if the `Disp_figs` initialisation option is set to 1 and saved to `./Outputs/Results/Eigenvalues_map.*` (where `.*` denotes `.fig` or `.pdf`) if `save_figs` and/or `save_pdfs` are set to 1.

The plot represents the Laplace variable space, in the range specified by the scan range configuration parameters, with the real part corresponding to growth rate on the abcissa, and the imaginary part converted to `[Hz]` on the ordinate. The resolution of the plot (number of residuals calculated) is 10 times the `num_freq` and `num_GR` parameters, in the respective axes. The eigenvalues previously found by the solver are overlaid on the contour plot as white stars. In some cases, the solver may miss one or more eigenvalues. If this happens, the contour plot will exhibit minima which have no white star, indicating that the solver was poorly configured and the calculation should be repeated. In most cases, increasing `num_freq` will solve the problem, but the user may also increase `num_GR` if there is large variation in growth rate.

3.2.3 Mode Shapes

OSCILOS$_{\text{lite}}$ also calculates the mode shape, i.e. the axial distribution of acoustic pressure and acoustic velocity magnitude, for each mode. These are visualised by the mode shape plots, displayed on screen if the `Disp_figs` initialisation option is set to 1 and saved to `./Outputs/Results/Mode_n.*` (where `.*` denotes `.fig` or `.pdf`) if `save_figs` and/or `save_pdfs` are set to 1. The number of mode shapes plotted is controlled by the `plot_modes` configuration parameter, and the number of modes found overall within the scan range - whichever is the lowest.

4 Example: the NoiseDyn burner

The NoiseDyn burner is a perfectly-premixed confined turbulent combustor located at EM2C laboratory [5–7]. Its thermoacoustic stability has been investigated experimentally and the objective of this section is to characterise it using OSCILOS$_{\text{lite}}$.

4.1 Input files

The first step is to generate the correct input files. The `Init.txt` file is set as follows:

Disp_figs	Small_plots	Save_pdfs	Save_figs	Save_eig	Plot_modes
1	0	1	1	1	3

Following this `Init.txt` file, all figures will be displayed on screen and saved to `./Outputs` to both .fig and .pdf format. The eigenvalues list file will also saved to that folder. Up to three mode shape plots will also be displayed on screen and saved to `./Outputs`.

The `Geometry.txt` file is then used to specify the geometrical shape of the NoiseDyn burner:

x[m]	r[m]	SectionIndex	TubeIndex
0	0.0325	0	0
0.295	0.011	0	0
0.315	0.02	0	0
0.345	0.007348	0	0
0.353	0.01	0	0
0.409	0.046263	11	0
0.663	0.0325	0	0
1.323	0.0325	0	0

It is worth noting that a fluctuating heat source, described by `SectionIndex = 11`, is located at the bottom of the combustion chamber at `x = 0.409`.

Next, the mean pressure, temperature, and velocity are specified at the inlet of the geometry using the `Mean_flow.txt` file. The temperature jump across the flame is also specified in that file:

P1[Pa]	T1[K]	choice_M1_u1	M1_u1	Choice_gamma	Delta_T_HS
101325.0	293.0	2	3	1	6

The acoustic boundary condition at the inlet is then set to a closed end in the `Inlet.txt` file:

Type	Param_1	Param_2	Param_3
2	-	-	-

Conversely, the acoustic boundary condition at the outlet is set to an unflanged Levine-Schwinger condition (used to accurately model an open end) in the `Outlet.txt` file:

Type	Param_1	Param_2	Param_3
11	-	-	-

The flame model is then set to a $n - \tau$ model with a unity gain and a time delay of 2.1 ms in the `Flame.txt` file:

Type	Param_1	Param_2
1	1	2.1

Finally, $\text{OSCILOS}_{\text{lite}}$ will look for modes appearing in the NoiseDyn burner in the [0 : 400 Hz] frequency range and in the [-200 : 200 s^{-1}] growth rate range, as specified in the `Scan_range.txt` file:

min_freq	max_freq	number_freq	min_GR	max_GR	number_GR
0	400	50	-200	200	50

4.2 Output files

Once the inlet files have been specified, the last step is to run the main script called `OSCILOS_lite.m` in Matlab. $\text{OSCILOS}_{\text{lite}}$ then finds the eigenfrequencies and eigenmodes of the problem and a number of plots are displayed on screen and saved to `./Outputs`.

The first initialisation plot, represented in Fig. 5, depicts the geometry used to represent the NoiseDyn burner in $\text{OSCILOS}_{\text{lite}}$. As expected from the `Geometry.txt` file, a fluctuating heat source is located at the bottom of the combustion chamber.

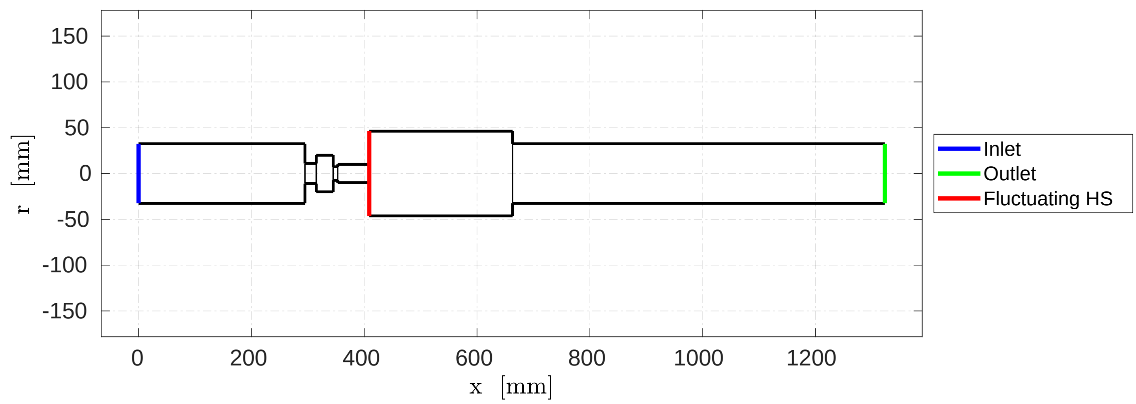

Figure 5: Geometry corresponding to the NoiseDyn burner as generated by $\text{OSCILOS}_{\text{lite}}$.

The second initialisation plot, represented in Fig. 6, depicts the acoustic reflection coefficient at the inlet and outlet of the domain. As expected from the `Inlet.txt` and `Outlet.txt` files, it corresponds to a closed end at the inlet and an open end with radiative losses at the outlet.

The third initialisation plot, represented in Fig. 7, depicts the mean flow parameters throughout the domain. These are computed by $\text{OSCILOS}_{\text{lite}}$ based on the `Mean_flow.txt` file.

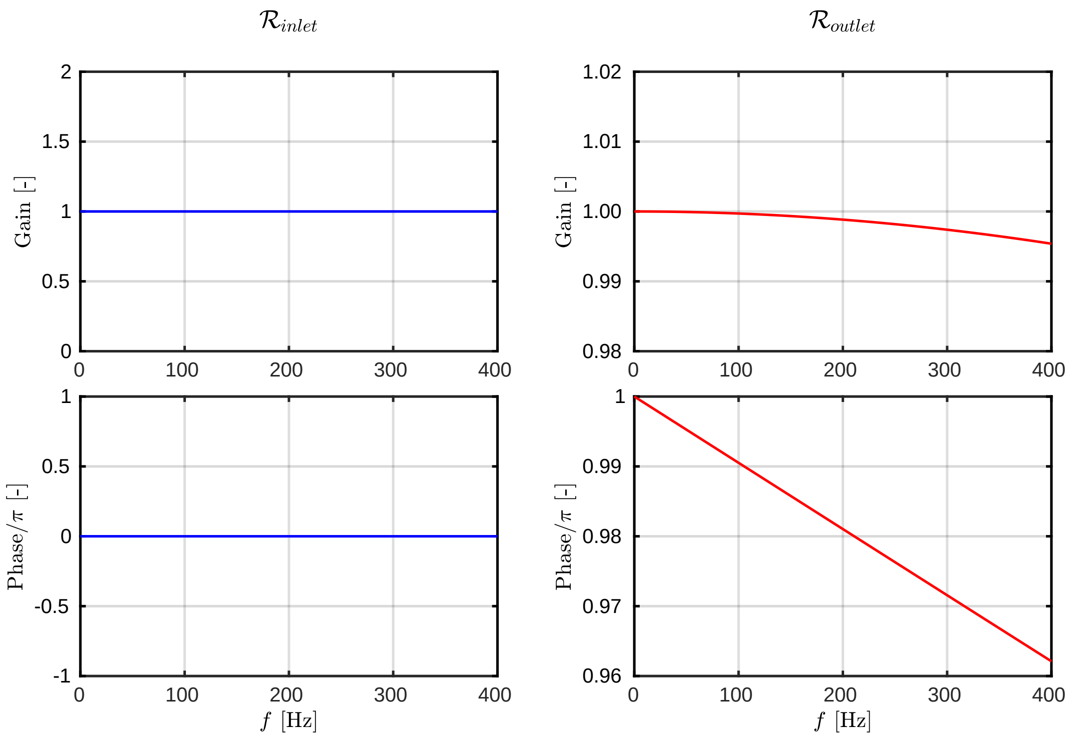

Figure 6: Gain (Top) and phase (Bottom) of the acoustic reflection coefficient at the inlet (Left) and outlet (Right) of the geometry as a function of frequency.

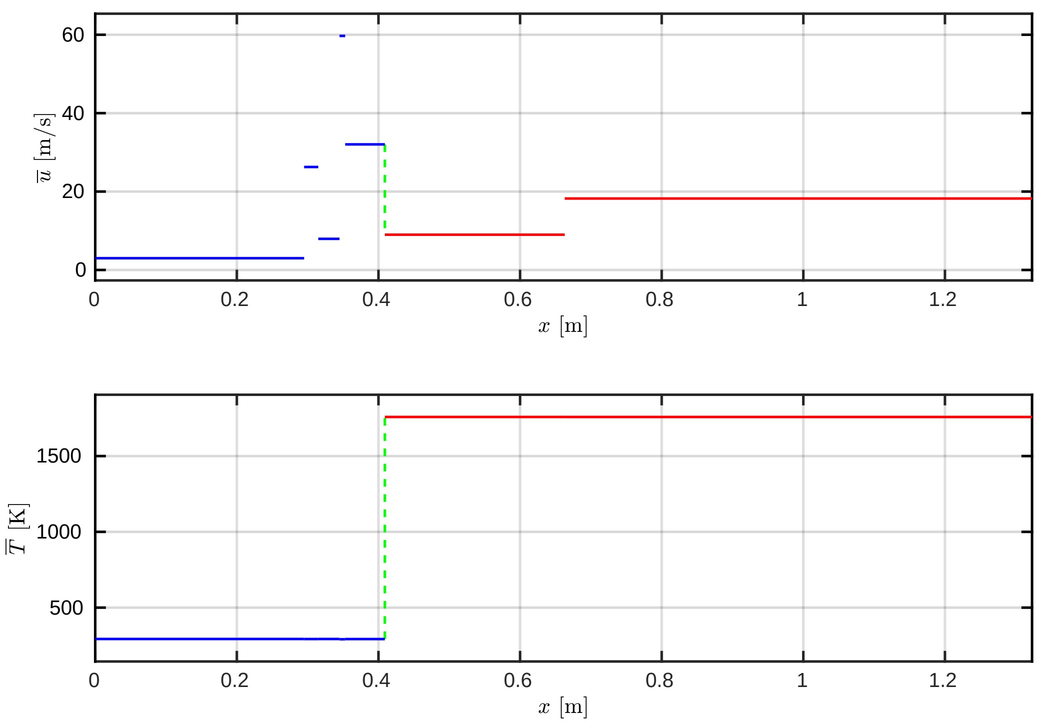

Figure 7: Mean velocity (Top) and mean temperature (Bottom) throughout the NoiseDyn burner as calculated by $\text{OSCILOS}_{\text{lite}}$.

The fourth and last initialisation plot, represented in Fig. 8, depicts the flame frequency response to incoming acoustic waves, as specified by the `Flame.txt` file.

In addition to these initialisation plots, several results plots are also generated by $\mathrm{OSCILOS}_{\mathrm{lite}}$. The first one is the eigenvalue contour map, shown in Fig. 9, which shows that there are two modes in the range of interest. The first one has a frequency $f \sim 50$ Hz and is linearly stable while the second one has a frequency $f \sim 180$ Hz and is linearly unstable. This indicates that the NoiseDyn burner is predicted to be thermoacoustically unstable by $\mathrm{OSCILOS}_{\mathrm{lite}}$, which is in agreement with experiments [5–7].

The mode shapes associated with these two modes are also generated by $\mathrm{OSCILOS}_{\mathrm{lite}}$, as represented in Fig. 10. This shows that the unstable thermoacoustic mode appearing in the NoiseDyn burner is reminiscent of a 3/4 wave mode with maximal pressure fluctuations inside the combustion chamber. These two observations are also in accordance with previously pubished experimental data [5].

Finally, the eigenvalues list file specifies the exact frequencies and growth rates for all modes identified by $\mathrm{OSCILOS}_{\mathrm{lite}}$, as reproduced in Table 2. The frequency of the unstable thermoacoustic mode is predicted to be 186.4 Hz, very close to the experimental value of 185.2 Hz [5].

Mode number	Frequency [Hz]	Growth rate [1/s]
1	53.71	-197.95
2	186.36	160.34

Table 2: Eigenvalues identified by $\mathrm{OSCILOS}_{\mathrm{lite}}$ for the NoiseDyn burner.

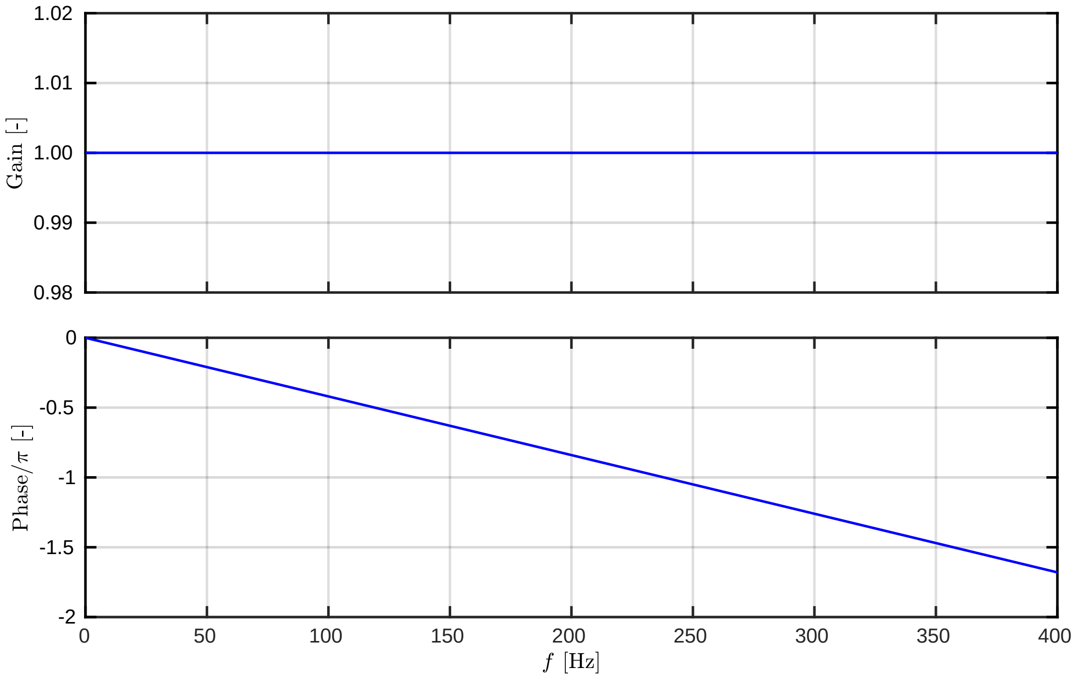

Figure 8: Gain (Top) and phase (Bottom) of the Flame Transfer Function as a function of frequency.

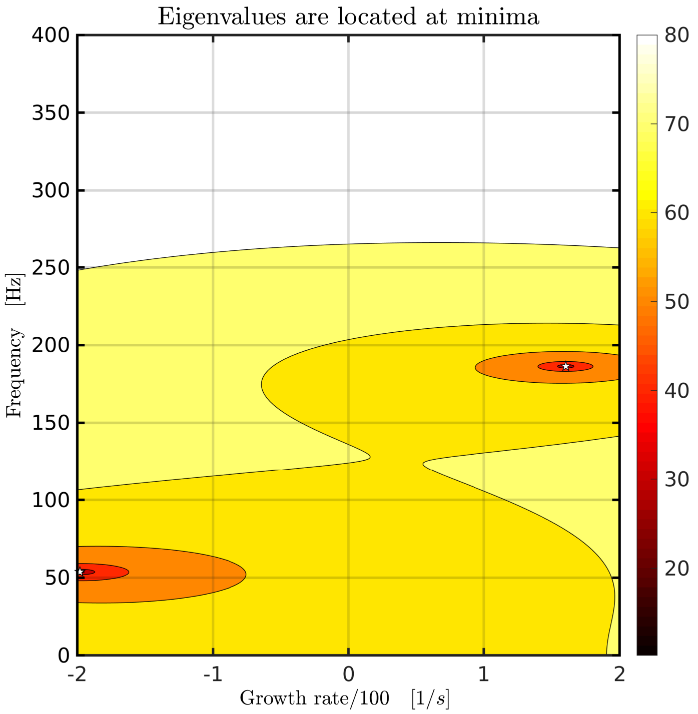

Figure 9: Eigenvalue contour map as computed by $\text{OSCILOS}_{\text{lite}}$. A white star indicates the presence of an acoustic mode.

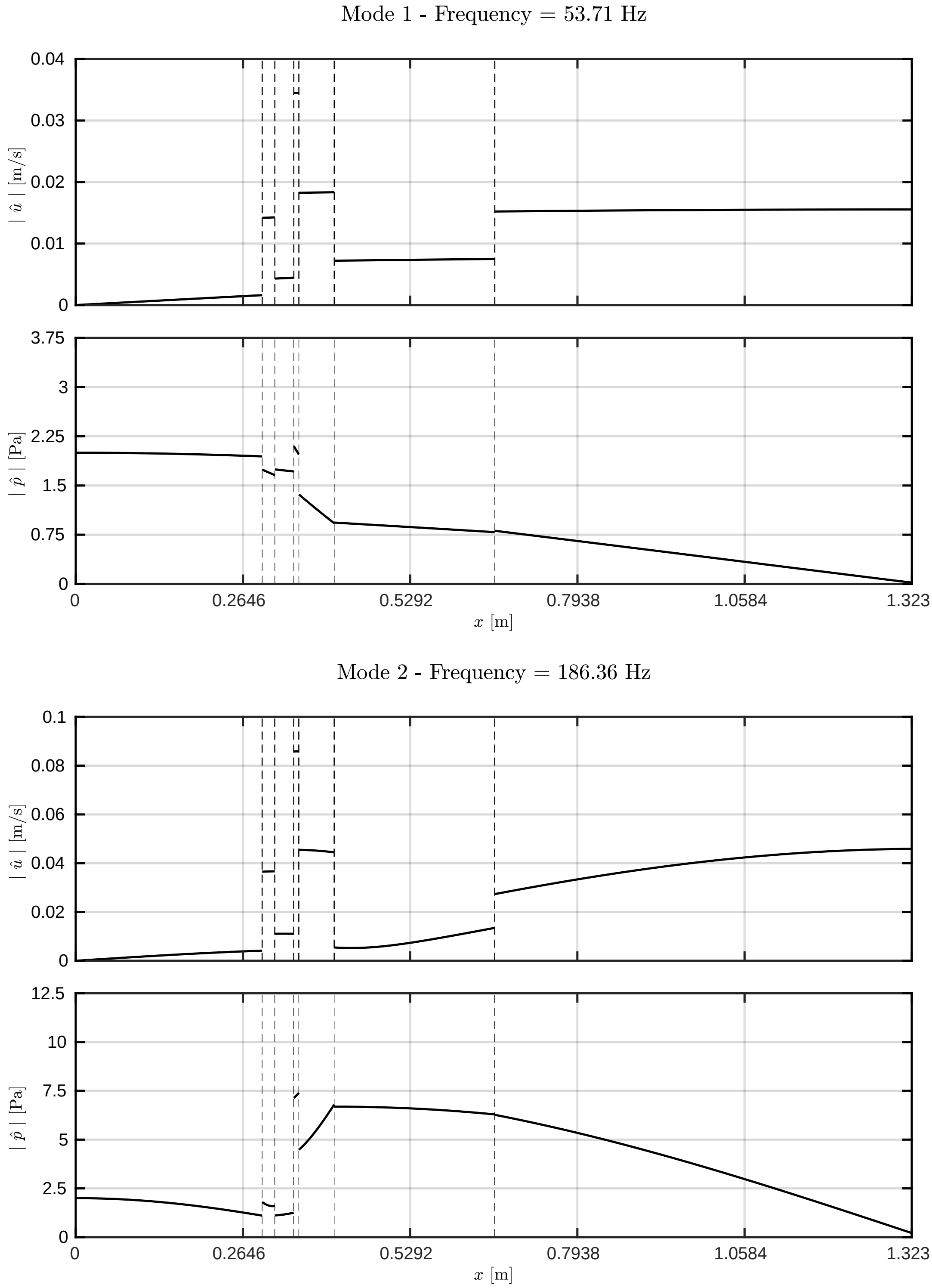

Figure 10: The first two modes appearing in the NoiseDyn burner according to OSCILOS$_{\text{lite}}$. For each mode, the modulus of the acoustic velocity (Top) and acoustic pressure (Bottom) are represented throughout the geometry.

References

[1] X. Han, J. Li, and A. S. Morgans. Prediction of combustion instability limit cycle oscillations by combining flame describing function simulations with a thermoacoustic network model. Combust. Flame, 162(10):3632–3647, 2015.

[2] J. Li, D. Yang, C. Luzzato, and A.S. Morgans. OSCILOS Report. Technical report, Imperial College London, London, 2017.

[3] H. Levine and J. Schwinger. On the Radiation of Sound from an Unfianged Circular Pipe. Physical Review, 73(4):137–153, 1948.

[4] A. N. Norris and I. C. Sheng. Acoustic radiation from a circular pipe with an infinite flange. Journal of Sound and Vibration, 135(1):85–93, 11 1989.

[5] R. Gaudron. Acoustic response of premixed flames submitted to harmonic sound waves. PhD thesis, Université Paris-Saclay, 2018.

[6] R. Gaudron, M. Gatti, C. Mirat, and T. Schuller. Flame Describing Function of a confined premixed swirled combustor with upstream and downstream forcing. J. Eng. Gas Turbines Power, 2018.

[7] R. Gaudron, M. Gatti, C. Mirat, and T. Schuller. Impact of the acoustic forcing level on the transfer matrix of a turbulent swirling combustor with and without flame. Flow Turbul. Combust., 103(12), 2019.