Master of Science Thesis

Linear aeroacoustic solver in OpenFOAM

J.T. Akhnoukh

June 18, 2015

Faculty of Aerospace Engineering

Delft University of Technology

Linear aeroacoustic solver in OpenFOAM

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace
Engineering at Delft University of Technology

J.T. Akhnoukh

June 18, 2015

Faculty of Aerospace Engineering · Delft University of Technology

TU Delft
Delft
University of
Technology

Copyright © J.T. Akhnoukh
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **“Linear aeroacoustic solver in OpenFOAM”** by **J.T. Akhnoukh** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: June 18, 2015

Head of department: ____________________
prof. dr. S. Hickel

Supervisor: ____________________
dr. ir. A.H. van Zuijlen

Supervisor: ____________________
ir. W.C.P. van der Velden

Reader: ____________________
dr. ir. M. Snellen

Summary

Due to rapid advances in computer technology the field of computational aeroacoustics has become an alternative to empirical and experimental methods for the prediction of sound production and propagation. However, in order to provide numerical aeroacoustic predictions for realistic applications a hybrid approach is still often the only option. OpenFOAM is a popular open-source CFD package in the academic world and its usage is increasing in industry as alternative for expensive commercial software. Currently there is no acoustic module in OpenFOAM. Following a hybrid methodology, the user would therefore need to transfer saved flow data to another software package to perform the acoustic calculation. This requires high data storage and makes the method (unnecessarily) complicated and cumbersome.

In this thesis an aeroacoustic method based on the linearized Euler equations (LEE) has been implemented in OpenFOAM. Instead of using the standard available OpenFOAM schemes, a different numerical method was implemented. This involves solving a Riemann problem to determine the convective terms. The code was verified on a series of benchmark problems for which analytical or reference solutions are available. The considered problems were one-dimensional wave propagation, the propagation of an acoustic pulse in uniform flow, the radiation by elementary acoustic sources and source radiation in two types of non-uniform flow. In all cases the found numerical solution was very close to the reference solution. The Riemann based solver was compared for these problems with a LEE solver that uses the readily available discretization schemes of OpenFOAM. Both implementations were shown to be able to correctly account for convection and refraction effects. The Riemann based solver has as advantage that it propagates waves in a less dispersive manner than the standard solver, but has as disadvantage that it is more diffusive. As validation case the sound production by a pair of co-rotating vortices was chosen. In addition to reference data, a direct simulation was carried out to also serve as reference solution. The hybrid method was tested with the LEE and with Lighthill's analogy. If the average acoustic source is not included as source term in the LEE, the LEE and Lighthill solutions were found to be very similar. Compared to the direct simulation, both hybrid methods showed a clear phase difference, which for this case can be attributed to the start-up phase of the hybrid simulation. Finally, it was demonstrated that the flow calculation can be done on small domain, followed by a mapping of the obtained acoustic source onto the larger acoustic grid, after which the acoustic equations are solved for that flow time step. This leads to considerable savings in computational power and disk space usage.

Acknowledgements

It took me some time to realize that sitting home, nice and comfortably, sometimes trying or pondering would for a thesis not lead to the progress that I got used to during the Bachelor and Master. I then started to visited Delft more frequently. The discussions with my supervisors dr. ir. Alexander van Zuijlen and ir. Wouter van der Velden were very important and fun too. I could often just march into their offices and ask for comments on my line of thought, fresh ideas or any sort of help, without having the feeling that I would soon be kicked out. Consequently, steady progress was made.

Apart from the help of my supervisors, I also received help from outside the TU Delft. The couple of visits of professor Hrvoje Jasak (the mastermind behind OpenFOAM) greatly speeded up the programming in OpenFOAM. I also benefited from many email discussions with dr. Lilla Koloszar of the von Karman Institute.

Last but not least, I would like to thank my parents for making studying possible in the first place.

Contents

Summary v

Acknowledgements vi

1 Introduction 1

2 Computational aeroacoustics 3

2.1 Hybrid methods 3

2.1.1 Lighthill's acoustic analogy 3

2.1.2 Curle's acoustic analogy 4

2.1.3 Ffowcs Williams-Hawkings formulation 5

2.1.4 Linearized Euler equations 6

2.2 Wave propagation errors 8

2.2.1 Basic concepts 8

2.2.2 Numerical schemes 10

2.3 Boundary conditions 10

2.3.1 Characteristic boundary condition 10

2.3.2 Buffer zone 12

3 The Riemann solver 13

3.1 The Riemann problem for linear hyperbolic equations 13

3.2 The Riemann based LEE solver 15

3.3 Implementation in OpenFOAM 18

4 Verification **21**
4.1 One-dimensional wave propagation 21
4.2 Gaussian pulse in uniform flow 26
4.3 Elementary sources 28
  4.3.1 Monopole 28
  4.3.2 Dipole 35
  4.3.3 Quadrupole 35
4.4 Sound reflection from a solid wall 36
4.5 Source radiation in non-uniform flow 38
  4.5.1 Monopole radiation in a shear flow 38
  4.5.2 Monopole radiation in a jet flow 42
4.6 Interim conclusion 45

5 Validation of the hybrid method **47**
5.1 Direct Noise Simulation 47
  5.1.1 Flow description 47
  5.1.2 Flow solver 49
  5.1.3 Mesh design 50
  5.1.4 Flow results 50
  5.1.5 Acoustic results 54
5.2 Hybrid Method 58
  5.2.1 Source reconstruction 58
  5.2.2 LEE simulation 60

6 Noise from a beveled trailing edge **67**
6.1 Flow simulation 67
6.2 Acoustic simulation 69

7 Conclusions and recommendations **71**
7.1 Conclusions 71
7.2 Recommendations 72

References **73**

A Green's function **77**

B Linearization of the Euler equations **79**
- B.1 Governing equations of fluid flow 79
  - B.1.1 Navier-Stokes equations 79
  - B.1.2 Euler equations 80
- B.2 Linearized Euler equations 80

C Inclusion of mean density variations **85**

Chapter 1

Introduction

Aircraft noise emission during landing and take-off is one of the main challenges in aeronautical engineering, in particular due the continuously stringent noise reduction targets imposed on civil aircraft. Noise reduction is also an active topic in the wind energy sector, because the power generation of onshore wind turbines can be limited by noise restrictions. More examples can be found in other engineering disciplines, such as automotive and civil engineering. In order to design for less noise, accurate tools are needed to predict the acoustic source terms generated by the flow and the subsequent propagation of waves. With experimental methods it is possible to investigate high Reynolds number flows, but it is not always feasible or cost-efficient to build, test and then perform the experiment for a broad range of flow conditions or designs. Empirical methods are easy to use, but in many cases not applicable or accurate enough. The field of computational aeroacoustics (CAA) provides an alternative for the prediction of sound compared with empirical and experimental methods. In conjunction with advances in computational fluid dynamics, CAA has matured over the years. However, most of the developed methods in CAA are not open-source. For academics it is preferred to work with open-source codes, since these are free and pose no restrictions on code sharing or publication. In order to increase the understanding and usage of CAA, it is necessary to provide the academic world with open-source alternatives. For this reason we would like to investigate the possibility to implement an aeroacoustic method in an open-source software package. The chosen software is OpenFOAM, due to its increasing popularity in both research communities and industry.

The coupling of aerodynamics and acoustics poses several computational difficulties, which are related to either scale differences or boundary conditions:

- For low Mach numbers a large length scale separation between the acoustic and flow fields exist [33].
- There is a large difference between the magnitudes of the acoustic and aerodynamic disturbances [8]. Numerical errors may overshadow sound production due to its low acoustic efficiency if both flow and sound are calculated simultaneously.

- Special treatment of the boundaries is essential, since reflection from outwards propagating waves back into the computational domain distorts the solution [26].
- Commonly used levels of artificial dissipation and dispersion in conventional CFD can greatly attenuate the waves as they propagate to the far field. This puts high demands on the numerical scheme [8].

The field of CAA can be divided in two main branches: a direct simulation and a hybrid approach. In the first branch the full compressible flow equations are solved for both the flow and sound propagation. The domain covers the sound production and at least part of the near acoustic field. Due to the high computational cost originating from the earlier mentioned scale separation, a direct calculation remains restricted to simple geometries and to low and moderate Reynolds numbers [33]. In a hybrid method the flow and acoustic field are calculated separately so that the numerical method can be optimized for the physics that are solved. A hybrid method assumes that there is a one-way coupling between flow and acoustics: the unsteady flow generates sound and alters its propagation, but there is no feedback from the sound waves to the flow. This assumption is justified for low Mach number flows [33]. Relevant engineering applications can be analyzed with this approach and hence in this project the focus lies on a hybrid method applied to the low Mach number regime. With OpenFOAM it is possible to do both the flow and acoustic calculation. This is more convenient than using two separate software packages, which would require flow data to be transferred to the acoustic solver.

As indicated above, one challenge when applying a hybrid method is the need for a good numerical scheme. The standard schemes of OpenFOAM are lacking from this perspective. We therefore propose to solve a Riemann problem with second order accuracy to determine the flux of the convective terms. The goals can be summarized as follows:

- Implementation and verification of a hybrid method based on a system of equations.
- Implementation of a high-resolution scheme, verify it and compare it with the standard schemes.
- Validation of the method.
- Real case application.

In chapter 2 an introduction is given to the necessary theory of aeroacoustics for this thesis. The derivation of the Riemann solver and its implementation in OpenFOAM is discussed in chapter 3. The results from benchmarking the solver are shown in chapter 4. Validation results are presented in chapter 5 and the results from a realistic case are shown in chapter 6. Conclusions and recommendations are given in the last chapter.

Chapter 2

Computational aeroacoustics

In this chapter a brief overview is given of hybrid methods (Sec. 2.1), basic concepts of numerically simulated wave propagation (Sec. 2.2) and boundary treatment (Sec. 2.3).

2.1 Hybrid methods

Noise predictions are typically made by a two-step approach due to the large difference in requirements on the flow field and acoustic propagation. In the following paragraphs a few of these so-called hybrid methods are explained.

2.1.1 Lighthill's acoustic analogy

An acoustic analogy is a reformulation of the conservation laws of fluid motion. The first acoustic analogy was due to Lighthill [21] and it forms the basis of much other work performed in the field of aeroacoustics. Lighthill's equation is:

$$\left(\frac{\partial^2}{\partial t^2} - c_0^2\nabla^2\right)\rho' = \frac{\partial^2 T_{ij}}{\partial x_i \partial x_j}, \tag{2.1}$$

where $T_{ij} = \rho u_i u_j - \tau_{ij} + [(p - p_0) - c_0^2(\rho - \rho_0)]\delta_{ij}$ is the Lighthill stress tensor and c_0 is the speed of sound. The aeroacoustic problem has been reformulated into an equivalent acoustic problem in a stagnant medium: An acoustic *analogy.* The terms on the right-hand side are assumed to be known from a flow simulation, while the left-hand side represents the propagation of the acoustic quantity. Usually the last two terms of T_{ij} (viscous stress, sound generation due to shear and entropy fluctuations) are neglected for flows with a high Reynolds number and low Mach number, leaving only the momentum contribution $\rho u_i u_j$.

In order to solve Eq. (2.1) analytically, a Green's function may be utilized. For free space propagation, such a function can be found and from this follows the formal solution to

Eq. (2.1) (see Appendix A for a clarification):

$$\rho'(\mathbf{x},t) = \frac{1}{4\pi c_0^2}\frac{\partial^2}{\partial x_i \partial x_j}\int\limits_V \frac{T_{ij}(\xi, t-\|\mathbf{x}-\xi\|/c_0)}{\|\mathbf{x}-\xi\|}\mathrm{d}^3\xi, \tag{2.2}$$

where $\mathbf{x}$ is the distance from a reference system to the observer and ξ the distance from the reference system to the source, see Fig. 2.1. Lighthill's analogy is limited by the fact that it does not account for solid surfaces and does not correctly describe refraction effects. For these reasons extensions have been proposed:

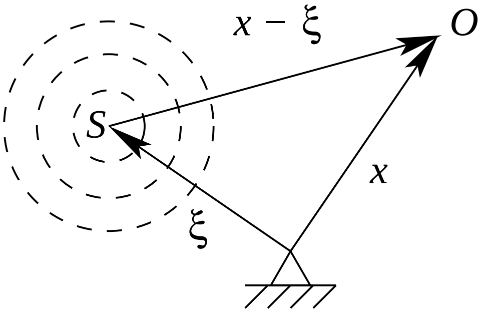


Figure 2.1: Coordinate system. O is the observer, S the source.

- Solid boundaries. Curle's analogy is a generalization of Lighthill's formulation for walls present in the flow.
- Moving, solid boundaries. The formulation of Ffowcs Williams and Hawkings accounts for reflections by solid, moving surfaces, making the theory applicable to arbitrarily moving immersed bodies.
- Extension to parallel shear flows due to Lilley and a modification by Goldstein [12].

These will be discussed in the next paragraphs.

2.1.2 Curle's acoustic analogy

Fixed, rigid bodies can produce sound by the interaction of the surface with the unsteady flow (e.g. turbulence). The presence of solid surfaces in the flow can be a significant sound source. In addition to the quadrupole volume sources exterior to the body, the acoustic field now also compromises a dipole surface distribution. Using again the free field Green function $G_0 = 1/(4\pi\|\mathbf{x}-\xi\|)\delta(t-\tau+\|\mathbf{x}-\xi\|)/c_0)$, the analogy becomes [25]:

$$\rho'(\mathbf{x},t) = \frac{\partial^2}{\partial x_i \partial x_j}\int\limits_V \frac{T_{ij}}{4\pi c_0^2\|\mathbf{x}-\xi\|}\mathrm{d}V - \frac{\partial}{\partial x_j}\int\limits_S \frac{p\delta_{ij}-\tau_{ij}}{4\pi c_0^2\|\mathbf{x}-\xi\|}n_i\mathrm{d}S, \tag{2.3}$$

both evaluated at the retarded time $\tau = t - \|\mathbf{x}-\xi\|/c_0$.[1] The first integral is Eq. (2.2). The second term, called "loading noise", represents the aerodynamic forces acting on the body.

---

[1] When the body acts like a compact source, i.e. $l/\lambda << 1$ with l as characteristic dimension of the body, the variation of the retarded time along the body may be neglected.

It is useful to know the relative importance of the volume and surface integrals. In Tab. 2.1 the scaling laws for the density fluctuation and the acoustic power intensity for a dipole and quadrupole source are listed. Their derivation can for example be found in [9]. For the expression of the acoustic power intensity the relations $I = \frac{\overline{p'^2}}{\rho_0 c_0}$ (planar wave, far field) and $p' = c_0^2 \rho'$ were used. The overbar denotes time averaging. The derivation of the quadrupolar scaling was originally done for a turbulent jet flow (replacing u_0 with the jet velocity and l for the jet diameter), but holds for unbounded turbulent flows in general. Entropy noise and the viscous stress contribution were neglected. For the dipole scaling a compact body was assumed.

Variable	**Dipole**	**Quadrupole**
$\rho'(\mathbf{x}, t)$	$\sim \frac{\rho_0}{c_0^3 \lVert \mathbf{x} - \xi \rVert} l u_0^3$	$\sim \frac{1}{c_0^4 \lVert \mathbf{x} - \xi \rVert} \left(\frac{u_0}{l}\right)^2 l^3 \rho_0 u_0^2$
I	$\sim \frac{\rho_0}{\lVert \mathbf{x} - \xi \rVert^2 c_0^3} l^2 u_0^6$	$\sim \frac{\rho_0}{c_0^5 \lVert \mathbf{x} - \xi \rVert^2} l^2 u_0^8$

Table 2.1: Scaling laws.

It can be seen that for a quadrupole source the acoustic intensity scales with the 8th power of velocity, while the acoustic intensity scales for a dipole distribution with the 6th power of velocity. For low Mach numbers, $M << 1$, the dipole term is thus expected to be dominant. If the body is non-compact (e.g. a long flat plate with a trailing edge located in a turbulent flow) the aerodynamic noise generation efficiency increases. The acoustic intensity then scales with the 5th power of velocity [9].

Curle's equation has been used for example for the sound prediction of a two-dimensional circular cylinder in a uniform flow [15] and the trailing edge of a NACA 0012 airfoil [34]. For both low Mach number cases the far-field was largely determined by the surface pressure dipoles. If the volume integral may be neglected, the computational cost and data storage can be reduced significantly.

2.1.3 Ffowcs Williams-Hawkings formulation

The integration method of Ffowcs Williams and Hawkings (FW-H) is the most general form of Lighthill's analogy. It has been used for e.g. helicopter rotor and propeller calculations. The idea of FW-H is to partition the domain into regions that are divided by a hypothetical surface on which the modeled problem exactly corresponds to the physical situation. The original domain is partitioned in a region inside the body and a region outside the body. The fluid motion on and outside the $f = 0$ surface is kept identical to the original problem. The inner solid domain is replaced by a fluid and given an arbitrary description, i.e. the freestream values $p = 0$, $\mathbf{u} = 0$ and $\rho = \rho_0$. One now attempts to construct conservation laws for the *entire* domain using the concept of generalized functions, here denoted by a tilde

$$\tilde{\rho} = \begin{cases} \rho & f > 0 \\ \rho_0 & f < 0 \end{cases} \tag{2.4}$$

See Fig. 2.2 for clarification of the domains. Similarly, the generalized stress tensor is $p\delta_{ij} - \tau_{ij}$ for $f > 0$ and zero otherwise, and $\tilde{\rho}_i = \rho u_i$ if $f > 0$ and zero for $f < 0$. Obviously, this will result in a mismatch with the exterior flow at the $f = 0$ boundary. To correct for this, discontinuous source terms on the right-hand side of the continuity and momentum equations appear, which will act as acoustic sources. In this way the exterior field "feels" the same effect as when the interior domain would be the solid body. After some manipulations, the following inhomogeneous wave equation is found [10]:

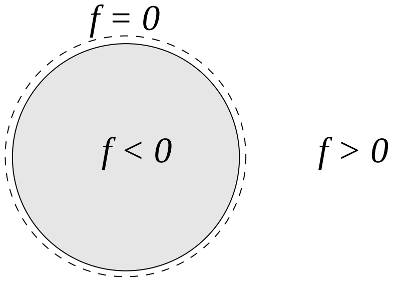


Figure 2.2: Classical FW-H. The control surface $f = 0$ is at the solid body.

$$\Box^2 p'(\mathbf{x}, t) = \frac{\partial^2}{\partial x_i \partial x_j} T_{ij} - \frac{\partial}{\partial x_i}\Big((p\delta_{ij} - \tau_{ij}) n_j \delta(f)\Big) + \frac{\partial}{\partial t}\Big(\rho_0 v_n \delta(f)\Big), \tag{2.5}$$

where the unit vector directed from the body is denoted by n_j and $\delta(f)$ is the Dirac delta function, which is only non-zero on the $f = 0$ surface. Eq. (2.5) is the classical formulation as derived by Ffowcs Williams and Hawkings. On the left hand-side a quadrupole, dipole and monopole (which represents volume displacement via the local normal velocity v_n of the surface) source can be seen. If the solid surface is stationary, the monopole source (for which the radiated density fluctuations scale with u^2) disappears and Eq. (2.5) reduces to Curle's formulation.

2.1.4 Linearized Euler equations

The linearized Euler equations (LEE) are known to be able to correctly account for convection effects, an inhomogeneous medium and refraction by a non-uniform flow. The primary assumption of the LEE is that the instantaneous variables can be decomposed into a time-averaged mean quantity and a small unsteady part:

$$\rho = \rho_0 + \rho', \quad u_i = u_{0i} + u_i', \quad p = p_0 + p'. \tag{2.6}$$

To obtain a system in terms of the fluctuating variables, the above decomposition is inserted into the Euler equations (viscous effects have a negligible influence on wave propagation), from which the steady mean flow equations are then subtracted. To linearize the system, the second order terms (multiplications of first order fluctuations) are neglected.

These are much smaller than first order fluctuations. This leads to the LEE [7]:

$$\frac{\partial \rho'}{\partial t} + \frac{\partial}{\partial x_j}(\rho' u_{0j} + \rho_0 u'_j) = S_c \tag{2.7a}$$

$$\frac{\partial \rho_0 u'_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho_0 u'_i u_{0j} + p'\delta_{ij}) + \frac{\partial u_{0i}}{\partial x_j}(\rho_0 u'_j + \rho' u_{0j}) = S_{m,i} \tag{2.7b}$$

$$\frac{\partial p'}{\partial t} + \frac{\partial}{\partial x_i}(\gamma p_0 u'_i + p' u_{0i}) + (\gamma - 1)(p' \frac{\partial u_{0i}}{\partial x_i} - u'_i \frac{\partial p_0}{\partial x_i}) = S_e \tag{2.7c}$$

A more complete motivation and derivation of the above equation set may be found in Appendix B. In the present work isentropic flow is considered, i.e. $\rho' = \frac{p'}{c^2} = \frac{\rho_0}{\gamma p_0} p'$. Because of this assumption, the density equation becomes unnecessary. If there are no large mean density jumps, we can normalize the pressure with respect to the density, making the system consistent with the incompressible solvers of OpenFOAM. Eqs. (2.7) then reduce to the following:

$$\frac{\partial u'_i}{\partial t} + \frac{\partial}{\partial x_j}(u'_i u_{0j} + p'\delta_{ij}) + \frac{\partial u_{0i}}{\partial x_j}(u'_j + \frac{p'}{\gamma p_0} u_{0j}) = S_{m,i} \tag{2.8a}$$

$$\frac{\partial p'}{\partial t} + \frac{\partial}{\partial x_i}(\gamma p_0 u'_i + p' u_{0i}) + (\gamma - 1)(p' \frac{\partial u_{0i}}{\partial x_i} - u'_i \frac{\partial p_0}{\partial x_i}) = S_e \tag{2.8b}$$

The left-hand side of Eqs. (2.8) is a linear form that contains all propagation effects. In general, the system is excited by (non-linear) source terms on the right-hand, i.e. monopoles in the energy equation and dipole and quadrupole-like sources in the momentum equation. As source term that follows from CFD data, Bailly [2] uses on the right-hand side of the momentum equation:

$$S_{m,i} = -\left\{ \frac{\partial u'_i u'_j}{\partial x_j} - \overline{\frac{\partial u'_i u'_j}{\partial x_j}} \right\}. \tag{2.9}$$

It is custom to perform aero-acoustic benchmark cases on a non-dimensional basis. The variables in Eqs. (2.8) are non-dimensionalized as follows:

$$t^* = \frac{c_0 t}{l}, \quad x_i^* = \frac{x_i}{l}, \quad u'^* = \frac{u'_i}{c_0}, \quad u_i^* = \frac{u_i}{c_0}, \quad p'^* = \frac{p'}{\rho_0 c_0^2}, \quad p_0^* = \frac{p_0}{\rho_0 c_0^2}. \tag{2.10}$$

Here l is a typical length scale and c_0 the speed of sound. The system of Eqs. (2.8) keeps the same form, but now one may think of the variables as Mach numbers, non-dimensional pressures and non-dimensional time and length scales. The simulations reported in chapter 4 were all carried out on a non-dimensional basis and with an acoustic Courant number less than one, which is defined as:

$$\text{acoustic Courant No.} = \frac{(u + c_0)\Delta t}{\Delta x} = \frac{(1 + M)\Delta t}{\Delta x} \tag{2.11}$$

as opposed to the Courant number definition used for flow simulations:

$$\text{Courant No.} = \frac{u\Delta t}{\Delta x}. \tag{2.12}$$

It is interesting to see the relation between the LEE and other acoustic analogies. Suppose we have a parallel mean flow, i.e. $\bar{u} = f(y)$ and $\bar{v} = 0$. The density and speed of sound are a function of y only, while the mean pressure $\bar{p}$ must be constant, because the steady mean flow obeys Euler's equations. Moreover, assume that the isentropic relation holds. In order to formulate a single equation expressed in term of p' we use the convective operator $\bar{D}/\bar{D}t = \partial/\partial t + \bar{u}\partial/\partial x$. Applying this operator on the continuity equation and subtracting from this the divergence of the momentum equations gives:

$$\frac{1}{c^2}\frac{\bar{D}^2 p'}{\bar{D}t^2} - \nabla^2 p' - 2\bar{\rho}\frac{d\bar{u}}{dy}\frac{\partial v'}{\partial x} = -\nabla \cdot \mathbf{S}. \tag{2.13}$$

This is Philips' analogy. The "d" is used to emphasize that $\bar{u}$ is only a function of the transverse coordinate. The term with v' can be eliminated by applying the $\bar{D}/\bar{D}t$ on Eq. (2.13) and adding $2d\bar{u}/dy$ times the y-derivative of the momentum equation. In this way several terms then cancel each other out. The result is:

$$\frac{\bar{D}}{\bar{D}t}\Big(\frac{1}{c^2}\frac{\bar{D}^2 p'}{\bar{D}t^2} - \nabla^2 p'\Big) + 2\frac{d\bar{u}}{dy}\frac{\partial^2 p'}{\partial x \partial y} = \Gamma, \tag{2.14}$$

where Γ is the following source term:

$$\Gamma = -\frac{\bar{D}}{\bar{D}t}\nabla \cdot \mathbf{S} + 2\frac{d\bar{u}}{dy}\frac{\partial S_y}{\partial x}. \tag{2.15}$$

The left hand side of Eq. (2.14) is now identical to Lilley's third order non-linear wave equation, which has as source term [9]:

$$\Gamma = -\frac{\bar{D}}{\bar{D}t}\frac{\partial^2 \rho u_i' u_j'}{\partial x_i \partial x_j} - 2\frac{d\bar{u}}{dy}\frac{\partial^2 \rho u_2' u_j'}{\partial x \partial x_j}. \tag{2.16}$$

Thus for a unidirectional shearflow the LEE reduce to Lilley's analogy if the source term in the LEE is given by:

$$S_{m,i} = -\frac{\partial \rho u_i' u_j'}{\partial x_j}. \tag{2.17}$$

If the medium is stagnant, the LEE reduce to Lighthill's analogy.

2.2 Wave propagation errors

2.2.1 Basic concepts

In this section a few relations and concepts useful for wave propagation analysis are described. The two main types of errors are dissipation and dispersion. These can be quantified using the amplification factor, which is defined as the change of the solution during a single timestep:

$$G = \frac{u(x, t+\Delta t)}{u(x,t)}. \tag{2.18}$$

The dissipation error is then the ratio between the magnitudes of the numerical and exact amplification factors:

$$\epsilon_d = \frac{|G_{num}|}{|G_{exact}|}. \tag{2.19}$$

The dispersion error is given by the difference in phase:

$$\epsilon_\phi = \frac{\phi_{num}}{\phi_{exact}} \qquad \text{or} \qquad \epsilon_\phi = \phi_{num} - \phi_{exact}, \tag{2.20}$$

where

$$\phi = \tan^{-1}\left(\frac{\Im\{G\}}{\Re\{G\}}\right). \tag{2.21}$$

An often used quantity is the wavenumber k. This is a spatial frequency:

$$k = \frac{2\pi}{\lambda}. \tag{2.22}$$

The number of points per wavelength that is required to resolve a wave (given a certain criterion) is a measure of the resolution of a numerical scheme:

$$PPW = \frac{\lambda}{\Delta x}, \tag{2.23}$$

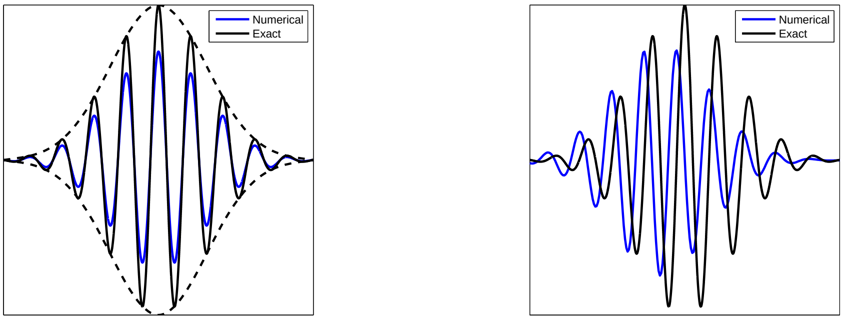
which means that PPW is related to k via

$$k\Delta x = \frac{2\pi}{PPW}. \tag{2.24}$$

The phase speed, v_p, is the speed at which the phase - e.g. the crest - propagates. During a time interval, a simple sine wave generates ωt waves, which travel over a distance kx. Thus $\omega t = kx$, leading to a velocity x/t:

$$v_p = \frac{\omega}{k}. \tag{2.25}$$

A wave can also be compromised of waves with different frequencies. The amplitudes of the internal waves then form an envelope, see Fig. 2.3. The speed of the amplitude envelope is called the group velocity. It is defined by:



(a) Dissipation.

(b) Dispersion.

Figure 2.3: Illustration of the dissipation and dispersion error. Left: Damping of the solution. Right: Crests propagate at the wrong (phase) velocity and the energy moves at the wrong (group) velocity (notice the change in shape of the envelope).

$$v_g \triangleq \frac{\partial \omega}{\partial k}. \tag{2.26}$$

The group speed v_g can be thought of as the energy propagating velocity. If the wave propagation is non-dispersive, the phase speed and group speed are the same. This is the case for the linear convection equation on differential level. However, on the discrete level this is generally not true. This is further discussed in the next section.

2.2.2 Numerical schemes

The majority of CAA studies uses high-order finite difference schemes [26, 27]. Although we will not use a finite difference method in this project, it is used here to show why high-order methods are often favored. In contrast to CFD where schemes opt to maximize the order of accuracy, in CAA schemes are optimized with respect to preserving wave properties. To illustrate this, consider the central discretization of a derivative:

$$\frac{\partial f}{\partial x}(x) \approx \frac{1}{\Delta x} \sum_{j=-N}^{N} a_j f(x + j\Delta x). \tag{2.27}$$

With a seven point stencil, a 6th order scheme can be constructed from Taylor series approximations, thus fixing all coefficients a_j. A different strategy is to opt for a fourth order scheme by fixing a_0, a_1 and a_2 ($a_{-1} = -a_1$, $a_{-2} = -a_2$) and leave a_3 as a free parameter. In order to take the wavespeed into account in the analysis, Eq. (2.27) can be Fourier transformed into wavenumber space, which after some manipulation yields a relation between the exact, physical wavenumber and the numerical wavenumber. The parameter a_3 can now be used as a design variable to minimize the difference between the exact and numerical wavenumbers. Tam and Webb [27] carried out such an analysis and developed dispersion-relation-preserving (DRP) schemes. These are one of the most popular choices in literature.

The group speed of a wave package that is compromised of different wavenumbers can shown to be equal to the derivative of the exact wavenumber with respect to the numerical wavenumber. If all the components travel with the same speed, this will be equal to one. Otherwise the wave package will spread out as it propagates. In Fig. 2.4 the relation between the exact and numerical wavenumber, respectively k and $\overline{k}$, is shown for a few schemes. It can be seen that for non-optimized schemes the high wavenumber waves travel at a quite different velocity from the physical velocity. This is known as a dispersion error. By increasing the stencil size, the error can be reduced. Alternatively, the standard central scheme can be used but with a small Δx.

2.3 Boundary conditions

In this section a brief description of two often used boundary conditions. These are the characteristic boundary condition and the bufferzone technique.

2.3.1 Characteristic boundary condition

Since a hyperbolic system propagates solutions in distinctive directions, it seems logical to use this property when constructing a non-reflective boundary condition. One of the early

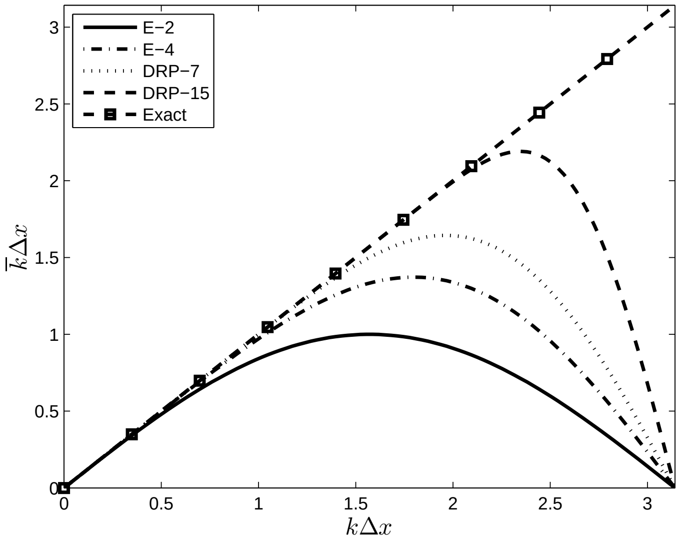


Figure 2.4: Illustration of the effect of scheme optimization. On the x-axis the exact, physical wavenumber is used and on the y-axis the numerical wavenumber. Shown are the classical, non-optimized central (E-2) and fourth order (E-4) schemes, 7 point fourth order optimized central DRP scheme (DRP-7) and 15 point optimized DRP scheme. Various other schemes lie between DRP-7 and DRP-15 and they follow the trend of a smaller dispersive error as the stencil size is increased. Note that waves move backwards for wavenumbers larger than the wavenumber corresponding to the maximum of a given scheme, i.e. the derivative is negative.

methods that does so was due to Thompson [28, 29]. His reasoning starts by separating the terms of a hyperbolic equation into those that are in the direction in which the boundary condition should work - say x - and those that are transverse to the boundary (y, z), i.e.:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} + \mathbf{C}, \tag{2.28}$$

where $\mathbf{A}$ contains terms like the velocity in the x-direction, but not in the y-direction, since these are gathered into $\mathbf{C}$. Diagonalizing the system yields

$$\mathbf{l_i^T}\frac{\partial \mathbf{U}}{\partial t} + \lambda_i \mathbf{l_i^T}\frac{\partial \mathbf{U}}{\partial x} + \mathbf{l_i^T}\mathbf{C}, \tag{2.29}$$

where $\mathbf{l_i^T}$ is the $\mathrm{i^{th}}$ eigenvector corresponding to the eigenvalue λ_i. Thompson defined

$$\mathcal{L}_i = \lambda_i \mathbf{l_i^T}\frac{\partial \mathbf{U}}{\partial x}. \tag{2.30}$$

A non-reflecting condition now follows:

$$\mathcal{L}_i = \begin{cases} \lambda_i \mathbf{l_i^T}\frac{\partial \mathbf{U}}{\partial x} & \text{for outgoing waves,} \\ 0 & \text{for incoming waves.} \end{cases} \tag{2.31}$$

A full discussion on how this can be applied to the Navier-Stokes equations is beyond the scope of the present text, but detailed descriptions can be found in [23, 28, 29].

Basically, 1D problems are considered by lumping transverse components together (which are optionally dropped, see Poinsot and Lele [23] who call this a Local One-Dimensional Inviscid problem, LODI) and then matching characteristics at the boundary. The specific values addressed to the $\mathcal{L}_i$'s then depend on the type of boundary condition, see the list Thompson provides in [29]. The `waveTransmissive` boundary condition that is used in the validation study is said to be based on the LODI, although this is rather hard to see from the code.

The `convectiveOutlet` boundary condition does something similar. It applies an unsteady convective condition at the boundaries by calculating on the boundaryfield:

$$\frac{d\phi}{dt} + U_n \frac{d\phi}{dn} = 0, \tag{2.32}$$

where ϕ is any dependent variable (ρ', p', $\mathbf{u}'$) and U_n the normal component of the convective velocity, in this case the speed of sound. This boundary condition is used in most of the verification cases of chapter 4.

2.3.2 Buffer zone

Instead of letting waves pass through the boundary without reflection, the buffer zone technique attempts to resolve the waves as they propagate to the boundaries. Adjacent to the domain, an absorbing layer is added where the flow physics are altered. The technique modifies the governing equations in the exit zone by adding a damping term to the right-hand side:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} = -\sigma(x)(\mathbf{U} - \mathbf{U}_{\text{target}}), \tag{2.33}$$

where $\mathbf{U}$ is the state vector, $\mathbf{U}_{\text{target}}$ the target (in this case zero) and σ a damping function:

$$\sigma(x) = \begin{cases} \sigma_{max}\left(\frac{x-x_0}{w}\right)^m & \text{if } x \geq x_0, \\ 0 & \text{if } x < x_0. \end{cases} \tag{2.34}$$

Here w is the width of the bufferzone, m an exponent that determines the profile shape and x_0 the location where the bufferzone starts. Adding the artificial damping term has a similar effect as cranking up the viscosity by adding $\mu(x)\frac{d^2\mathbf{U}}{dx^2}$ to the left-hand side.

Chapter 3

The Riemann solver

In this chapter the theoretical basis and practical implementation of the Riemann solver is described. First some concepts are briefly described in section 3.1, after which the approach is applied to our system in section 3.2. The chapter is concluded with an outline of the implementation in OpenFOAM.

3.1 The Riemann problem for linear hyperbolic equations

An excellent treatment on Riemann solvers is given in the books by Leveque [20] and Toro [30]. Here we will not attempt to give a complete presentation similar to theirs, but instead outline some basic theory by considering an example problem. Since Eqs. (2.8) form a linear hyperbolic system, we restrict our discussion to such systems. For a linear problem a Riemann solver is the equivalent of a characteristic solver and an analytical approach is possible.

Consider the following:

$$\mathbf{q}_t + \mathbf{A}\mathbf{q}_x = \mathbf{0}. \tag{3.1}$$

When $\mathbf{A}$ is a constant coefficient matrix, one may find that the above problem comes down to p decoupled advection equations expressed in a transformed variable:

$$\mathbf{w}_t^p + \lambda^p \mathbf{w}_x^p = 0, \tag{3.2}$$

where $\mathbf{w} = \mathbf{R}^{-1}\mathbf{q}$, with $\mathbf{R}$ as the right eigenvector matrix of the diagonal matrix that corresponds to $\mathbf{A}$. A Riemann problem is a special initial value problem where the initial states are given by:

$$\mathbf{w}_{t=0}^p(x) = \begin{cases} \mathbf{w}_l^p & \text{if } x < 0, \\ \mathbf{w}_r^p & \text{if } x > 0. \end{cases}$$

The solution at (x,t) depends on the initial data evaluated at the characteristic $x - \lambda^p t$, i.e. $\mathbf{w}^p(x,t) = \mathbf{w}_{t=0}^p(x - \lambda^p t)$:

$$\mathbf{w}^p(x,t) = \begin{cases} \mathbf{w}_l^p & \text{if } x - \lambda^p t < 0, \\ \mathbf{w}_r^p & \text{if } x - \lambda^p t > 0. \end{cases}$$

The solution to the Riemann problem is a superposition of p waves travelling with different speeds λ^p

$$\mathbf{q}(x,t) = \sum_{p:\lambda^p > \frac{x}{t}}^{m} \mathbf{w}_l^p \mathbf{r}^p + \sum_{p:\lambda^p < \frac{x}{t}}^{m} \mathbf{w}_r^p \mathbf{r}^p. \tag{3.3}$$

This is illustrated in Fig. 3.1. The tracing of characteristics to find the solution can be expressed conveniently in terms of jump relations:

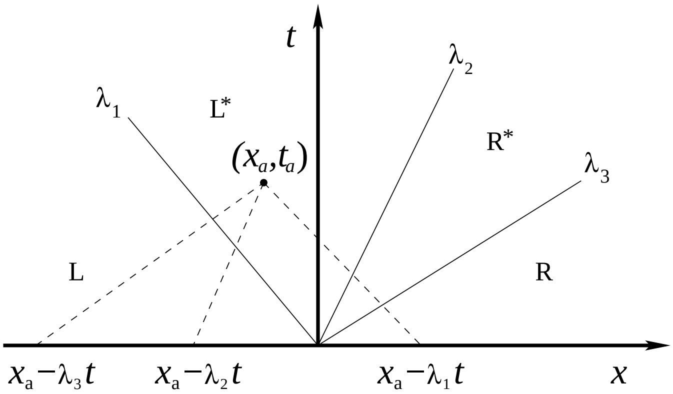


Figure 3.1: Solution construction at point x_a at time level t_a. In this case the solution is $\mathbf{q}(x_a, t_a) = \mathbf{w}_r^1\mathbf{r}^1 + \mathbf{w}_l^2\mathbf{r}^2 + \mathbf{w}_l^3\mathbf{r}^3$. After LeVeque [20].

$$\begin{aligned}
\mathcal{W}^1 &= \mathbf{q}_l^* - \mathbf{q}_l = (\mathbf{w}_r^1 - \mathbf{w}_l^1)\mathbf{r}^1 \\
\mathcal{W}^2 &= \mathbf{q}_r^* - \mathbf{q}_l^* = (\mathbf{w}_r^2 - \mathbf{w}_l^2)\mathbf{r}^2 \\
\mathcal{W}^3 &= \mathbf{q}_r^* - \mathbf{q}_r = -(\mathbf{w}_r^3 - \mathbf{w}_l^3)\mathbf{r}^3
\end{aligned}$$

after which the state at any point follows from

$$\mathbf{q}(x,t) = \mathbf{q}_l + \sum_{p:\lambda^p < \frac{x}{t}}^{m} \mathcal{W}^p \tag{3.4}$$

$$= \mathbf{q}_r - \sum_{p:\lambda^p > \frac{x}{t}}^{m} \mathcal{W}^p. \tag{3.5}$$

Eq. (3.4) may be interpreted as the left state plus the change due to leftwards moving waves. The solution at the face $(x = 0)$ is often written as:

$$\mathbf{q}(0,t) = \frac{1}{2}(\mathbf{q}_l + \mathbf{q}_r) + \frac{1}{2}\Big(\sum_{p:\lambda^p<0}^{m} \mathcal{W}^p - \sum_{p:\lambda^p>0}^{m} \mathcal{W}^p \Big), \tag{3.6}$$

Since $\mathbf{A}$ is a constant coefficient matrix, the flux is simply $\mathbf{F}_{face} = \mathbf{A}\mathbf{q}$. Combining this with Eq. (3.4) and Eq. (3.5) yields

$$\begin{aligned}
\mathbf{F}_{face} &= \mathbf{A}\mathbf{q}_l + \sum_{p:\lambda^p<0}^{m} \mathbf{A}\mathcal{W}^p \\
&= \mathbf{A}\mathbf{q}_l + \sum_{p:\lambda^p<0}^{m} \lambda^p \mathcal{W}^p \qquad (3.7)
\end{aligned}$$

$$
\begin{aligned}
\mathbf{F}_{face} &= \mathbf{A}\mathbf{q}_r - \sum_{p:\lambda^p>0}^{m} \mathbf{A}\mathcal{W}^p \\
&= \mathbf{A}\mathbf{q}_r - \sum_{p:\lambda^p>0}^{m} \lambda^p \mathcal{W}^p.
\end{aligned} \tag{3.8}
$$

Here we used that $\mathcal{W}$ contains the right eigenvectors of $\mathbf{A}$, i.e. $\mathbf{AR} = \mathbf{\Lambda R}$. For implementation purposes, the most convenient formulation is

$$
\begin{aligned}
\mathbf{F}_{face} &= \frac{1}{2}\mathbf{A}(\mathbf{q}_l + \mathbf{q}_r) + \frac{1}{2}\sum_{p=1}^{m} \Big[(\lambda^p)^- - (\lambda^p)^+\Big]\mathcal{W}^p \\
&= \frac{1}{2}\mathbf{A}(\mathbf{q}_l + \mathbf{q}_r) - \frac{1}{2}\sum_{p=1}^{m} |\lambda^p|\mathcal{W}^p.
\end{aligned} \tag{3.9}
$$

Note that

$$
\lambda^+ = \max(\lambda, 0), \qquad \lambda^- = \min(\lambda, 0). \tag{3.10}
$$

3.2 The Riemann based LEE solver

The LEE given by Eqs. (2.8) can be written as:

$$
\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{Aq}}{\partial x} + \frac{\partial \mathbf{Bq}}{\partial y} + \frac{\partial \mathbf{Cq}}{\partial z} + \mathbf{H} = \mathbf{S}, \tag{3.11}
$$

with state vector

$$
\mathbf{q} = \begin{bmatrix} u' \\ v' \\ w' \\ p' \end{bmatrix}. \tag{3.12}
$$

$\mathbf{H}$ contains part of the gradients in the mean flow terms and is not used in the derivation of the Riemann solver. Since OpenFOAM can use unstructured grids, the Jacobian $\mathbf{J} = \mathbf{A}n_x + \mathbf{B}n_y + \mathbf{C}n_z$ is introduced:

$$
\frac{\partial \mathbf{Aq}}{\partial x} + \frac{\partial \mathbf{Bq}}{\partial y} + \frac{\partial \mathbf{Cq}}{\partial z} = \frac{\partial (\mathbf{Jq})_i}{\partial x_i}. \tag{3.13}
$$

So:

$$
\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial (\mathbf{Jq})_i}{\partial x_i} + \mathbf{H} = \mathbf{S} \tag{3.14}
$$

We will focus on $\mathbf{J}$ now, since this is the only part that is considered in the Riemann solver. We find:

$$
\mathbf{J} = \begin{bmatrix} U & 0 & 0 & n_x \\ 0 & U & 0 & n_y \\ 0 & 0 & U & n_z \\ \gamma p_0 n_x & \gamma p_0 n_y & \gamma p_0 n_z & U \end{bmatrix}, \tag{3.15}
$$

where $U = u_0 \cdot n_x + v_0 \cdot n_y + w_0 \cdot n_z$. The eigenvalues follow from:

$$
\det(\mathbf{J} - \lambda \mathbf{I}) = \mathbf{0}. \tag{3.16}
$$

This yields

$$\lambda_{1,2} = U, \qquad \lambda_3 = U + \sqrt{\gamma p_0}, \qquad \lambda_4 = U - \sqrt{\gamma p_0}. \tag{3.17}$$

Recall that in OpenFOAM pressure is divided by density, meaning that $\lambda_{3,4} = U \pm c$, where c is the speed of sound. Four different states can be distinguished, as shown in Fig. 3.2. The corresponding right eigenvectors are[1]:

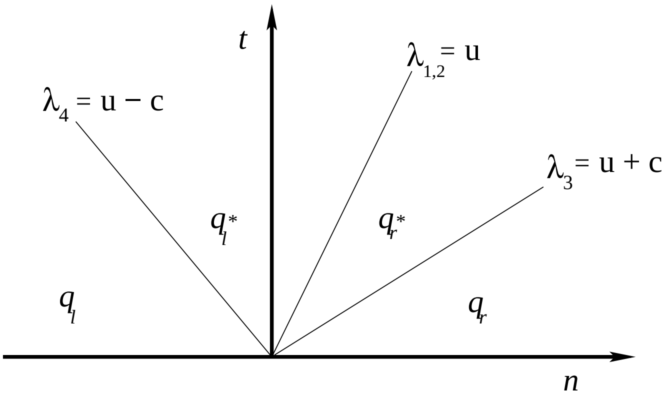


Figure 3.2: Characteristic waves.

$$\mathbf{r}^1 = \begin{bmatrix} 1 \\ 0 \\ -n_x/n_z \\ 0 \end{bmatrix}, \quad \mathbf{r}^2 = \begin{bmatrix} 0 \\ 1 \\ -n_y/n_z \\ 0 \end{bmatrix}, \quad \mathbf{r}^3 = \begin{bmatrix} n_x \\ n_y \\ n_z \\ c_r \end{bmatrix}, \quad \mathbf{r}^4 = \begin{bmatrix} n_x \\ n_y \\ n_z \\ -c_l \end{bmatrix}. \tag{3.18}$$

The right eigenvector matrix is thus:

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & n_x & n_x \\ 0 & 1 & n_y & n_y \\ -n_x/n_z & -n_y/n_z & n_z & n_z \\ 0 & 0 & c_r & -c_l \end{bmatrix} \tag{3.19}$$

and its inverse

$$\mathbf{L} = \frac{1}{c_l + c_r} \begin{bmatrix} (c_l + c_r)(n_y^2 + n_z^2) & -(c_l + c_r)n_x n_y & (c_l + c_r)n_x n_z & 0 \\ -(c_l + c_r)n_x n_y & (c_l + c_r)(n_x^2 + n_z^2) & -(c_l + c_r)n_y n_z & 0 \\ c_l n_x & c_l n_y & c_l n_z & 1 \\ c_r n_x & c_r n_y & c_r n_z & -1 \end{bmatrix}. \tag{3.20}$$

The characteristic variable $\mathbf{w} = \mathbf{L}\mathbf{q}$ is now introduced

$$\mathbf{w} = \begin{bmatrix} u' - n_x U' \\ v' - n_y U' \\ \frac{c_l}{c_l + c_r} U' + \frac{1}{c_l + c_r} p' \\ \frac{c_r}{c_l + c_r} U' - \frac{1}{c_l + c_r} p' \end{bmatrix}. \tag{3.21}$$

---

[1] In the eigenvectors $\mathbf{r}^1$ and $\mathbf{r}^2$ a division by n_z is made. A possible difficulty of division by zero is however later canceled out.

The wave strengths are:

$$\alpha = \mathbf{w}_r - \mathbf{w}_l. \tag{3.22}$$

The p^{th} jump in $\mathbf{q}$ across the p^{th} wave is denoted by

$$\mathcal{W}^p = \alpha^p \mathbf{r}^p. \tag{3.23}$$

This leads to the following wave vectors:

$$\begin{aligned}
\mathcal{W}^1 + \mathcal{W}^2 &= \mathbf{q}_r^* - \mathbf{q}_l^* \\
&= \alpha^1 \mathbf{r}^1 + \alpha^2 \mathbf{r}^2 \\
&= \Big((u_r' - u_l') - n_x(U_r' - U_l')\Big) \begin{bmatrix} 1 \\ 0 \\ -n_x/n_z \\ 0 \end{bmatrix} + \Big((v_r' - v_l') - n_y(U_r' - U_l')\Big) \begin{bmatrix} 0 \\ 1 \\ -n_y/n_z \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} (u_r' - u_l') - n_x(U_r' - U_l') \\ (v_r' - v_l') - n_y(U_r' - U_l') \\ (w_r' - w_l') - n_z(U_r' - U_l') \\ 0 \end{bmatrix},
\end{aligned} \tag{3.24}$$

$$\begin{aligned}
\mathcal{W}^3 &= \mathbf{q}_r - \mathbf{q}_r^* \\
&= \alpha^3 \mathbf{r}^3 \\
&= \begin{bmatrix} \frac{c_l n_x}{c_l+c_r}(U_r' - U_l') + \frac{n_x}{c_l+c_r}(p_r' - p_l') \\ \frac{c_l n_y}{c_l+c_r}(U_r' - U_l') + \frac{n_y}{c_l+c_r}(p_r' - p_l') \\ \frac{c_l n_z}{c_l+c_r}(U_r' - U_l') + \frac{n_z}{c_l+c_r}(p_r' - p_l') \\ \frac{c_l c_r}{c_l+c_r}(U_r' - U_l') + \frac{c_r}{c_l+c_r}(p_r' - p_l') \end{bmatrix},
\end{aligned} \tag{3.25}$$

$$\begin{aligned}
\mathcal{W}^4 &= \mathbf{q}_l^* - \mathbf{q}_l \\
&= \alpha^4 \mathbf{r}^4 \\
&= \begin{bmatrix} \frac{c_r n_x}{c_l+c_r}(U_r' - U_l') - \frac{n_x}{c_l+c_r}(p_r' - p_l') \\ \frac{c_r n_y}{c_l+c_r}(U_r' - U_l') - \frac{n_y}{c_l+c_r}(p_r' - p_l') \\ \frac{c_r n_z}{c_l+c_r}(U_r' - U_l') - \frac{n_z}{c_l+c_r}(p_r' - p_l') \\ -\frac{c_r c_l}{c_l+c_r}(U_r' - U_l') + \frac{c_l}{c_l+c_r}(p_r' - p_l') \end{bmatrix}.
\end{aligned} \tag{3.26}$$

The wavevectors are slightly different when mean density variations are important, see Appendix C. The flux at the face now follows from:

$$\mathbf{F}_{face} = \frac{1}{2}(\mathbf{F}_r + \mathbf{F}_l) - \frac{1}{2}\sum_{p=1}^{m} |\lambda^p| \mathcal{W}^p. \tag{3.27}$$

Eq. (3.27) is Godunov's first-order upwind method. To make the solver less diffusive, a higher-order extension is needed. This is done by extrapolating the state variables from the cell centers towards the face in a limited linear fashion. On unstructured grids, the Barth-Jespersen slope limiter is suitable for this purpose [3].

3.3 Implementation in OpenFOAM

OpenFOAM is an open-source `C++` library. Since the code is open-source, much solvers and template files are readily available and can be used as a basis for further development.

In the present work, the code uses roughly the following steps. First the Riemann problem is set up by looping over all cells. Each of these cells (called "owners") is connected to multiple neigbouring cells via a common face. For each such face of an owner cell, the variables of the current cell and a neighbouring cell are passed to a separate code file in which the Riemann problem is solved in accordance to the description of Section 3.2. After calculating the face fluxes, they are passed to the main acoustic solver where an explicit 4-stage Runge-Kutta time integration algorithm is used to march in time. This implementation will from hereon referred to as aeroacousticFoam or "aacFoam" for short. In Fig. 3.3 a visualization of the code is shown.

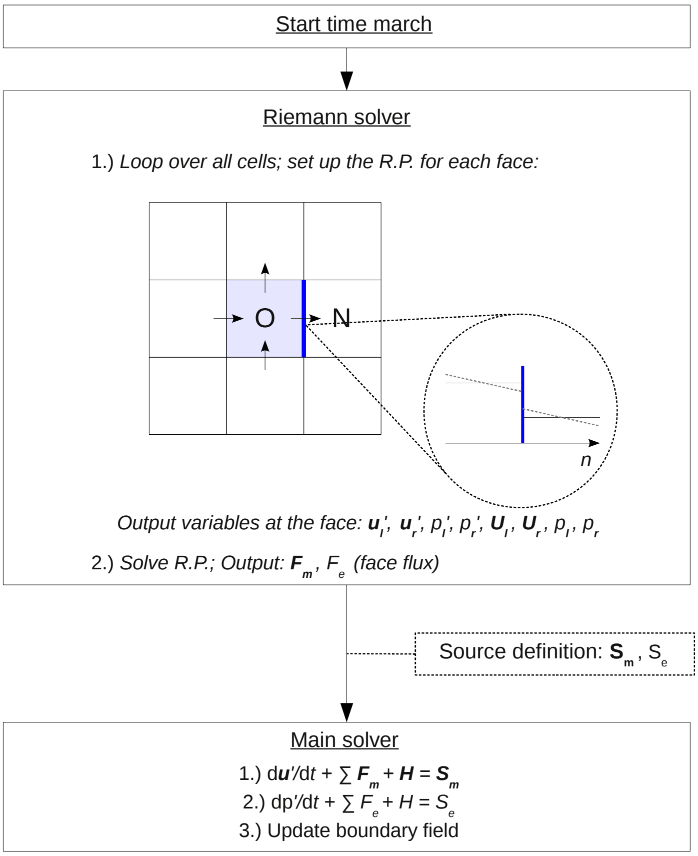

Start time march
Riemann solver
1.) Loop over all cells; set up the R.P. for each face:
O
N
n
Output variables at the face: $\boldsymbol{u}_l'$, $\boldsymbol{u}_r'$, p_l', p_r', $\boldsymbol{U}_l$, $\boldsymbol{U}_r$, p_l, p_r
2.) Solve R.P.; Output: $\boldsymbol{F}_m$, F_e (face flux)
Source definition: $\mathbf{S}_m$, S_e
Main solver
1.) $d\boldsymbol{u}'/dt + \sum \boldsymbol{F}_m + \boldsymbol{H} = \boldsymbol{S}_m$
2.) $dp'/dt + \sum F_e + H = S_e$
3.) Update boundary field

Figure 3.3: Code visualization.

In the present work three other schemes are used and therefore given for reference. For a detailed description, see the doctoral thesis of Jasak [16].

In a finite volume method, terms in a partial differential equation are converted to integrals over a volume. To facilitate the numerical evaluation of, for example, convection of a variable ϕ by a velocity $\mathbf{U}$, one would write:

$$\int\limits_V \nabla \cdot (\mathbf{U}\phi)\mathrm{d}V = \sum_{faces} \mathbf{U}_f \phi_f \mathbf{S_f}. \tag{3.28}$$

Since our variables are defined at the cell center of a volume, a certain (interpolation) strategy is needed to determine the value at the cell's faces. The situation is sketched in Fig. 3.4.

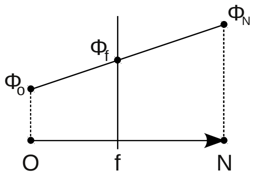


Figure 3.4: Face interpolation. O = owner, f = face, N = neighbour.

With the central scheme the face value is determined according to:

$$\phi_f = f_x \phi_O + (1 - f_x)\phi_N, \tag{3.29}$$

where $f_x = \frac{\overline{fN}}{\overline{ON}}$ is a ratio of distances.

The linear upwind scheme is given by:

$$\phi_f = \begin{cases} \phi_O + (x_f - x_O)(\nabla\phi)_O & \text{if } \mathbf{S} \cdot \mathbf{U}_f \geq 0, \\ \phi_N + (x_f - x_N)(\nabla\phi)_N & \text{if } \mathbf{S} \cdot \mathbf{U}_f < 0, \end{cases} \tag{3.30}$$

with $(\nabla\phi)_N = \frac{1}{V_N} \sum_{faces} \phi_f \mathbf{S}$ and $\mathbf{S}$ again being the surface.

The limited linear scheme consists of a lower order scheme (upwind) and a higer order correction limited by a Sweby limiter:

$$\phi_f = (\phi)_{LO} + \Psi[(\phi)_{HO} - (\phi)_{LO}], \tag{3.31}$$

where $(\phi)_{HO}$ is the face value of ϕ evaluated with the higher order scheme and Ψ is the flux limiter.

Chapter 4

Verification

In this chapter the aim is to compare the performance of the standard and Riemann-based solver and to verify the order and the dissipative and dispersive character of the implementation.

First these properties of the numerical schemes are investigated using a simple wave convection problem. Next the solver is verified using benchmark studies for which analytical solutions are available. For these problems the source data is known and thus no flow calculation is needed. The cases that are investigated are: the propagation of an acoustic pulse, the fundamental acoustic sources, acoustic scattering from a wall and source radiation in non-uniform flow.

4.1 One-dimensional wave propagation

The dissipative and dispersive properties of OpenFOAM's built-in numerical schemes and the implemented method can be assessed with an initial value problem for the linear advection equation:

$$\frac{dp'}{dt} + c\frac{dp'}{dx} = 0, \tag{4.1}$$

where $p'(x,0) = f(x)$ is an initial disturbance that is convected with a non-dimensional speed c, which is set equal to 1. The solution at later times is $p' = f(x-t)$. We consider the following disturbances:

- A Gaussian function:
$$f(x) = e^{-(\ln(2)\frac{x}{3})^2}. \tag{4.2}$$
- A sine function:
$$f(x) = \sin(2\pi x/\lambda), \qquad 0 \leq x \leq \lambda. \tag{4.3}$$

  The wavelength is set equal to 10.

Hence both a symmetric aperiodic signal and a periodic signal is tested. A domain of non-dimensional length 200 is selected, so that at the end of the simulation the disturbance is still far enough away from the boundary and thus no boundary influence is noticed.

To evaluate the $\frac{dp'}{dx}$ term we have selected the standard Gauss linear and Gauss limitedLinear methods to compare with the new implementation. In this case the "Riemann" scheme should reduce the code to a second order reconstruction (with limiter) of the left and right variables at the face, after which the upwind value is taken to determine the flux. Gauss linear is a second order central discretization, which is fine as long as it does not results in any numerical instability. The limitedLinear scheme is a TVNI scheme that combines upwind with central, see Sec. 3.3.

The pulse is initially located at $x = 0$. For each of the schemes we vary the number of points per wavelength - or in the case of the pulse, the number of points per pulse-width - as follows: 10, 12, 15, 20 and 25. From initial tests it was found that the temporal error is eliminated when the Courant number is lower than 0.1, hence the timestep is set according to this criterion. In Fig. 4.1 the convection of the pulse is shown for the three schemes at three time instances.

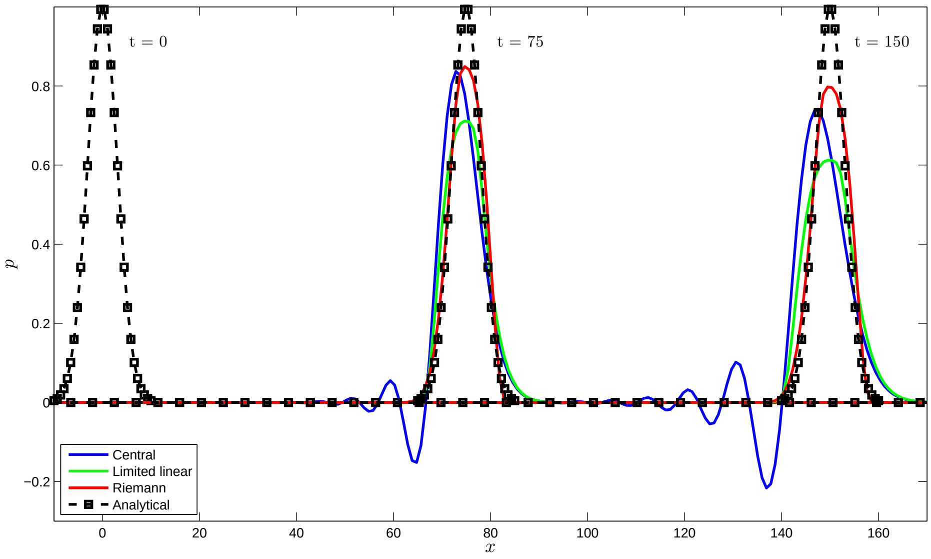


Figure 4.1: Convection of a pulse. Three instances are shown: $t = 0$, $t = 75$ and $t = 150$. PPW = 20.

As can be seen, large trailing waves are formed behind the pulse when the central scheme is used. This is a dispersion error, which originates from a variation in the group velocity inherit to the scheme. From Fig. 2.4 it was found that waves with a large wavelength (low wavenumber) travel at the same group speed. At a certain higher wavenumber, the group velocity for a scheme becomes less than one. Thus, shorter waves will propagate at a lower speed than the long waves. The spatial Fourier transform of a Gaussian pulse contains mainly low frequency content, but it is the high frequency part that forms the observed trailing waves. The waves are absent when the scheme limitedLinear is used.

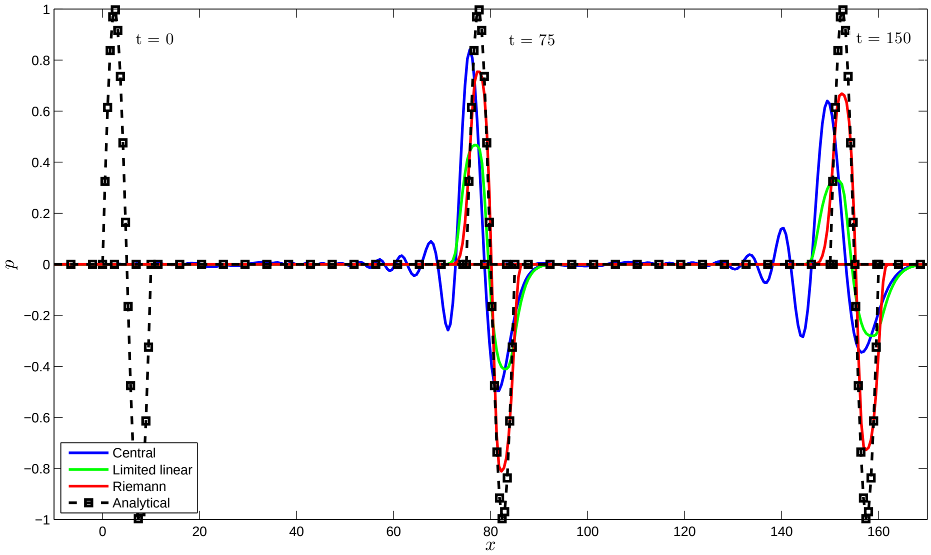


Figure 4.2: Convection of a sinusoidal function. Three instances are shown: $t = 0$, $t = 75$ and $t = 150$. PPW = 20.

This scheme is however too diffusive. The Riemann method adds a moderate amount of diffusion, while preserving the dispersion relation. No oscillations are present, not even on coarse meshes. On coarse meshes the diffusion error becomes increasingly apparent for the Riemann scheme. A similar behaviour can be seen from the sinusoid testcase, see Fig. 4.2. At the beginning and the end of the sinusoid a discontinuity occurs. This gives a lot of high frequency waves to represent the discontinuity. Another explanation: When the sinusoid is properly resolved, the spatial Fourier transform will yield a two-sided frequency spectrum of two Dirac pulses. When the wave becomes under-resolved, the frequency spectrum will contain spurious waves that form trailing waves.

Case	Scheme	Order
Pulse	Central	1.1 (1.2)
	Limited linear	1.1 (1.3)
	Riemann	1.9 (1.8)
Sine	Central	0.8 (0.8)
	Limited linear	0.8 (0.9)
	Riemann	2.0 (1.8)

Table 4.1: Order of the tested schemes for the two cases. Since there are slight kinks in the lines on the interval 10 PPW - 25 PPW, the order was calculated on the straight part 12 PPW - 20 PPW. In brackets the order according to the slope from 10 PPW - 25 PPW is given.

In Fig. 4.3 log-log plots of the root-mean-square error are shown. The order of the methods are extracted from these figures and summarized in Tab. 4.1.

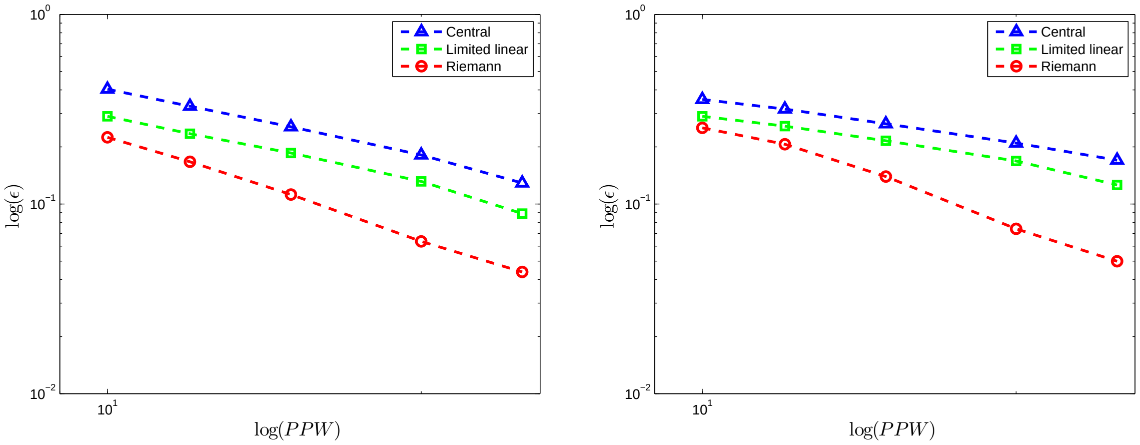


Figure 4.3: loglog plot for the pulse case (left) and sine function case (right).

The influence of disabling the Barth-Jespersen limiter is shown in Fig. 4.4. Clearly, the solution deteriorates. Lastly, we numerically determine the exact k vs numerical k plot for the Riemann scheme, as was shown for other schemes in Fig. 2.4. The set-up is similar to the previous two cases. We give an initial sinusoidal signal of 7 wavelengths long. Since the convection speed is equal to unity, the period T is equal to the wavelength λ. The dispersion error has a local nature, i.e. it is generated over a single wavelength. Thus, we may sample the time history at a fixed point in space, e.g. $x = 5\lambda$ and then the period of the first wave that passes this observation point is equal to the numerical wavelength. For a fixed mesh the wavelength is then varied from $\lambda = 20$ to $\lambda = 4$. The result of this approach is shown in Fig. 4.5 for both the central scheme and the Riemann scheme. The numerically determined central scheme is given to show the accuracy of the approach. The numerically determined data points (fitted by a cubic interpolant) for the central scheme are very close to the analytical line when $\lambda \leq 4$ ($k\Delta x \leq \pi$). For lower PPW

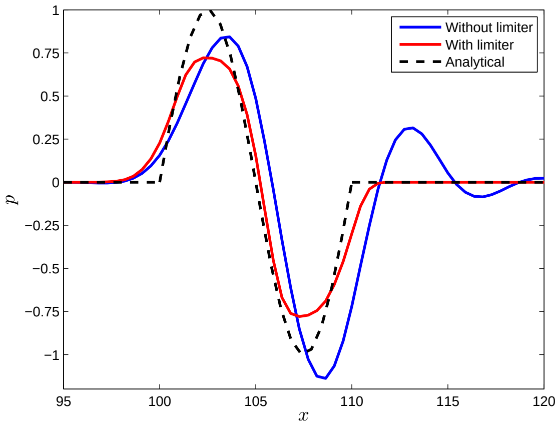


Figure 4.4: Comparison of the solution obtained with and without Barth-Jepsersen limiter at $t = 100$.

resolutions, it is no longer possible to determine the numerical wavelength accurately. The obtained line for the Riemann scheme is very close to the analytical relation for the fourth order central scheme.

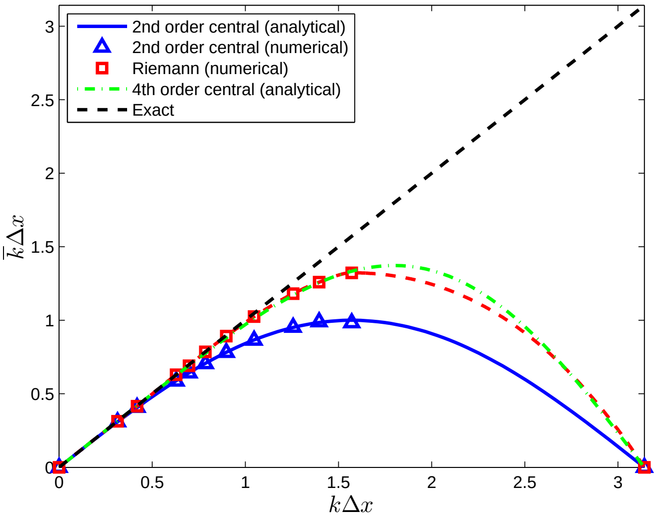


Figure 4.5: Exact versus numerical wavenumber. Cubic interpolants (dashed lines) are fitted to the data for the central and Riemann schemes.

We conclude from the discussed 1D testcases that the new implementation shows good results regarding dispersion and a reasonable amount of diffusion.

4.2 Gaussian pulse in uniform flow

This model problem is used to study the convection effect and to assess the quality of the imposed boundary conditions. The following initial disturbance is given:

$$p'(\mathbf{x}, t_0) = \epsilon e^{-\alpha(x^2+y^2+z^2)}. \tag{4.4}$$

The corresponding solution is [6]:

$$p'(\mathbf{x}, t) = \frac{\epsilon}{\beta} \int_0^\infty \xi^2 e^{-\xi^2/(4\alpha)} \cos(t\xi) j_0(\xi\eta) \mathrm{d}\xi, \tag{4.5}$$

where ξ is a dummy variable. The other parameters are listed in Tab. 4.2.

Parameter	Symbol	Value
Amplitude	ϵ	1
Pulse half-width	b	9
-	α	$\ln(2)/b^2$
-	η	$\sqrt{(x - M \cdot t)^2 + y^2 + z^2}$
-	β	$2\alpha\sqrt{\pi\alpha}$
-	j_0	$\sin(\xi\eta)/\xi\eta$

Table 4.2: Gaussian pulse parameters. M is the Mach number and j_0 is the spherical Bessel function of the first kind and zeroth order.

A uniform grid of $N = 100$ cells in all three directions on a domain of $L = 100$ was used. The Mach number is set at 0.2. In Fig. 4.6 the numerical solution is compared with the exact solution. The pulse is correctly convected, although a slight boundary influence is observed around $x = 50$.

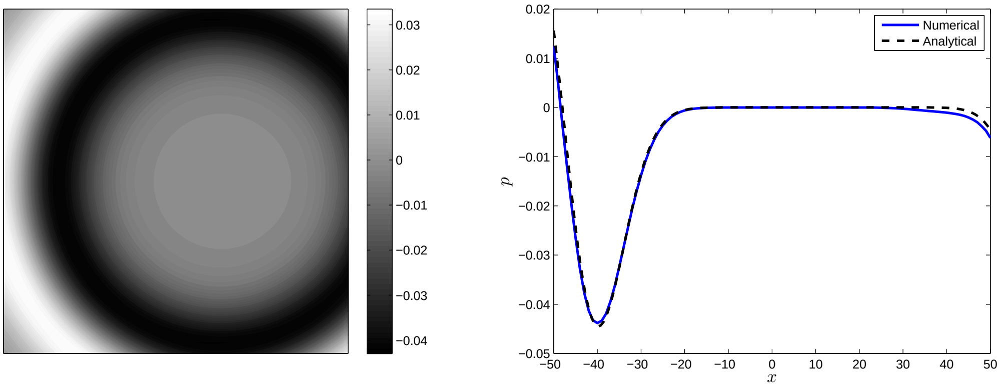


Figure 4.6: Left: Acoustic pressure contours of the pulse in the xy plane at $t = 60$. Right: Comparison with the analytical solution along the x-axis.

A grid refinement study was conducted on a two-dimensional larger domain of 200 x 200, varying the number of cells from $N = 25^2$ up to $N = 400^2$ for both the standard implementation and the Riemann based solver. The root-mean-square error was computed

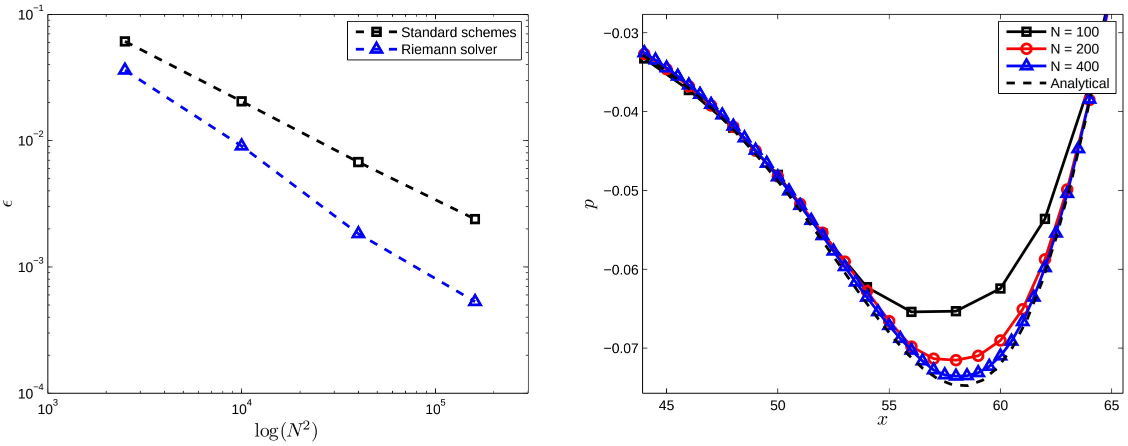


Figure 4.7: Left: Comparison between the standard solver (with central discretization) and the Riemann based solver. Right: Convergence of the solution obtained with the Riemann solver along the x-axis (close-up).

along the x-axis and is shown in Fig 4.7. It can be seen that the standard solver has an order of around 1.6, while the Riemann based solver shows truly second order accuracy in space. On coarse grids it was observed that dispersive errors occur when the standard implementation is used, while the Riemann based solver was able to preserve the propagation pattern. In this test case, the Riemann solver gives on a grid of 50 x 50 almost the same error as the standard solver with four times as much cells.

Buffer zone. As described in Section 2.3.2 a buffer zone can be implemented by adding a source term $\mathbf{S} = -\sigma \mathbf{q}$ to the right-hand side with σ given by:

$$\sigma(r) = \begin{cases} \sigma_{max}\left(\frac{r-r_0}{w}\right)^m & \text{if } r \geq r_0, \\ 0 & \text{if } r < r_0. \end{cases} \tag{4.6}$$

A two-dimensional circular mesh with a radius of 75 was made with OpenFOAM's meshing tool blockMesh. We set $\sigma_{max} = 2$, $r_0 = 50$ (radial position where the bufferzone starts), $w = 25$ (width of the bufferzone) and $m = 2$, which determines the gradualness of the damping. The pulse is initially located in the center of the domain. When the first wavefronts reach $r = 50$ the damping starts. In Fig. 4.8 the influence of the damping zone on the wave propagation is shown at different times. As can be seen, no sudden reflections occur at the border $r = 50$. At times $t = 50$, $t = 60$ and $t = 70$ the inner domain (i.e. the part where the solution is not modified) is not influenced by the presence of the bufferzone. At $t = 80$ a bulge becomes visible, but it dies out later. The reason for the appearance of the bulge at this later time is that the exponent m was set equal to two. This causes the wavefronts to experience the steeper part of the damping profile at a further radial position and thus at a later timelevel. When a linear profile ($m = 1$) is used, more reflections become visible in an earlier stage. From this test case and other tests it is found that a buffer zone can produce reasonable results. However, although the method is easily implemented compared to advanced methods such as PML, there are also reasons why not to use the technique. A few reasons are:

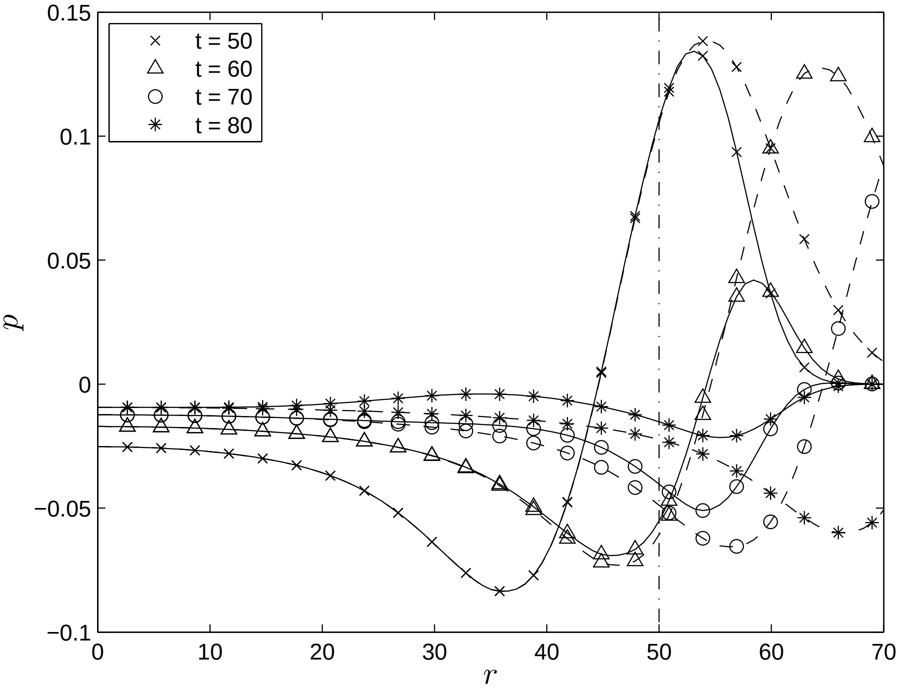


Figure 4.8: Acoustic pressure along a radial line for $t = 50$, $t = 60$, $t = 70$ and $t = 80$. The solid lines are the solution when the buffer zone is used, the dashed line is the solution on a very large domain (no boundary influence). The borders between the physical domain and the bufferzone are marked by the vertical dash-dotted line.

- Buffer zones generally work best if the waves are aligned with the bufferzone.
- The optimal setting of the parameters is problem specific.
- The extra layer adds computational cost.

For these reasons, the convective outlet is used in the rest of the project.

4.3 Elementary sources

The solver needs to be able to preserve the radiation patterns of simple acoustic sources. In this section the wave propagation from a monopole, dipole and quadrupole source is verified.

4.3.1 Monopole

A monopole represents mass injection and hence source terms in the continuity and energy equations. Because of the isentropic assumption, in our system this is simplified to only a term in the energy equation:

$$\mathbf{S} = \sin(\omega t) \cdot f(x, y) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \tag{4.7}$$

where $f(x, y) = \epsilon e^{-\alpha(x^2+y^2)}$ with ϵ as the pulse amplitude and α as a parameter that determines the source decay. The simulation is carried out with this forcing function in both a stagnant medium and in presence of a non-zero uniform velocity field. For a uniform flow with velocity u_0 (or on non-dimensional basis, Mach number M) in the x-direction, the LEE reduce to

$$\frac{D\mathbf{u}'}{Dt} + \frac{1}{\rho_0}\nabla p' = \frac{1}{\rho_0}\mathbf{S}_m, \tag{4.8a}$$

$$\frac{Dp'}{Dt} + \gamma p_0 \nabla \cdot \mathbf{u}' = S_e, \tag{4.8b}$$

where $\frac{D}{Dt} = \frac{\partial}{\partial t} + u_0 \frac{\partial}{\partial x}$, $\mathbf{S}_m$ the momentum source and S_e the energy source. By subtracting γp_0 times the divergence of the momentum equation from the substantial derivative of the energy equation, Eqs. (4.8) can be cast into a convective wave equation:

$$\frac{D^2 p'}{Dt^2} - c_0^2 \nabla^2 p' = \frac{DS_e}{Dt} - c_0^2 \nabla \cdot \mathbf{S}_m. \tag{4.9}$$

c_0 is the speed of sound. For this differential equation, an analytical solution can be found using a free space Green's function $\mathcal{G}(x, y, t)$. The differential operator remains the same and on the right hand side we place our harmonic source:

$$\frac{D^2 \mathcal{G}}{Dt^2} - c_0^2 \nabla^2 \mathcal{G} = \delta(x, y) e^{-i\omega t}. \tag{4.10}$$

For a monopole, the exact solution to this problem follows from the convolution product:

$$p'(x, y, t) = f(x, y) * \Big(\frac{\partial \mathcal{G}}{\partial t} + M \frac{\partial \mathcal{G}}{\partial x}\Big), \tag{4.11}$$

where $\mathcal{G}(x, y, t)$ is the following Green's function [1]:

$$\mathcal{G}(x, y, t) = \frac{i}{4c_0^2\sqrt{1 - M^2}} H_0^{(1)} \underbrace{\Big(\omega \frac{\sqrt{x^2 + (1 - M^2)y^2}}{(1 - M^2)c_0}\Big)}_{\xi} \exp \underbrace{\Big(-i\frac{M}{1 - M^2} kx - i\omega t\Big)}_{\eta}. \tag{4.12}$$

Here $H_0^{(1)}$ denotes the zeroth order Hankel function of the first kind (Bessel function of third kind). The spatial derivative of the Green's function is thus:

$$\frac{\partial \mathcal{G}}{\partial x} = \frac{\omega}{4c_0^3(1 - M^2)^{3/2}} \Big[M H_0^{(1)}(\xi) - i \frac{x}{\sqrt{x^2 + (1 - M^2)y^2}} H_1^{(1)}(\xi)\Big] \exp(\eta). \tag{4.13}$$

ξ and η are used here to denote the same arguments as in Eq. (4.12). In a stagnant medium Eq. (4.12) simplifies to:

$$\mathcal{G}(x, y, t) = \frac{i}{4c_0^2} H_0^{(1)}(\omega r / c_0) e^{-i\omega t}, \tag{4.14}$$

with $r = \sqrt{x^2 + y^2}$ so that:

$$\frac{\partial \mathcal{G}}{\partial x} = \frac{-i\omega x}{4c_0^3 r}\Big(J_1(\frac{\omega r}{c_0}) + iY_1(\frac{\omega r}{c_0})\Big)e^{-i\omega t}. \tag{4.15}$$

J_1 and Y_1 are the first order Bessel functions of respectively the first and second kind (note $H_1^{(1)} = J_1 + iY_1$). The returned pressure of Eq. (4.11) is complex. Since our harmonic source was a sinusoid, the requested acoustic pressure is $-\Im\{p'\}$.

Now that we know the reduced system of equations and have the exact solution, we turn to the numerical part. As a first case, we consider a stagnant medium and set $\epsilon = 1$, $\omega = \pi/15$ ($\lambda = 30$) and $\alpha = \ln(2)/9$ on a square domain with dimensions 200 x 200. The following grids are used:

1. An orthogonal grid with 100 x 100 equal sized hexahedra's, which comes down to 15 points (or cells) per wavelength (PPW).
2. An unstructured grid with an approximately equal amount of cells as for the orthogonal grid.
3. A very fine orthogonal grid with 400 x 400 cells.

In all cases the aacFoam solver is used. In Fig. 4.11 a comparison with the exact solution is shown for the first two medium sized grids and in Fig. 4.12 the exact solution is compared with the solution on the very fine mesh. For the medium sized meshes the numerical results are quite close to the exact solution, except near the origin. Excellent agreement is found with the very fine mesh. An impression of the found pressure field with this fine mesh can be seen in Fig. 4.9.

Secondly, because of the uniform flow, Eq. (4.9) accounts for two one-way interactions between flow and acoustics, namely a wavenumber shift (the Doppler effect) and convective amplification/damping (the Doppler factor). To check these, the monopole is placed in a Mach 0.5 uniform flow. The obtained pressure contours are shown in Fig. 4.10. The numerical solution along the x-axis is compared with the analytical solution in Fig. 4.13. Again good agreement is found. Upstream (left of the x-axis) the wavelength should be modified to $\lambda_u = (1 - M)\lambda = 15$ and downstream to $\lambda_d = (1 + M)\lambda = 45$. This is in agreement with the simulation result.

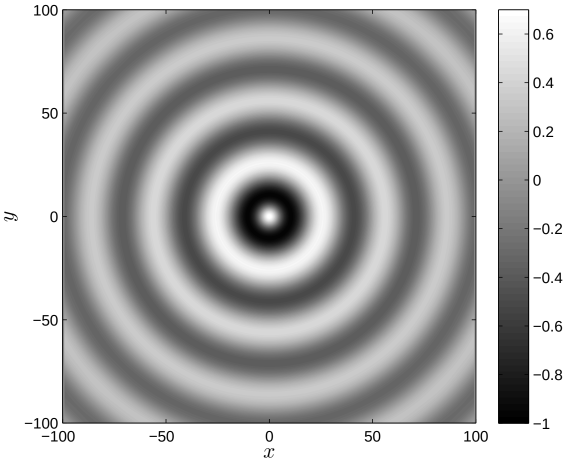


Figure 4.9: Monopole radiation in a stagnant medium. Instantaneous pressure contours at $t = 150$.

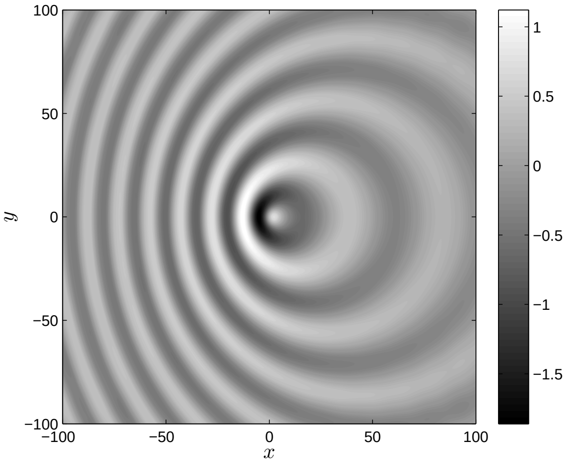


Figure 4.10: Monopole radiation in a uniform flow. Instantaneous pressure contours at $t = 270$.

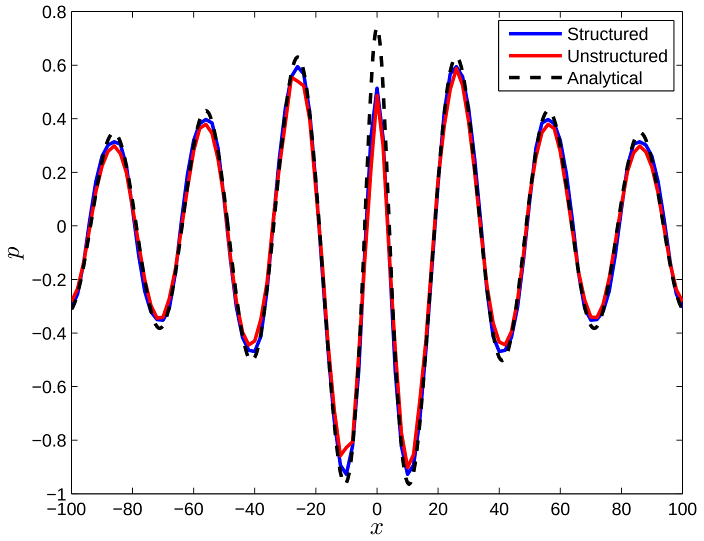


Figure 4.11: Numerical vs. analytical solution along the x-axis at $t = 150$ for the first two meshes.

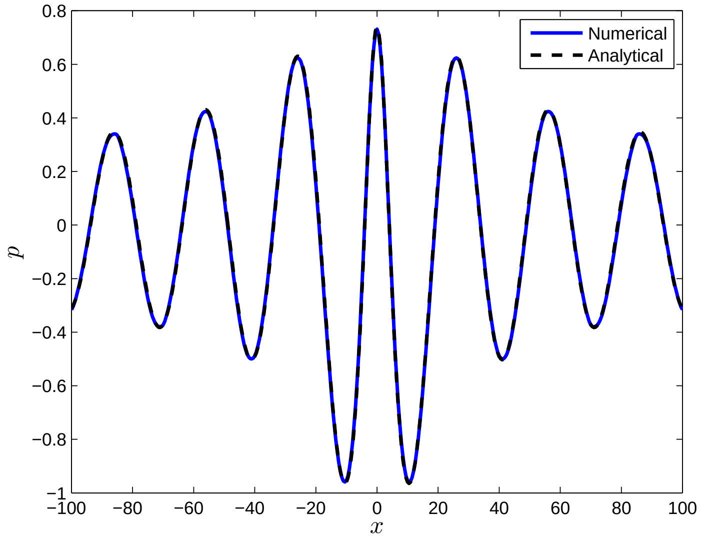


Figure 4.12: Numerical vs. analytical solution along the x-axis at $t = 150$. Fine mesh.

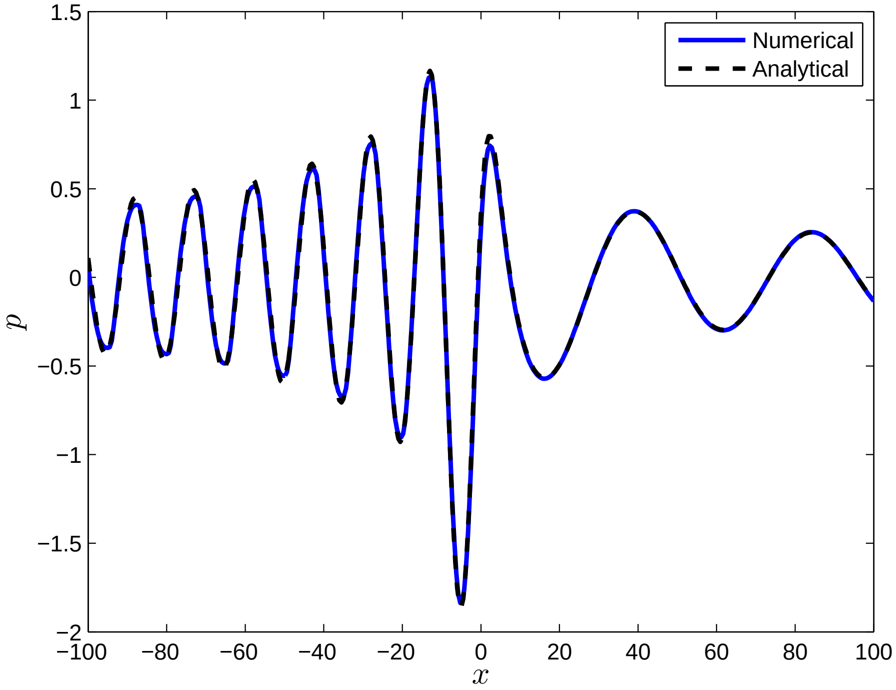
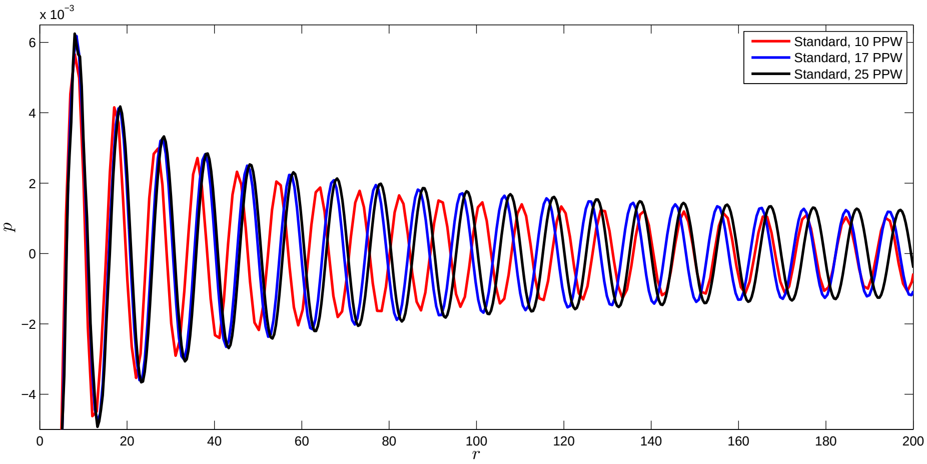


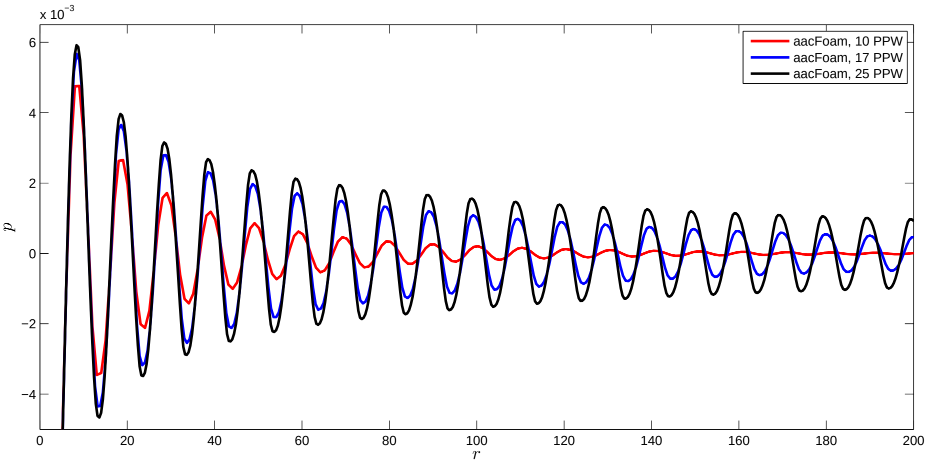
Figure 4.13: Numerical vs. analytical solution along the x-axis at $t = 270$ for the convected monopole.

Now that the preservation of the monopole radiation pattern with the aacFoam solver has been established, we look at the dispersive and dissipative properties of both solvers with a grid study. We set $\omega = \pi/5$ in Eq. (4.7), which gives $\lambda = 10$ and use a circular grid with radius 300. In Fig. 4.14 the acoustic pressure along a radial line is shown for a few refinement levels.

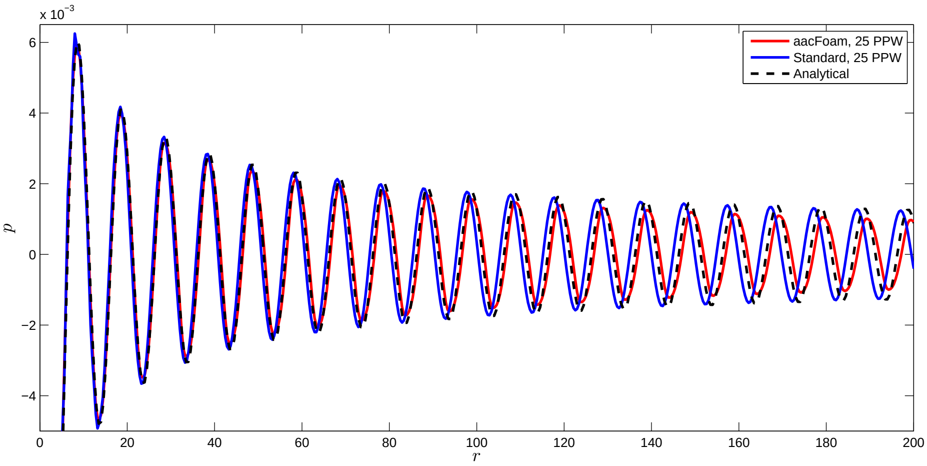
It can be seen that the standard solver shows clear differences in phase (Fig. 4.14(a)). The solver aacFoam is consistent regarding phase, but gets more dissipative as the grid is coarsened (Fig. 4.14(b)). From Fig. 4.14(c) it can be seen that during the first 100 meter no significant dispersion error has been built up and both solvers have the correct phase. At further distances, the standard solver shows a slightly more dispersive behavior due to the pollution effect (i.e. the dispersion error is generated over a single wavelength, but builds up as the wave travels through the computational domain).



(a)



(b)



(c)

Figure 4.14: Variation of p' along r for various grids at $t = 300$. Top: standard solver, middle: aacFoam, bottom: comparison between standard and aacFoam solver for the finest mesh. The PPW count is based on the radial direction; 5 to 6 pie slices are used for the circumferential direction.

4.3.2 Dipole

A dipole corresponds to a unsteady external force. Its source term is

$$\mathbf{S} = \sin(\omega t) \begin{bmatrix} f_x \\ f_y \\ 0 \end{bmatrix}, \tag{4.16}$$

with $f_x = \epsilon \cos(\pi/10x)e^{-\alpha y^2}$ evaluated where $(x, y) \in [-5; 5] \times [-5; 5]$ and $f_y = 0$. The exact solution is [1]:

$$p'(x, y, t) = -f_x(x, y) * \frac{\partial \mathcal{G}(x, y, t)}{\partial x}, \tag{4.17}$$

A fine mesh of 400 x 400 cells on a rectangular domain of 200 x 200 is used. In Fig. 4.15 the obtained pressure contours are shown. A clear dipole pattern is visible. A quantitative comparison between the numerical and analytical solutions is shown in Fig. 4.16. The agreement is perfect.

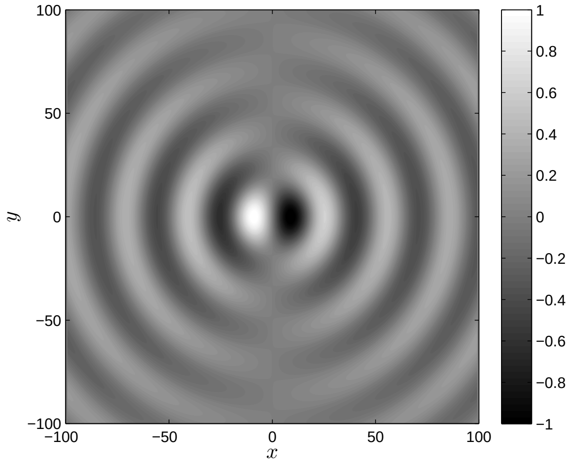


Figure 4.15: Acoustic pressure contours at $t = 150$.

4.3.3 Quadrupole

The source term is now given by

$$\mathbf{S} = \begin{bmatrix} \frac{\partial T_{xx}}{\partial x} + \frac{\partial T_{xy}}{\partial y} \\ \frac{\partial T_{yx}}{\partial x} + \frac{\partial T_{yy}}{\partial y} \\ 0 \end{bmatrix}, \tag{4.18}$$

where

$$T_{ij} = \begin{bmatrix} -\cos(\pi/20x)e^{-\alpha y^2} & 0 \\ 0 & \cos(\pi/20y)e^{-\alpha x^2} \end{bmatrix} \frac{20}{\pi} \epsilon \sin(\omega t) \tag{4.19}$$

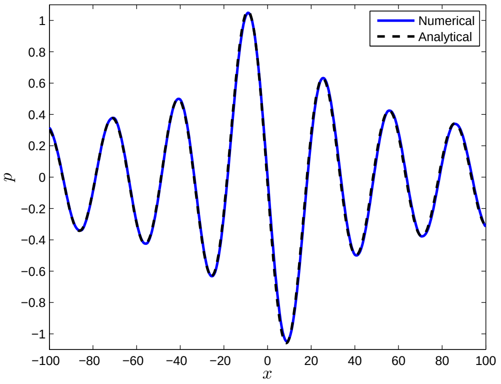


Figure 4.16: Numerical vs. analytical solution along the x-axis for the dipole source.

in $(x, y) \in [-10; 10] \times [-10; 10]$. The exact solution is:

$$p(x, y, t) = -\frac{\partial T_{xx}}{\partial x} * \frac{\partial \mathcal{G}}{\partial x} - \frac{\partial T_{yy}}{\partial y} * \frac{\partial \mathcal{G}}{\partial y}. \tag{4.20}$$

$\frac{\partial \mathcal{G}}{\partial x}$ is given by Eq. (4.15) and $\frac{\partial \mathcal{G}}{\partial y}$ follows analogously by replacing x for y in this expression. The numerical results are shown in Fig. 4.17 and Fig. 4.18. Again the expected radiation pattern is found.

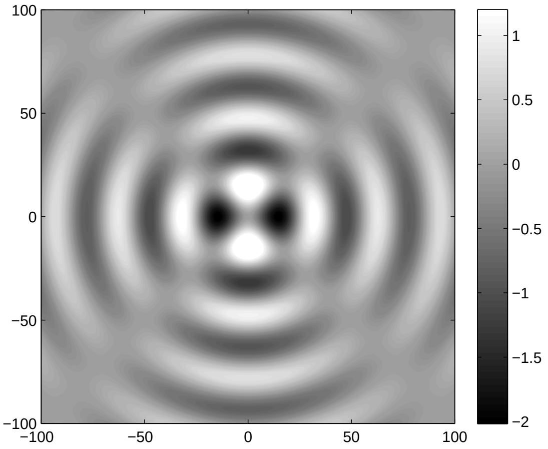


Figure 4.17: Acoustic pressure contours at $t = 150$. The same domain and discretization as for the dipole case was used.

4.4 Sound reflection from a solid wall

On solid surfaces acoustic scattering takes place. The boundary conditions on a solid surface are:

$$\frac{\partial p'}{\partial n} = 0, \qquad \mathbf{u}' \cdot \hat{\mathbf{n}} = \mathbf{0}. \tag{4.21}$$

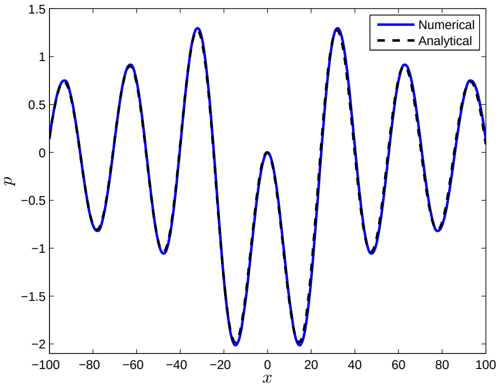


Figure 4.18: Numerical vs. analytical solution along the x-axis for the quadrupole source.

Note that since the acoustic equations are inviscid, only a condition for the normal component of the velocity is needed while the tangential component may slip at the wall. To study acoustic scattering, a two-dimensional Gaussian pulse is placed somewhere in the domain, which has along the x-axis a solid wall. The medium is stagnant. The initial pressure disturbance when the pulse is placed at $x = 0$ and $y = 25$ is

$$p'(\mathbf{x}, t_0) = e^{-\alpha(x^2+(y-25)^2)}, \tag{4.22}$$

where α is now chosen as $\ln(2)/25$. The initial velocity perturbations are - like in the previous cases - set equal to zero. The analytical solution is [14]:

$$p'(\mathbf{x}, t) = \frac{1}{2\alpha}\int_0^{\infty} e^{-\frac{\xi^2}{4\alpha}}\cos(\xi t)[J_0(\xi\eta) + J_0(\xi\zeta)]\xi \mathrm{d}\xi \tag{4.23}$$

J_0 is a Bessel function of the first kind and zeroth order, $\eta = \sqrt{x^2 + (y-25)^2}$, and $\zeta = \sqrt{x^2 + (y+25)^2}$. Eq. (4.23) is compared with the numerical solution along the wall at $t = 40$ in Fig. 4.19. The agreement is perfect.

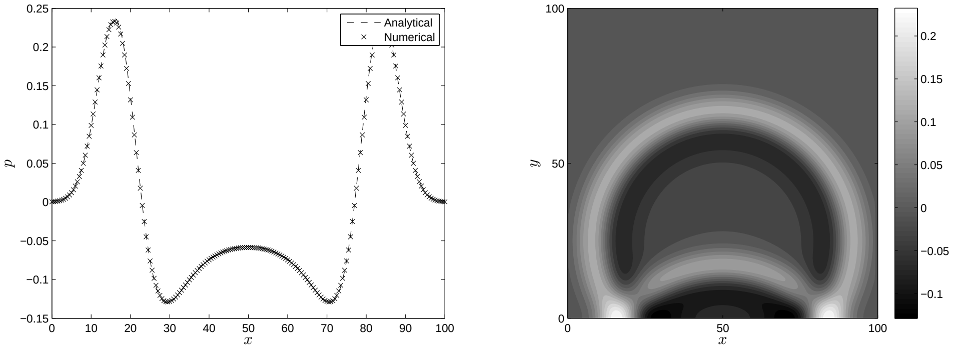


Figure 4.19: Left: Numerical vs. analytical solution for p' at $t = 40$. Right: Isocontours of p' at $t = 40$. A uniform grid of 200 x 200 cells was used on a domain of 100 x 100.

4.5 Source radiation in non-uniform flow

When the background flow is non-uniform refraction effects occur. To check whether the solver can correctly represent refraction, the radiation of a monopole source in two flows with non-zero vorticity are considered: a shear flow and a jet. Instability waves are absent in the first case, but can be formed in the latter case. Although no analytical solutions exist, reference solutions are available.

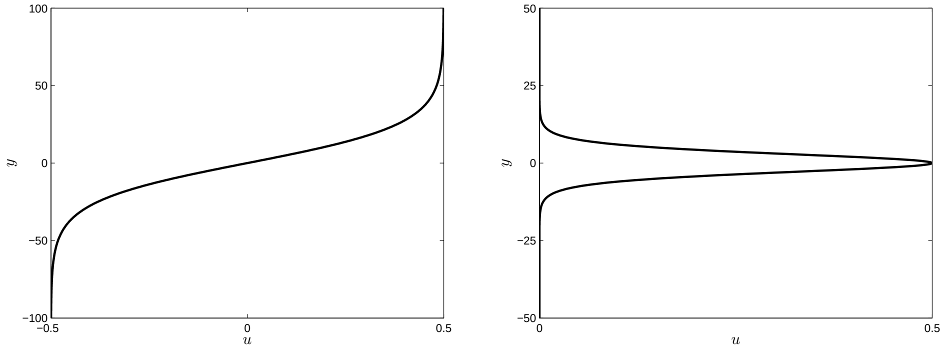


Figure 4.20: Velocity profiles of a shear flow (left) and a jet (right).

4.5.1 Monopole radiation in a shear flow

A monopole is placed in a two-dimensional shear flow of which the profile is given by:

$$u = \frac{1}{2}\tanh(2y/\delta_w). \tag{4.24}$$

Here δ_w is the shear layer thickness, here set equal to 50. The convection velocity is 0 at $y = 0$ and has a maximum of 0.5 (this is also the Mach number). The y-component of the velocity is zero. The monopole is located in the center of a rectangular domain of size 200 x 200 and is governed by a source term in the energy equation:

$$S(\mathbf{x}, t) = \cos(\omega t) e^{-\alpha(x^2+y^2)}, \tag{4.25}$$

with $\alpha = \ln(2)/9$ and $\omega = 0.5$, i.e. identical to the set-up of Ewert and Schröder in [11]. In this reference the fourth-order dispersion relation preserving (DRP) scheme of Tam and Webb [27] is used for the space discretization, while for the temporal discretization the fourth-order low-dissipation and low-dispersion Runge-Kutta scheme (LDDRK 5-6) by Hu is used. They used a uniform mesh with 201 x 201 points. Initially, this meshsize was also used here, but it was found that the wave amplitudes are then underpredicted. For the next results, the mesh size is double of that of Ewert and Schröder, which comes down to 25 PPW[1]. In Fig 4.21 the solution with aacFoam is shown and the reference solution is shown in Fig 4.22.

At a first glance, the patterns look very similar. However, a difference in phase can be observed. Even with a very fine mesh (the maximum resolution that was tested was 50

---

[1] The PPW count is based on the original wavelength.

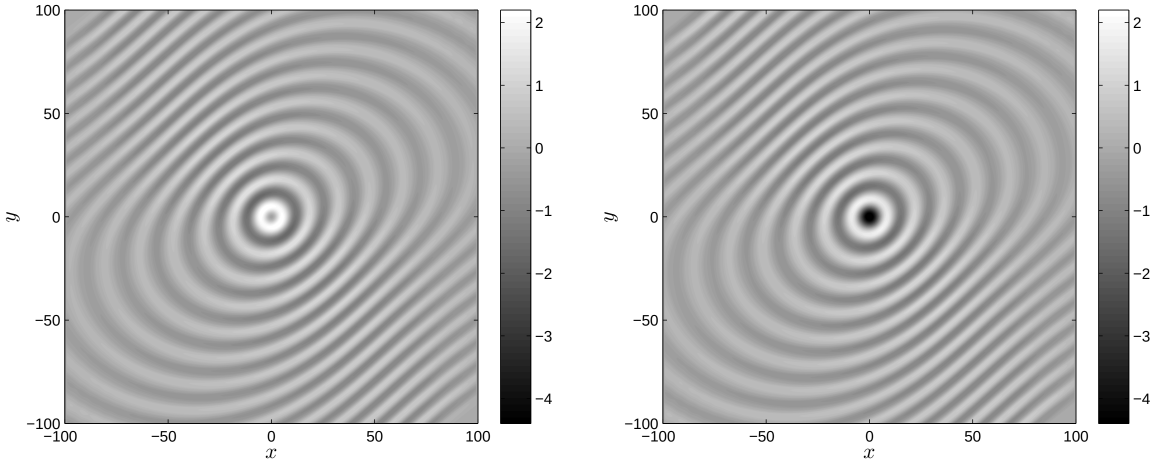


Figure 4.21: Acoustic pressure contours obtained with aacFoam. Time instant $t = 180$ with original source term (left) and with the source term that has a phaseshift (right).

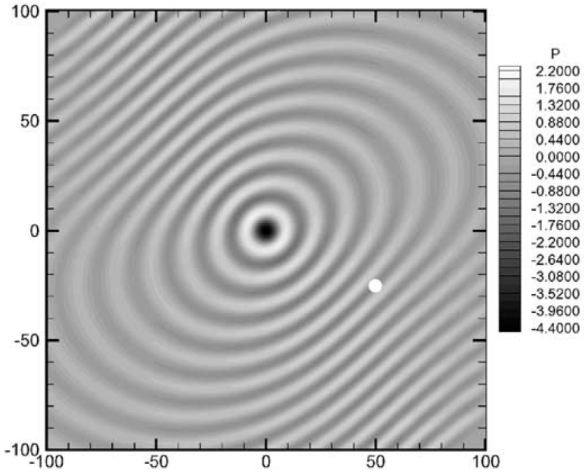
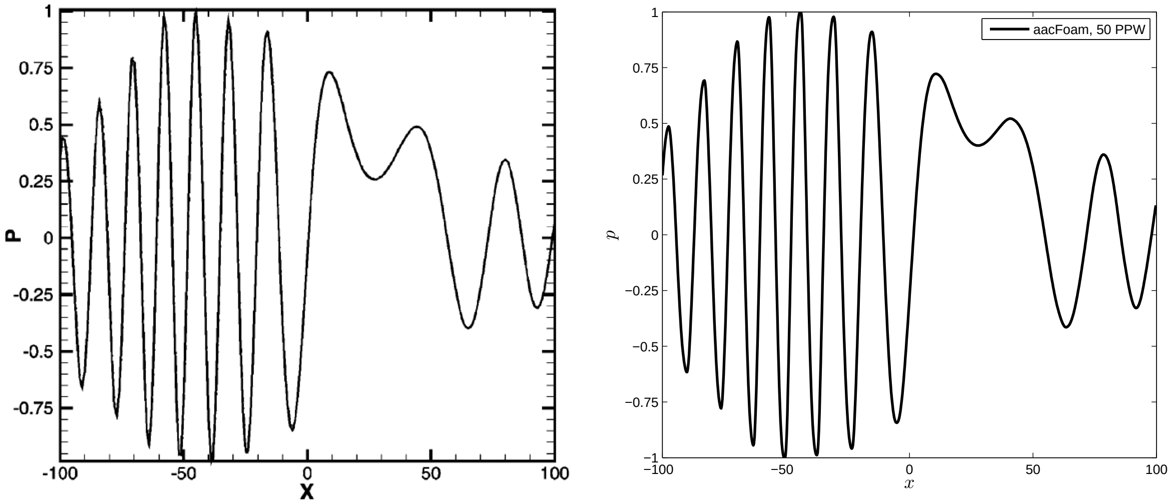


Figure 4.22: Acoustic pressure contours of the reference solution. Time instant $t = 180$.

PPW), this difference remained. It was however realized from testcases in a stagnant medium that it is impossible to have the phase shown in the reference with the present source term $f(x, y)\cos(\omega t)$ at $t = 180$. Setting the source term as $f(x, y)\cos(\omega t + 1/2\pi) = -f(x, y)\sin(\omega t)$ gives qualitatively nearly exact agreement (compare Fig. 4.21 (right) with Fig. 4.22) and we therefore think that observed phase difference is not due to the solver, but due to a typo in the definition of the source term in the reference.

In Fig. 4.23(a) the distribution of p' along $y = 70$ of the reference solution is shown. To enable a comparison with Fig. 4.23(a), the next results we obtained with the modified source term.

From Fig. 4.24 it can be seen that when the number of points per wavelength is increased and aacFoam is used, the phase of the waves remains the same, but the amplitude in-



(a) Reference.

(b) Fine solution.

Figure 4.23: Reference solution along the line $y = 70$ at $t = 180$ and the OpenFOAM solution on a very fine grid.

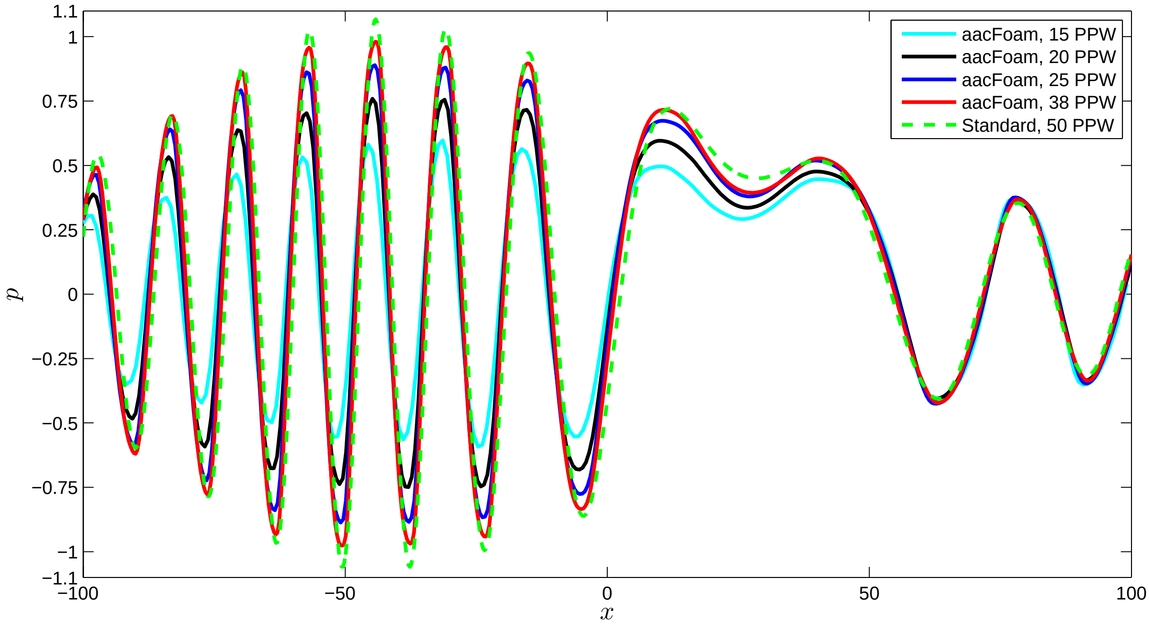


Figure 4.24: Grid refinement with the aacFoam solver.

creases. The amplitude with the standard solver is larger in most of the domain. However, from Fig. 4.25 it is revealed that the amplitudes are overpredicted with the standard solver and that for a reasonable amount of PPW (20, 25) the standard solver is also out of phase. Again the central differencing scheme was used for the standard solver. It was tested that

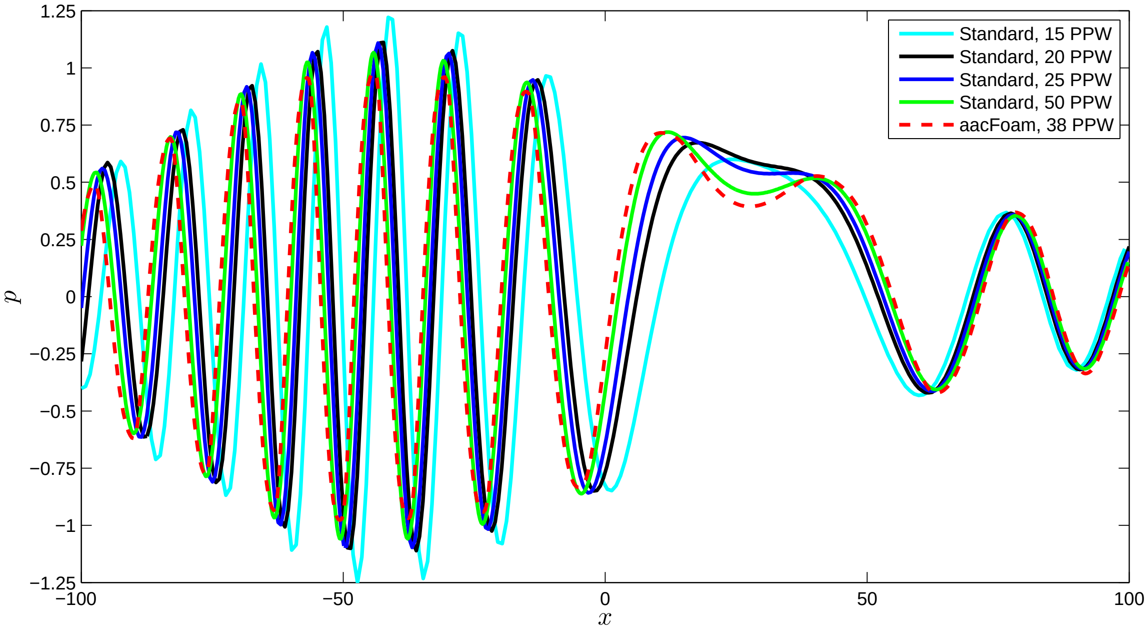


Figure 4.25: Grid refinement with the standard solver. Note that the y-axis has different limits than in Fig. 4.24.

with 25 PPW the results are quite similar when the limited linear or linear upwind scheme is used. With 50 PPW the standard solver and aacFoam become nearly coherent, which indicates that aacFoam performs better regarding dispersion. A rough quantitative indication is given in Tab. 4.3, where the amplitude and phase errors at $x = 50$ are determined with respect to a very fine solution using aacFoam (50 PPW). Overall, the aacFoam solver shows good quantitative agreement with the reference figure, Fig. 4.23(a).

	Central		**aacFoam**
PPW	ϵ_A [%]	$\epsilon_\phi [^\circ]$	ϵ_A [%]
15	+25	110	-45
20	+10	60	-26
25	+9	40	-11
38	+6	20	-2
50	+6	14	reference value

Table 4.3: Errors in amplitude (ϵ_A) and phase for the PPW range around the $x = -50$ position. Since for aacFoam the phase is relatively consistent on the various refinements, it has been omitted here.

To strengthen the verification for non-uniform flows, a second non-uniform flow case is considered.

4.5.2 Monopole radiation in a jet flow

In this section the influence of a jet flow on the wave propagation is studied. The velocity field of the jet is governed by:

$$u = \frac{M}{\cosh^2[(1+\sqrt{2})y/b]}, \tag{4.26}$$

where M is the axial Mach number and b the jet half-width, here taken as respectively 0.5 and 10. A monopole source is placed along the center-line of the jet in $(x, y) = (-100, 0)$ and governed by a source term in the energy equation:

$$S(\mathbf{x}, t) = 0.01\cos(2\pi t/9)e^{-\ln(2)/9\cdot(x^2+y^2)}. \tag{4.27}$$

The angular frequency is set such that $\lambda \approx b$. The resulting radiation pattern should be altered by the mean flow such that the maximum acoustic level occurs in the downstream direction at an angle $\theta = \cos^{-1}(\frac{1}{1+M}) \approx 47°$, and forming a shadow zone for smaller angles [1].

In Fig. 4.26 the results obtained with the two solvers are shown at time level $t = 400$. Both figures are very similar: the peak acoustic level occurs at roughly 45°, upstream small waves are trapped in the jet and in the downstream direction waves are starting to detach near the shadow zone. The lobes of the original solver are slightly longer and more divergent. A uniform mesh with 1 million cells was used. In Fig. 4.27 the result with aacFoam at $t = 900$ is compared with the solution obtained by Bailly and Juvé [1].

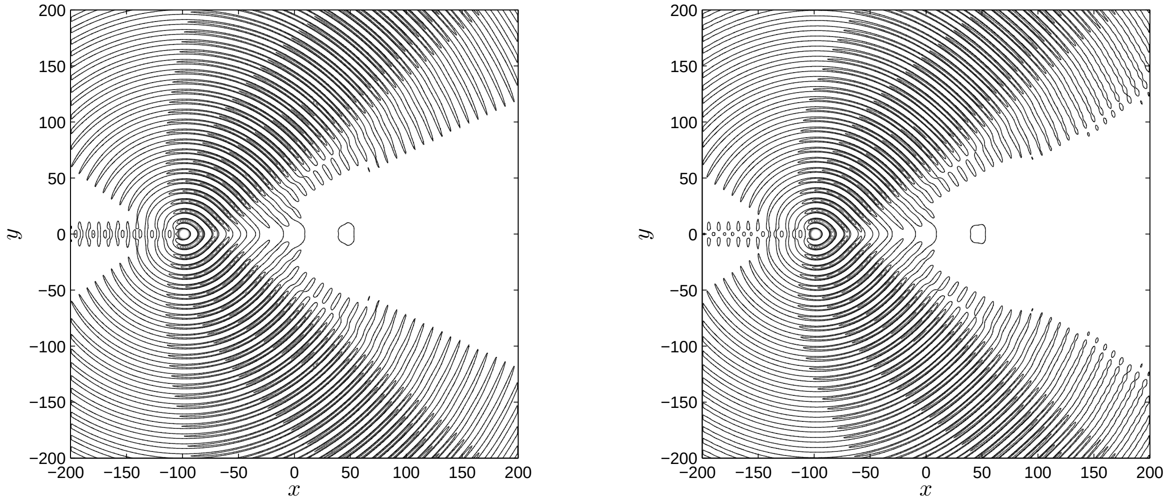


Figure 4.26: Pressure contours 0.001 to 0.01 (increment 0.001) and $1 \cdot 10^{-4}$. Left: Original solver. Right: aacFoam. Snapshot at $t = 400$.

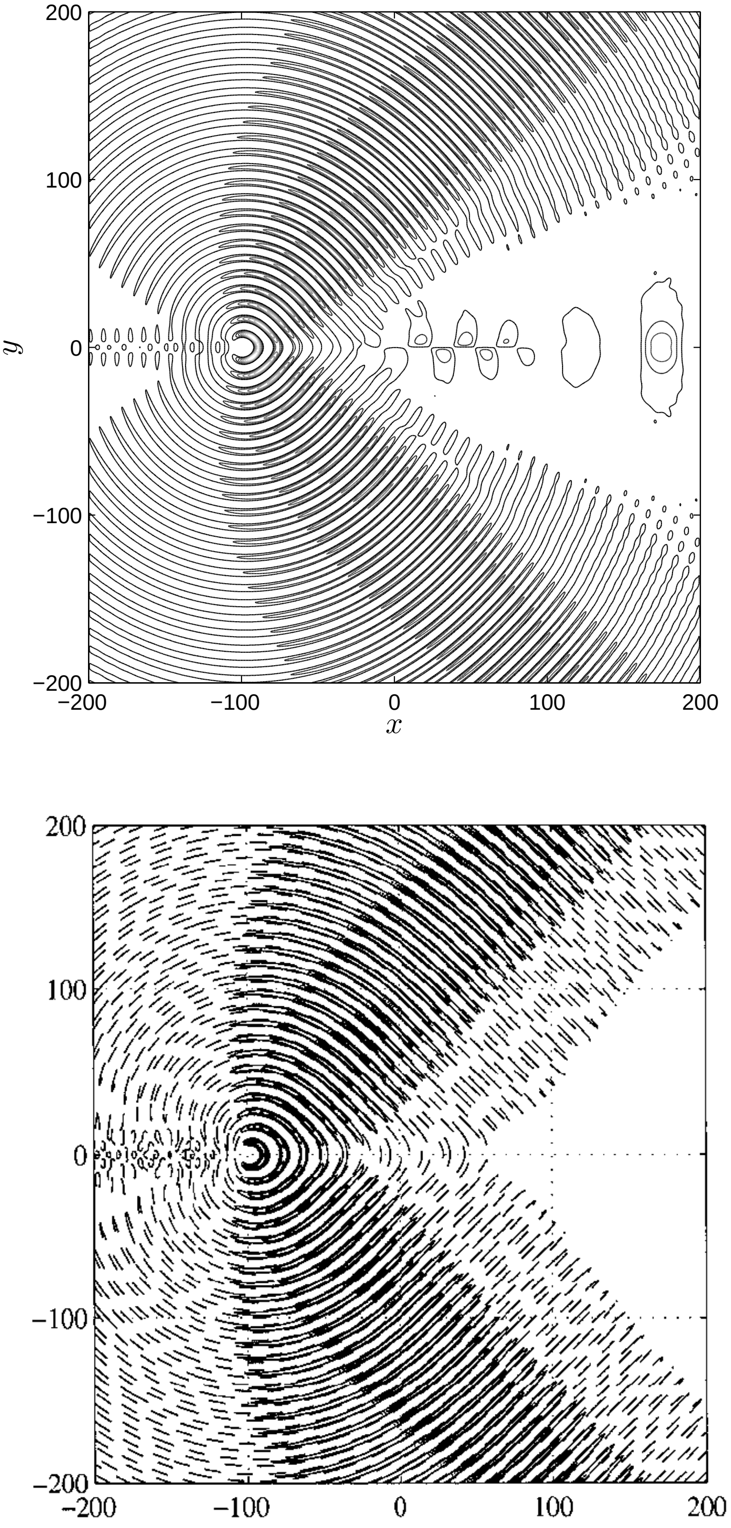


Figure 4.27: Iso-contours of p'. Contours from 0.001 to 0.01 (increment 0.001) and $1 \cdot 10^{-4}$. Top: aacFoam. Bottom: Reference. Snapshots at $t = 900$ (100 wave periods).

The reference solution was obtained using a fourth-order 7-point dispersion relation preserving scheme as well as an artificial selective damping technique that filters out high frequency waves by adding damping terms. The obtained result shows good agreement with the reference solution. As can be seen from the aacFoam solution, at $t = 900$ some rearward waves have detached, a process that could be observed in an early stage in Fig. 4.26. As they move downstream they are damped out. To damp instability waves, in the reference an artificial damping technique is used. Nevertheless, the general features strongly resemble that of the reference.

To show the effect of decreasing the meshsize, the acoustic pressure is extracted along a line that passes through the main lobe, see the schematic in Fig. 4.28. The previously discussed results obtained with aacFoam are used as reference solution, since the original solver showed at $t = 900$ an overprediction of the lobesizes (i.e. an overprediction in wave amplitudes) as well as a slightly bigger dispersion error and stronger downstream instability waves. Repeating the case on a quarter million point mesh, yields the results shown in Fig. 4.29.

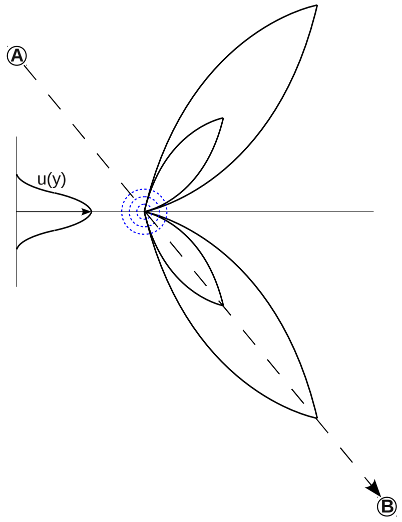
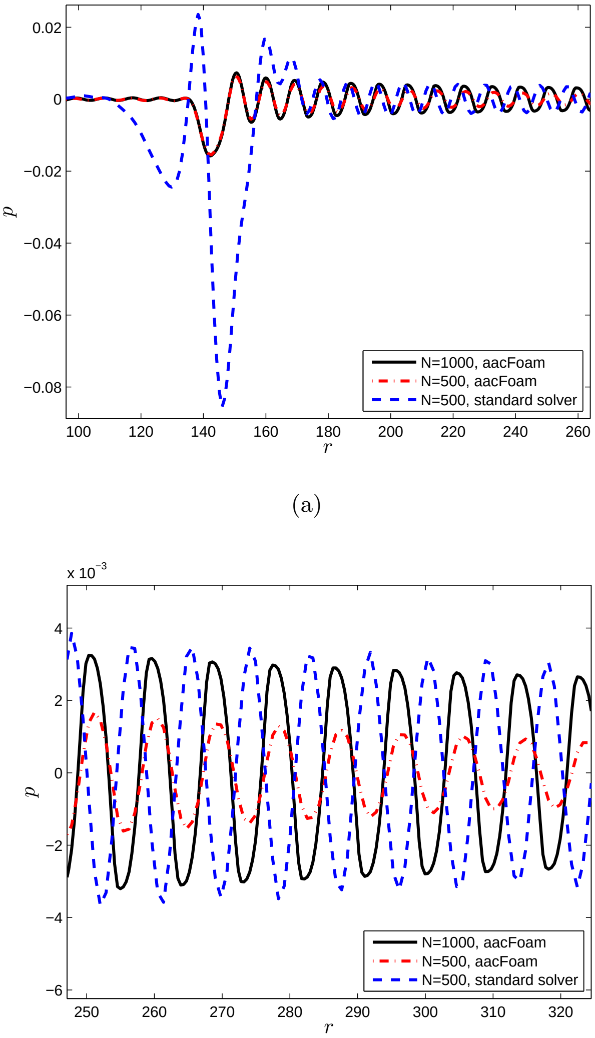


Figure 4.28: Line from point A(-200,100) to point B(100,-200), passing through the monopole source and one of the main lobes.

Based on the cases of the previous paragraphs, the results are as expected. The acoustic pressure obtained with aacFoam with $N = 500$ (≈ 11 PPW) follows the same trend as with $N = 1000$, but is more dissipative. The original solver performs better regarding the amplitude, but shows large oscillations in the upstream region and a phase error. In the reference already 14 PPW in the downstream direction were used, while the method is fourth-order accurate method in both space and time. It is thus hardly a surprise that the result with 11 PPW are no longer close to that of the reference.



(a)

(b)

Figure 4.29: Acoustic pressure versus the radial coordinate r measured along the line. A close-up p' in the far-field (towards point B) is shown in 4.29(b). $t = 900$.

4.6 Interim conclusion

From the test cases of the previous paragraphs it can be concluded that, given the correct description of mean flow and source data, both the standard LEE and Riemann-based LEE implementation are able to correctly account for convection and refraction effects. The aacFoam solver has as advantage that it propagates waves in a less dispersive manner than the standard LEE implementation. The aacFoam solver has as disadvantage that it is more diffusive than the standard implementation. When the overall performance is expressed in terms of the root-mean-square error, the new implementation is better.

Chapter 5

Validation of the hybrid method

An often used academic test case for validating an aeroacoustic code is the sound production by a pair of co-rotating vortices. This is a laminar, two-dimensional case. The chapter is structured as follows. In Sec. 5.1 the Direct Noise Simulation is described. Here the flow simulation results and the directly calculated acoustic field are discussed. Next, in Sec. 5.2 the results of the flow simulation are used to construct the acoustic source and injected in the acoustic solver. The acoustic field obtained with the hybrid method is compared with that of the direct simulation. Efficiency matters of the hybrid method are adressed.

5.1 Direct Noise Simulation

Before we test the hybrid method, the correctness of the flow simulation needs to be ensured. Moreover, the direct simulation is compressible and thus includes acoustic perturbations. The acoustic field found with the direct simulation will be used to compare with the results from the hybrid method.

5.1.1 Flow description

The noise generation from two co-rotating vortices in a stagnant medium is studied, see Fig. 5.1. Each vortex is given the following initial tangential velocity profile:

$$V_\theta(r) = -\frac{\Gamma r}{2\pi(r_c^2 + r^2)}, \tag{5.1}$$

where Γ is the circulation strength, r_c the vortex core radius and r the radial distance from the vortex core. OpenFOAM requires the x and y-components of the velocity, which follow from Eq. (5.1) after some simple algebraic manipulations. The case parameters are summarized in Tab. 5.1. The other fluid properties are set as follows: $\gamma = 1.4$ (ratio of specific heats), $p_0 = 1.0125 \cdot 10^5$ Pa (ambient pressure), $Pr = 0.7$ (Prandtl number) and $c_0 = 340$ m/s (speed of sound).

Parameter	Formula	Value
Half separation distance, r_0	-	$3.6 \cdot 10^{-3}$ m
Core radius, r_c	$\frac{2}{9}r_0$	$8 \cdot 10^{-4}$ m
Circulation intensity, Γ	-	1.71 m^{2}s^{-1}
Angular frequency, ω	$\frac{\Gamma}{4\pi r_0^2}$	$1.05 \cdot 10^4$ s^{-1}
Period, T	$\frac{8\pi^2 r_0^2}{\Gamma}$	$5.98 \cdot 10^{-4}$ s
Acoustic wavelength, λ_a	$\frac{1}{2}c_0 T$	$28.3 r_0$
Reynolds number, Re	$\frac{\Gamma}{\nu}$	$1.14 \cdot 10^5$
Tangential Mach number, M_t	$\frac{\Gamma}{4\pi r_c c_0}$	0.5
Rotational Mach number, M_r	$\frac{\Gamma}{4\pi r_0 c_0}$	$\frac{1}{9}$

Table 5.1: Summary of the case parameters and relations.

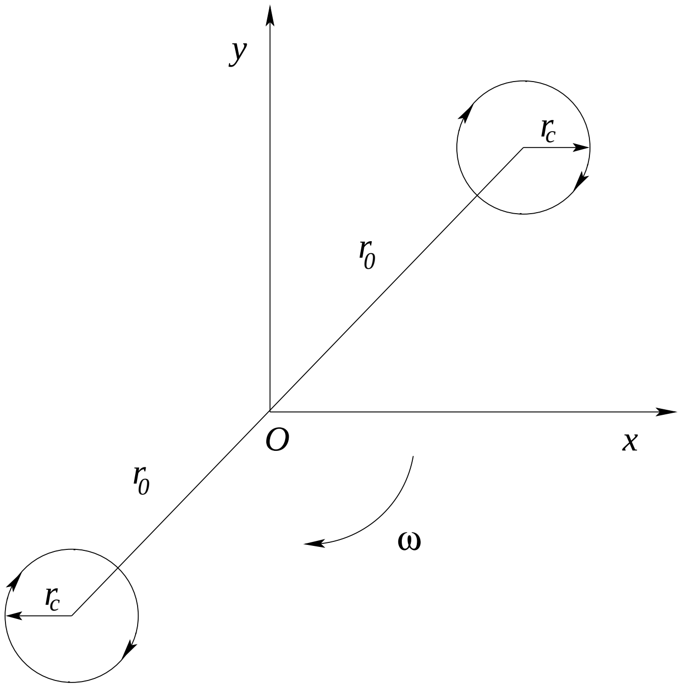


Figure 5.1: Schematic of two vortices rotating around the origin of a two-dimensional plane.

5.1.2 Flow solver

OpenFOAM's compressible flow solver rhoPimpleFoam is used. It uses the PISO-SIMPLE ("PIMPLE") algorithm for the pressure-velocity coupling. The PIMPLE algorithm is the PISO algorithm with two possible extensions, namely an outer correction loop and under-relaxation of variables. The goal of the algorithm is to circumvent the $\mathrm{Co} < 1$ limitation by stabilizing the simulation in this way, allowing for larger timesteps to be taken. A rough lay-out of the algorithm for the rhoPimpleFoam code is shown in Fig. 5.2. A detailed description of the steps can be found in chapter 3 of the thesis of Jasak [16], where also the same notation is used as in the OpenFOAM code.

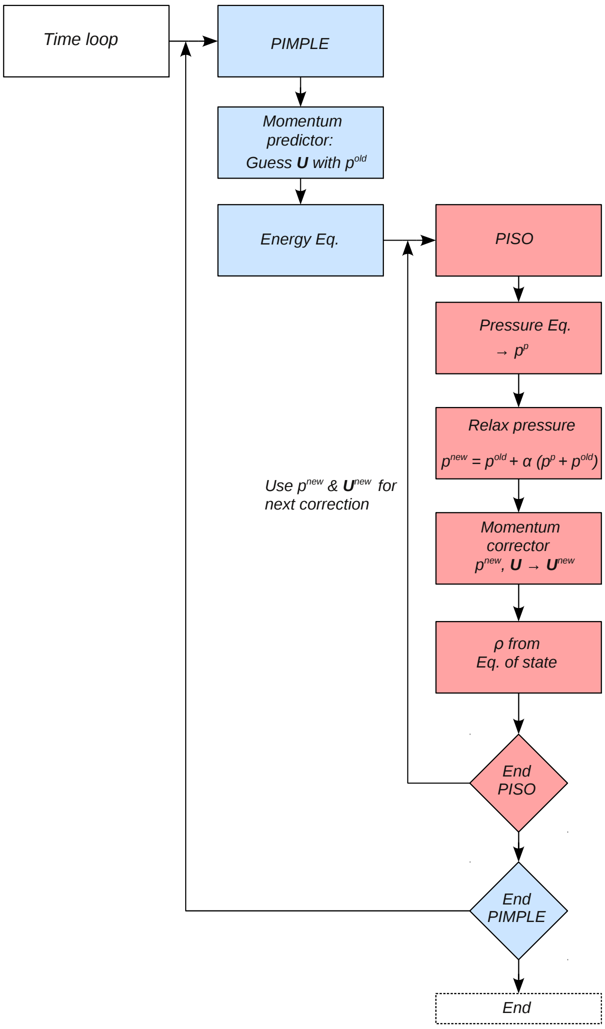
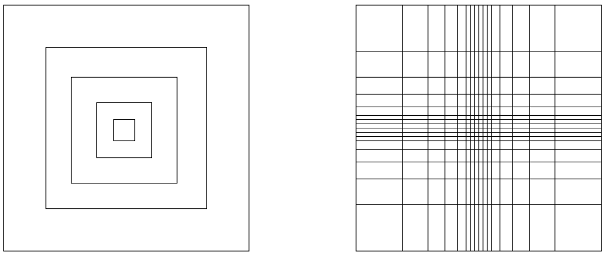

Time loop
PIMPLE
Momentum predictor: Guess $\boldsymbol{U}$ with p^{old}
Energy Eq.
PISO
Pressure Eq. → p^p
Relax pressure
$p^{new} = p^{old} + \alpha\,(p^p + p^{old})$
Use p^{new} & $\boldsymbol{U}^{new}$ for next correction
Momentum corrector p^{new}, $\boldsymbol{U} \rightarrow \boldsymbol{U}^{new}$
ρ from Eq. of state
End PISO
End PIMPLE
End

Figure 5.2: PIMPLE algorithm. The pressure equation is corrected for non-orthogonality during each PISO iteration. The energy equation is solved outside the PISO loop. One might expect it inside the PISO loop when a strong pressure-temperature coupling exists.

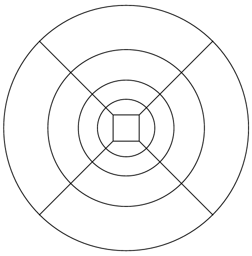
I noticed that when the solver operated in PISO mode (only 1 corrector) at an initial Courant number of 0.5, the simulation would diverge. Early tests indicated that three outer correctors, one pressure correction and one non-orthogonality correction would yield satisfactory results, so these settings were kept.

5.1.3 Mesh design

As mentioned in the introductory chapter, there is a large length scale separation between flow and acoustic. In this case the characteristic length scale for the flow is r_0 and for acoustic $\lambda = 28.3r_0$. For the flow simulation we therefore need a much finer grid. The velocity field of the co-rotating vortices is only non-zero in a small part of the domain. To resolve the flow field a uniform grid with dimensions $8r_0$ x $8r_0$ is used. The cellsize in this region is $\Delta = 1/36r_0$, which is about twice as small as in other studies [7, 22] where higher order schemes are used. Initial tests showed that $\Delta = 1/18r_0$ and $\Delta = 1/27r_0$ also work fairly well, but lead to a slightly different rotation period and earlier merging. Since an entirely uniform mesh leads to a very high cell count, a certain mesh strategy is needed to couple this central region and the farfield. Several types of meshes were tested, see the sketches in Fig. 5.3.



(a) Zonal. (b) Cross.



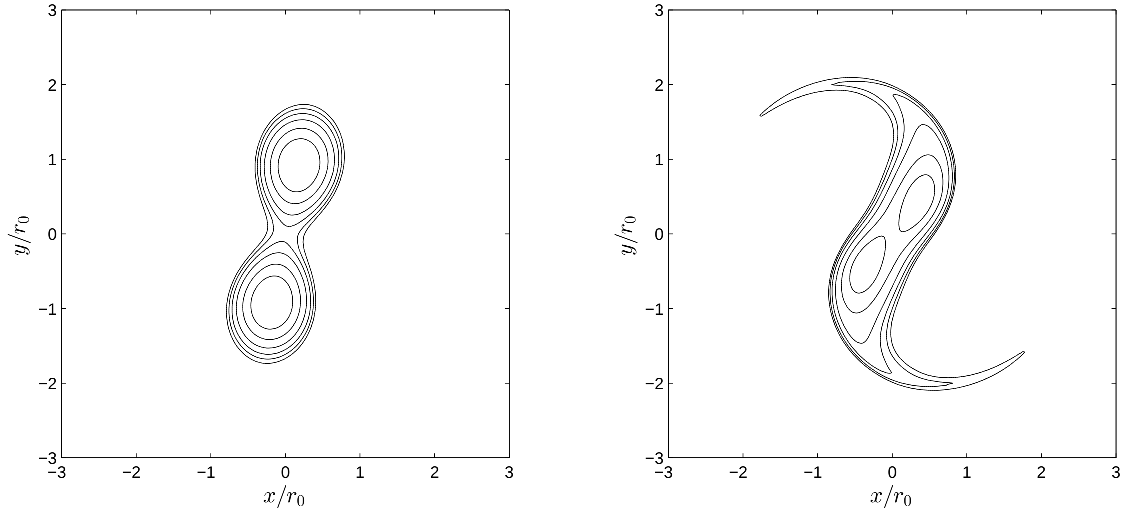
(c) Circular.

Figure 5.3: Mesh types.

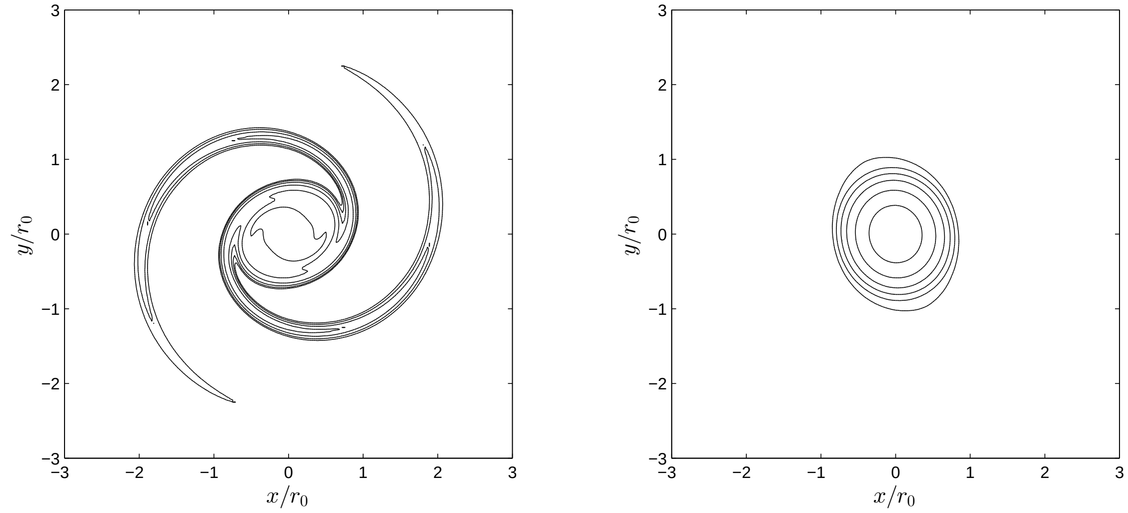
Since all meshes have the same center region, the results of the flow simulation (flow only) are the same; we will discuss the influence of the above mesh types on the acoustic in Sec. 5.1.5.

5.1.4 Flow results

From theory [22, 24] the evolution of the vortices is known to be as follows. Initially the vortices are well separated and rotate with a nearly constant angular velocity around their common center. This phase extends for approximately six periods. After this, the vortices start to merge, while ejecting vorticity filaments. The combined structure is an elliptically shaped vortex, which after some time becomes a single circular vortex. The simulated stages of the vortex system are shown in Fig. 5.4.



(a) Rotation, $t = 2.2 \cdot 10^{-3}$ s. (b) Acceleration, $t = 3.9 \cdot 10^{-3}$ s.



(c) Merging, $t = 4.1 \cdot 10^{-3}$ s. (d) Axissymmetrization/diffusion, $t = 5.0 \cdot 10^{-3}$ s.

Figure 5.4: Evolution of the vortex system. Six vorticity iso-contours from 10^4 to $1.22 \cdot 10^5$ s^{-1} with a geometrical ratio of 1.65.

For validating the hybrid method, only the phase of co-rotation with a constant angular speed is of interest. The rotation period directly influences the sound frequency, which is due to symmetry expected to be twice the rotation frequency of the vortices. As noted before by other authors [18], the numerically simulated number of periods of co-rotation is strongly dependent on the order of the discretization in space and time. Indeed, in this research several numerical schemes were tested and confirmation of this statement was found. In Fig. 5.5 the recorded pressure fluctuation versus time at the point $(50r_0, 0)$ is shown for a few schemes. With a central discretization the vortices start their merging remarkably early. Oscillations in the velocity field and the reconstructed acoustic source were found with this scheme. This is due to the odd-even decoupling of cells and the lack of dissipation. On the other hand, with a first-order upwind scheme the vortices merge far too early due to too much numerical diffusion. The best results were obtained using the second order upwind scheme for the convective terms. With this set-up, smooth acoustic source contours were obtained (as will be presented in Sec. 5.2.1) and a roughly constant rotation period of $5.9 \cdot 10^{-3}$ s was found during the first six rotations.[1] At the seventh rotation, when the vortices suddenly get closer, the rotation period is reduced.

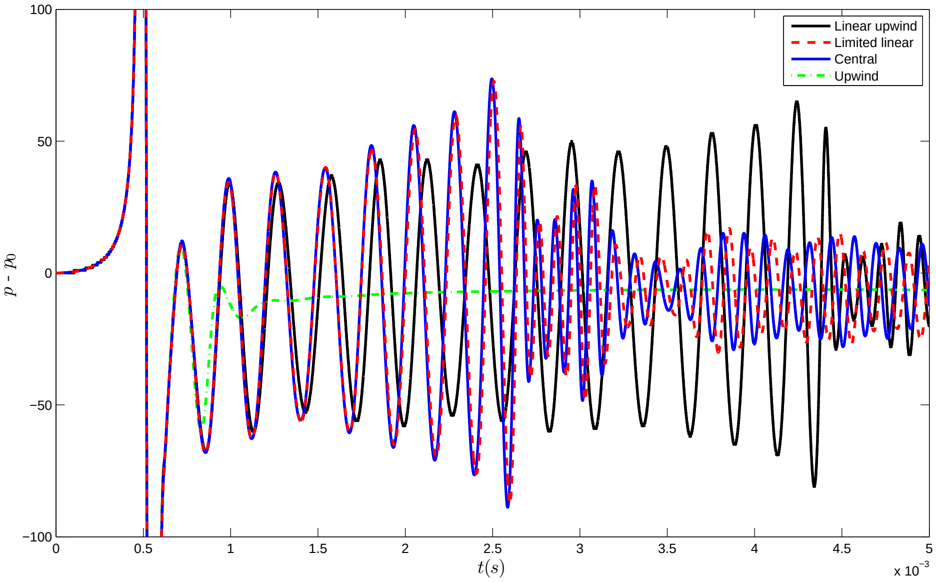


Figure 5.5: Static pressure history (in Pa) at the point $x = 50r_0$, $y = 0$ as function of time for several discretizations of the convective terms.

For time integration the backward scheme was used. The influence of the time step is shown in Fig. 5.6. Clearly, the evolution of the vortices is far less dependent on the Courant number than on the spatial discretization. This is because of the three outer correctors, which greatly increase the stability of simulation. Operating the solver in PISO mode causes the solver to quickly blow up, even at low Courant numbers. The reference pressure history is shown in Fig. 5.7. It is similar to the presented results.

---

[1] Relaxation factors were found to strongly influence the rotation period. The common choices of 0.7 and 0.3 for U and p respectively increased it by roughly 20 percent. Without relaxation factors the rotation speed was 3 to 5 percent faster than according to theory.

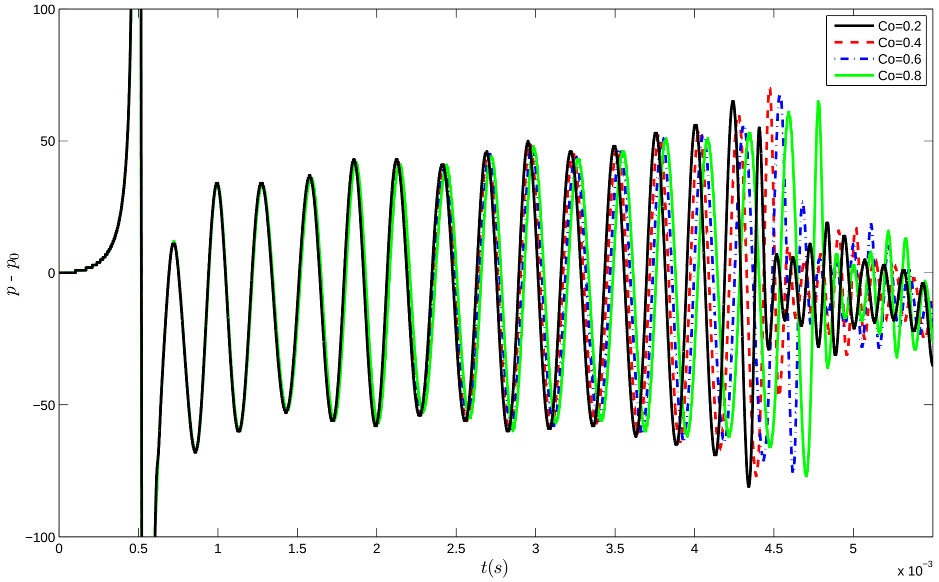


Figure 5.6: Static pressure history (in Pa) at the point $x = 50r_0$, $y = 0$ as function of time for different *initial* Courant numbers.

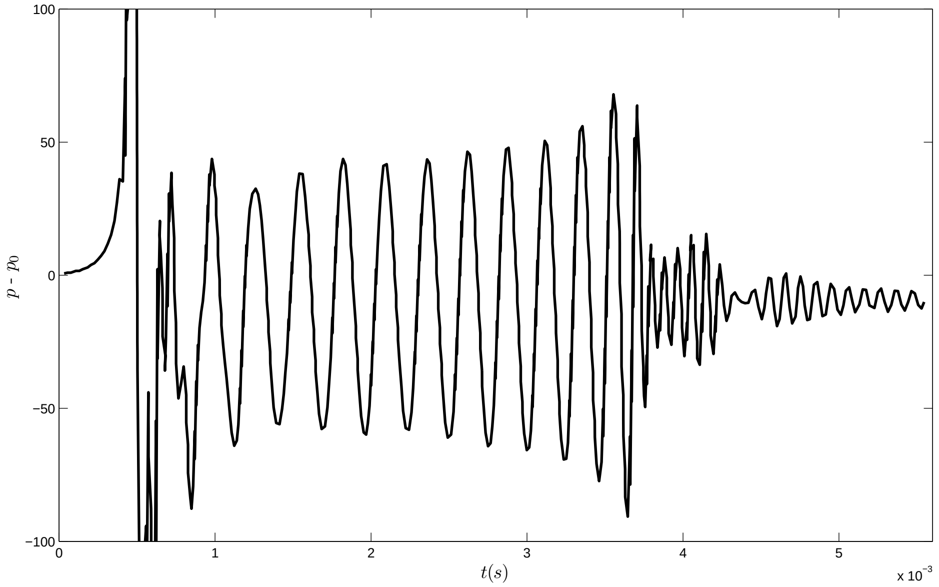


Figure 5.7: Reference pressure history.

Prior to the clearly distinguishable co-rotation phase and merging/merged phases (diminished radiation), a very high peak is visible in the pressure history. This initial transient is due to the initial conditions. The wave is not of importance for the hydrodynamic field, but it has a profound impact on the acoustic field. The transient poses not only a large problem for the boundary conditions, it is also the culprit behind reflections in the internal domain.

5.1.5 Acoustic results

To view the acoustic field, in the reference as acoustic variable the dilatation is used:

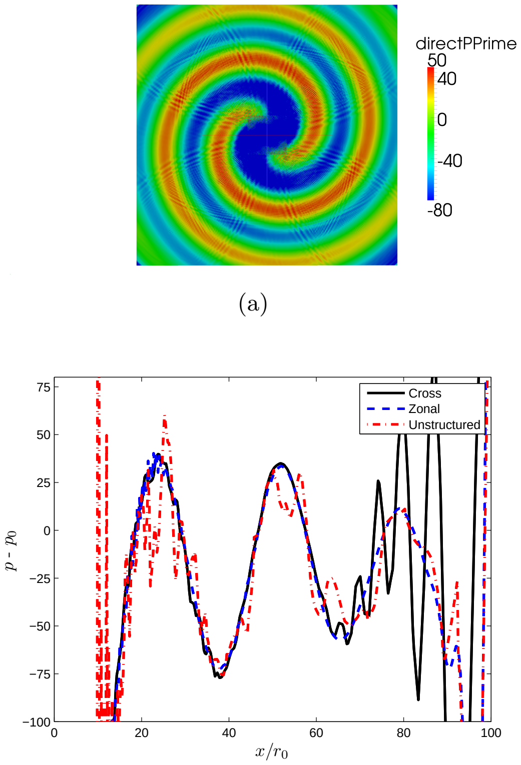
$$\Theta = \nabla \cdot \mathbf{u} = -\frac{1}{\rho_0}\frac{\partial \rho'}{\partial t}. \tag{5.2}$$

Thus the divergence of the direct simulation may be compared to the time derivative of the normalized acoustic density fluctuations (of a hybrid method). The reason why the dilatation is used is because in one of the applications in the reference paper there is a pressure drift, i.e. in $p - \bar{p}$, $\bar{p}$ is not constant. For the stagnant medium case, this does not occur and we will also use $p - \bar{p}$ as acoustic pressure for the direct simulation to later compare it with the acoustic pressure obtained from the LEE simulation. This prevents an additional dependence of the results on the numerical scheme.

As mentioned before, several meshes were tested. The first mesh that was considered was the zonal-type of mesh made using OpenFOAM's blockMesh and refineMesh utilities. A row of triangular cells is formed at the border of each zone transition, after which the rectangular cell size is doubled. The acoustic field with this type of mesh was found to be of poor quality. Although the expected pattern for the acoustic field could be seen (a spiral), the presence of each zone caused reflections (predominantly from the strong initial wave) and high frequency noise was formed near the corners of each transition, see Fig. 5.8(a). Next, cross shaped meshes as illustrated in Fig. 5.3(b) were tested, in the hope that carefully stretching the cells would solve the problem of internal reflections. The meshes of this type were programmed in blockMesh using the codeStream library. A stretching ratio of 4% was initially used and later reduced to 1.5% resulting in a mesh with roughly 0.7 million cells for a domain that extends $100r_0$ from the center. However, even with this stretching ratio numerical noise was present. The simulation was also performed on an unstructured mesh build around the uniform, structured box in the center of the domain. At the boundary of these two regions, the triangles were given the same characteristic length as the uniform cells in the square. This mesh yielded - perhaps not surprisingly - results where one could only vaguely distinguish a spiral pattern underneath a lot of numerical noise. Extracts of $p - p_0$ along the x-axis for these meshes are plotted in Fig. 5.8(b). The large oscillations starting from $x = 70r_0$ are due to the reflection of the strong initial wave with the domain boundary. Note that the average pressure is not equal to p_0. This is again due to the initial wave, which has lowered the pressure of the region through which it has passed.

From the above observations it was concluded that for this particular direct simulation the mesh needs to be as perpendicular as possible to the wavefront. The final mesh therefore has circular rings that are stretched (with a growth rate of 1 percent from one cell to another) in the radial direction around the central box.

For the boundaries, OpenFOAM's characteristic based boundary condition "waveTransmissive" was used. For the earlier described meshes it was found to be insufficient to deal



(a)

(b)

Figure 5.8: Illustration of the problems encountered with some of the early meshes. a) Close-up of the pressure fluctuation w.r.t. the ambient for the zonal mesh and b) pressure fluctuation along the x-axis at $t = 1 \cdot 10^{-3}$ s.

with the initial wave, see Fig. 5.8(b). In particular the corners lead to strong reflections. A slightly better performance was visible on the circular mesh, which is likely due to the one-dimensional (orthogonal) assumption of the boundary condition. The boundary condition was nonetheless found to be not good enough for simulating the acoustic field after the wave has reached the boundary. Therefore an extra region where the cells are stretched was added to the circular mesh. In order to compare with the reference and to later make a comparison with the results from the hybrid method, the direct simulation was carried out on an enlarged domain with radius $300r_0$.

The qualitative result for the simulated dilatation field is shown in Fig. 5.9. The expected acoustic field originating from a rotating quadrupole is found. In the near field thin lines can be seen at the angles 45°, 135°, 225° and 315°, exactly at the borders of the separate blocks that together build up the mesh. The lines can be attributed to the dilatation calculation and do not occur when the acoustic pressure is determined with $p - \bar{p}$. The dilatation along the line $y = x$ is plotted in Fig. 5.10 together with the reference solution [7]. The reference solution was obtained using fourth order schemes for space and time and a coarser mesh (about 80,000 cells) that extends to $104r_0$ from the center. It was not reported whether their solution is the grid-independent solution. The first notable difference with the reference is the first peak, which is much lower.

The amplitude of the other peaks are in reasonable agreement with the reference and the amplitude decay follows that of the two-dimensional wave equation. Looking at the position $50r_0$, the wave runs in phase with the reference, but at $150r_0$ a clear phase difference can be seen. The reference solution is better, keeping a relatively constant wavelength of around $25r_0$.

Based on the extract along the 45° line, the wavelength is underpredicted by the direct method, being at first around $24.5r_0$. The result is a bit different along a line drawn at 43°, see Fig. 5.11. Along this axis, the x-axis and the y-axis the wavelength is initially predicted to be around $26.4r_0$.

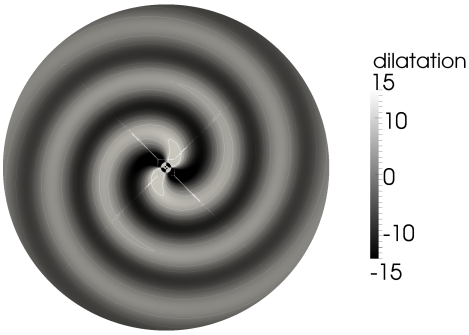


Figure 5.9: Close-up of the OpenFOAM DNS result for the dilatation field. $t = 2.82 \cdot 10^{-3}$ s.

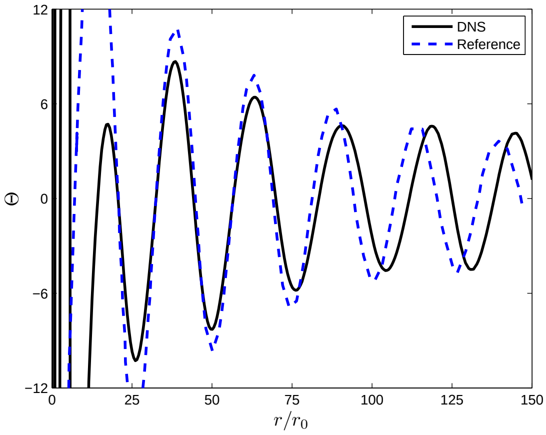


Figure 5.10: Comparison between the DNS and the reference for the dilatation. Extracts along the line $y = x$ at $t = 2.82 \cdot 10^{-3}$ s.

The influence of the grid resolution on the present result is investigated by coarsening the mesh. The mesh is changed by altering the stretching ratio - and therefore the number of cells - of the circular part of the mesh. The obtained dilatation profiles with stretching ratios of 2% and 4% are shown along with the previously discussed result with 1% in Fig. 5.12. The behavior is as expected. As the mesh is coarsened, the waves are damped out faster and dispersion errors become more obvious. Even the strong initial wave is completely negated, without visible reflections. In the present set-up, applying the same

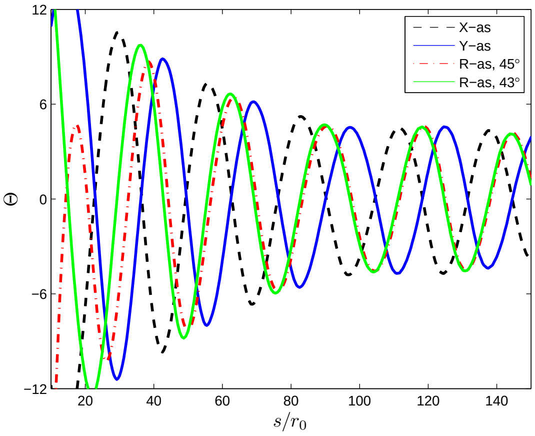


Figure 5.11: Dilatation as function of distance for several axes. $t = 2.82 \cdot 10^{-3}$ s.

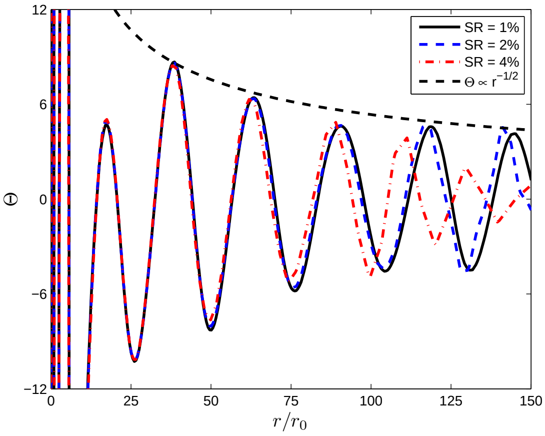


Figure 5.12: Dilatation profiles for various mesh stretching ratios (SR). Extracts along the radial line $y = x$ at $t = 2.82 \cdot 10^{-3}$ s.

stretching ratio of 1% for a domain of $500r_0$ yields only a 13% higher cell count than a domain with an extend of $250r_0$.

5.2 Hybrid Method

In this section the hybrid methodology is discussed. The reconstruction of the acoustic source from the flow simulation is described in section 5.2.1. In section 5.2.2 the results from the LEE simulation are presented. For the validation tests shown here we have initially used the same mesh for flow and acoustic. At the end of the chapter it is shown how source data from a flow simulation done at a relatively small domain can be coupled with an acoustic simulation on a large domain.

5.2.1 Source reconstruction

In case of isentropic flow, entropic sound sources are not taken into account so that the noise generation is provided by the source terms in the momentum equation. For a two-dimensional case the source term as given by Bogey et al. [7] is:

$$\mathbf{S} = \begin{bmatrix} S_1 - \overline{S_1} \\ S_2 - \overline{S_2} \\ 0 \end{bmatrix} \tag{5.3}$$

with

$$S_i = -\frac{\partial \rho u_i' u_j'}{\partial x_j}, \qquad \overline{S_i} = -\frac{\overline{\partial \rho u_i' u_j'}}{\partial x_j}. \tag{5.4}$$

As can be seen, the source term formulation for the LEE is expressed in terms of fluctuating velocity components. This is in contrast to the acoustic analogy of Lighthill, which uses the total velocity. This difference is due to the linear operator on the left hand side. The LEE already account for the convection effect in the operator and thus - unlike in Lighthill's analogy - does not need to include this in the source term. Note that in Eq. (5.4) the instantaneous density is used, meaning that for this case acoustic perturbations are also included. This has however a negligible influence on the LEE calculation.

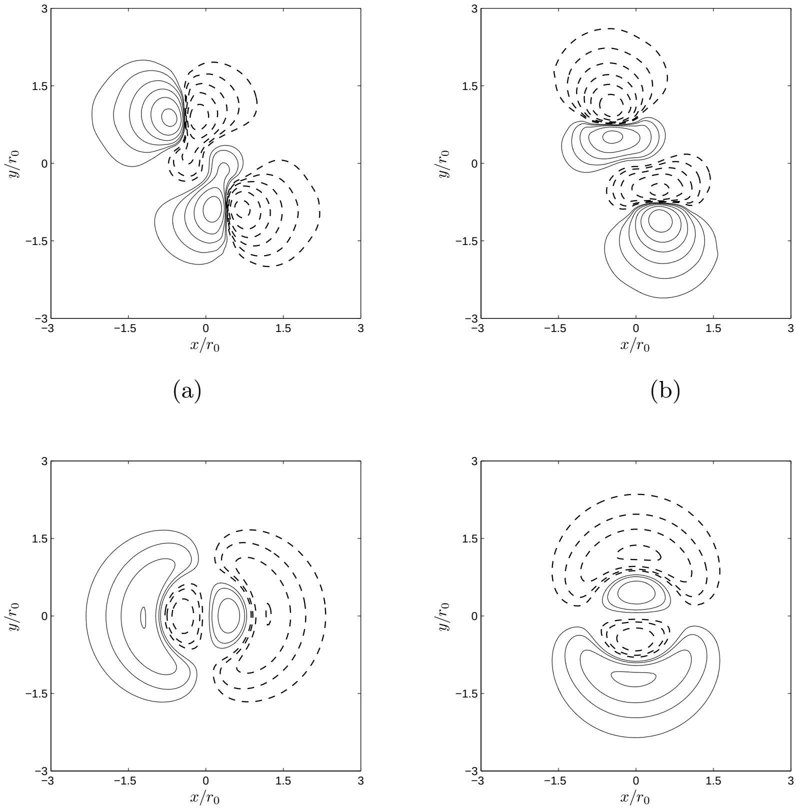
As described earlier, a strong initial wave is emitted at the start up of the simulation. For the hybrid simulation this transient is not included, since the sources are recorded once the flow is fully developed. From Fig. 5.5 the source reconstruction period can be determined. Here we have chosen the time interval from $1 \cdot 10^{-3}$ s until $3 \cdot 10^{-3}$ s.

Iso-contours of the reconstructed acoustic sources are shown in Fig 5.13. These may be compared with those of the reference solution, shown in Fig. 5.14. Since our rotation speed is slightly faster than theoretically prescribed, the iso-contours would have rotated a bit further clockwise with respect to the reference contours. For comparison reasons, the instantaneous contours are instead shown at the position and time at which they theoretically should be.

Both qualitatively and quantitatively good agreement is found.[2]

---

[2] In the original paper it is written that the solid and dashed lines correspond to positive and negative iso-contours respectively. This is a mistake. In the PhD thesis of one of the authors [5] the minus sign in the definition of the source (Eq. (5.4)) has been taken outside of **S**, so that the LEE system is defined as "left hand side = -**S**". The results of the source contours are then in agreement with the present results. Note the (incorrect) small asymmetric deviation in the average source.



(a) (b)

(c) (d)

Figure 5.13: Iso-contours of the acoustic source terms obtained at $t = 1.28{\cdot}10^{-3}$ s using the solver rhoPimpleFoam. Shown are: S_1 (a), S_2 (b), $\bar{S_1}$ (c) and $\bar{S_2}$ (d). Negative values are indicated by the solid line and positive values by the dashed line. Contours from $6 \cdot 10^5$ to $1.1 \cdot 10^7$ $\mathrm{kg} \cdot \mathrm{m}^2\mathrm{s}^{-2}$ with a geometrical ratio of 1.8.

For an efficient application of a source-equivalent hybrid method it is necessary to know how large the source region is. It was found that at $x = 4r_0$ the magnitude of the source $|S|$ has decayed to less than one percent of its maximum value. This means that the flow simulation can be done on a square domain with sides of $8r_0$ and then the source is later propagated with the acoustic solver on a larger domain. In practice, performing the flow simulation on such a small domain results in too much boundary influence. Instead a circular region with stretched cells is added. We will later elaborate on this splitted domain approach. In order to prevent the initial wave to be included in the source term, the source is multiplied with a dummy parameter that is 1 within the central box (plus some extra margin) and 0 outside. This is important when the cross mesh is used. Otherwise a sudden reflection becomes visible when the wave hits the boundaries, which later causes small ripples in the acoustic field.

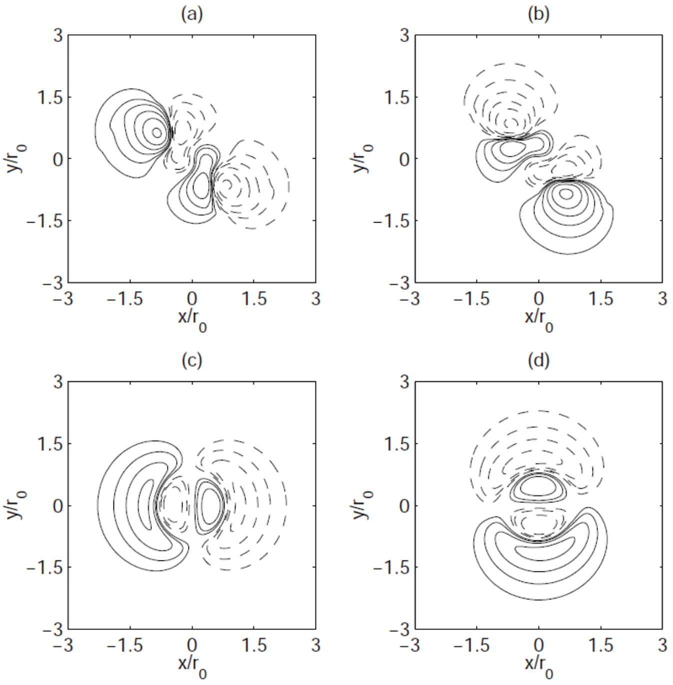


Figure 5.14: Source terms obtained by Bogey et al. [7] at $t = 1.32 \cdot 10^{-3}$ s. Contours from $6 \cdot 10^5$ to $1.1 \cdot 10^7$ $\mathrm{kg \cdot m^2 s^{-2}}$ with a geometrical ratio of 1.8 (the same contours as in Fig. 5.13).

5.2.2 LEE simulation

As described in section 3.2, the acoustic solver does not account for large density jumps. Although the present case concerns compressible flow, the compressibility effects are restricted to the source region. Practically the entire propagation domain is stagnant with uniform density and pressure. Effectively the following is thus solved:

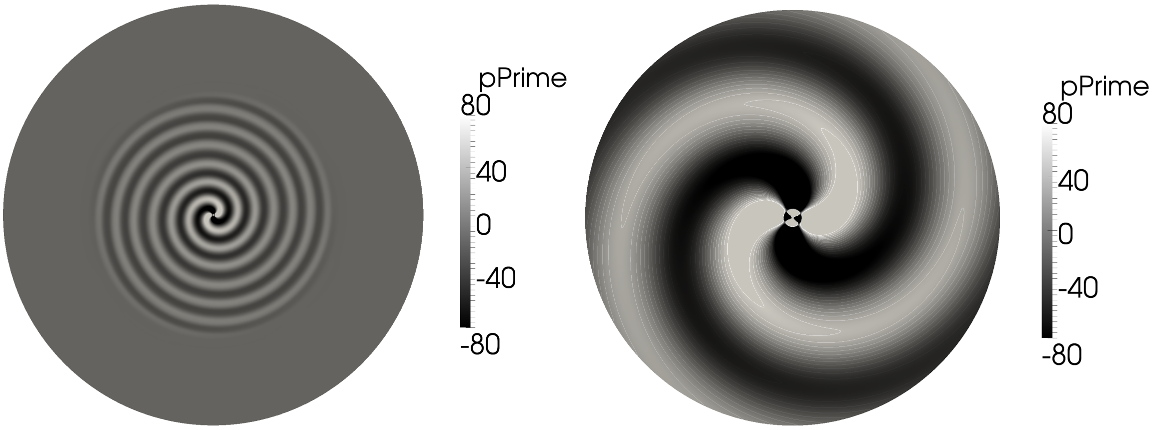
$$\frac{\partial u_i'}{\partial t} + \frac{\partial p_{norm}'}{\partial x_i} = \frac{S_i}{\rho_0}, \tag{5.5a}$$

$$\frac{\partial p_{norm}'}{\partial t} + \frac{\partial}{\partial x_i}(\gamma p_{0,norm} u_i') = 0, \tag{5.5b}$$

with the normalized pressures $p_{norm} = \frac{p_0}{\rho_0}$ and $p'_{norm} = \frac{p'}{\rho_0}$ with $\rho_0 = 1.225\ \mathrm{kg \cdot m^{-3}}$. As mentioned before, the source term includes small variations in the density with an acoustic nature. Although this is strictly not preferred, it does not matter since the fluctuations are so small. Using incorrectly the density ρ_0 for the source term yields an overprediction of roughly 8% of the wave amplitudes. The mean flow is zero, hence solving the LEE is on the differential level equivalent to solving Lighthill's analogy.

Since the flow solver does not use a Runge-Kutta method for time marching and thus does not provide the acoustic source at stagelevels, we are forced to disable the Runge-Kutta loop in the aacFoam solver. It was found that with the standard Euler time integration methods the simulation diverges when for each flow timestep at a Courant number of 0.2 only one acoustic solve is performed. A solution for this is to solve the acoustic equations for every flow timestep at smaller intermediate steps. This can be implemented using OpenFOAM's subcycle class. In this way the acoustic system is solved multiple times with the same source data (0^{th} order interpolation between source levels). Another point of attention is the start of the simulation. At the very beginning of the source injection, two strong initial wavefronts propagate from the vortices. To smoothen the start of the acoustic simulation the source terms were multiplied with a cosine window. The source term is gradually ramped up from zero to full strength during the time interval $1 \cdot 10^{-3}$ s to $1.1 \cdot 10^{-3}$ s.

A snapshot of the acoustic field obtained using the Riemann based LEE solver with temporal ramping and 3 subcycles for each source data level (that was obtained from the flow simulation performed at an initial Courant number of 0.2) can be seen in Fig. 5.15.



(a) Large domain simulation. $t = 2.82 \cdot 10^{-3}$ s.

(b) Small domain simulation. $t = 1.74 \cdot 10^{-3}$ s.

Figure 5.15: Acoustic pressure from the LEE simulation using a) the same (too) large domain that was used for the DNS and b) a small domain.

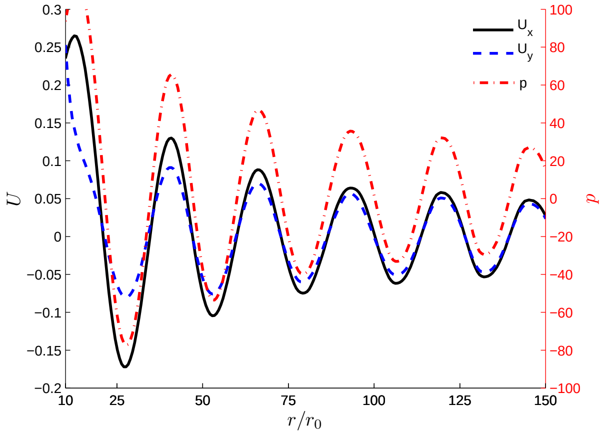


Figure 5.16: Acoustic variables from the LEE simulation extracted along the radial line $y = x$ at $t = 2.82 \cdot 10^{-3}$ s.

The expected spiral pattern is found. A plot of the variables p' and $\mathbf{u}'$ along a radial line is shown in Fig. 5.16. As can be seen, the iso-contours of p' and the extract along the line are very smooth.

For the sake of comparing the LEE solution with the reference, it would be necessary to take the time derivative of the acoustic pressure (Eq. (5.2)). When this is added in OpenFOAM small wiggles occur in the near field and the wave amplitudes seem very far off. For the phase of the wave it does not matter whether one considers $p - p_0$ or the dilatation, because the graphs only differ by an amplification factor. We therefore compare the LEE results only with our direct simulation. Several simulation settings are varied so that we at least know how sensitive the presented solution is to the set-up.

In Fig. 5.17 the DNS, LEE (with and without time-averaged source $\bar{S}$) and Lighthill solutions are plotted along the x-axis. As can be seen there is a phase difference between the hybrid methods and the DNS calculation, which remarkably already exists in the region $x < 25r_0$. The Lighthill simulation is very close to the LEE simulation (without $\bar{S}$), which is expected. Since the source data is the same, both methods should yield in the limit of a very fine space and time discretization the same result. It also means that the difference with the DNS is less likely due to the wave propagation performance of the hybrid methods. In fact, it turns out that the difference already exists at the very beginning of the simulations. It is postulated that the difference has to do with the start-up phases of the DNS and/or the LEE (we do know though that the late start of the LEE simulation or the ramping function are not the cause). From Fig. 5.10 is was observed that the wavelength predicted by the DNS does not remain constant. For this case, only a slightly better performance is seen with the LEE. This is most likely because the source data itself differs per rotation. A 3 percent error in the rotation period for example would become a 6 percent error in the wavelength. As was already noted in [7], the source term $\bar{S}$ produces a steady pressure field that decays much faster than the acoustic waves. This can also been by looking at the difference between the maxima of the LEE simulations with and without the average source. The effects of including $\bar{S}$ in the LEE are that it centers the source term so that the wave oscillates around zero, it decreases the amplitude of the initial wave (although temporal ramping is much more effective) and it corrects the near field solution.

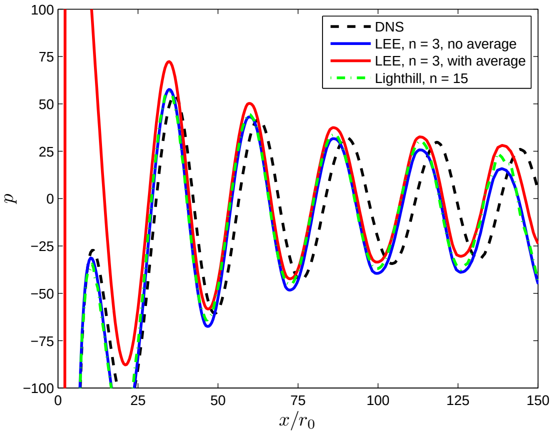


Figure 5.17: Direct and hybrid solutions for p'. $t = 2.82 \cdot 10^{-3}$ s.

In the previous result five times as much subcycles were used for Lighthill's analogy than for the LEE. The reason why 15 cycles were used for the Lighthill simulation is shown in Fig. 5.18. The temporal error is very apparent when fewer cycles are used. When six cycles are used the wave amplitude is twice as large as that of the solution with 15 cycles and the wave is out of phase. The second derivative with respect to time is only first order in OpenFOAM. Furthermore, it was observed that when e.g. 12 cycles are used, the solution is anisotropic. The pressure signal then has a higher amplitude along the x-axis than along the y-axis. For the LEE and DNS calculations the signals along the x and y-axes were 90° out of phase and had the same amplitude level.

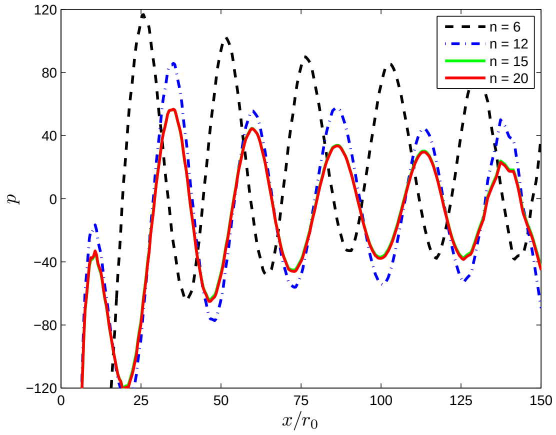


Figure 5.18: Influence of the number of subcycles on the Lighthill solution. Flow data obtained at an initial Courant number of 0.2. $t = 2.82 \cdot 10^{-3}$ s.

The influence of the number of subcycles on the LEE is investigated with saved source data. The flow simulation was done with an initial Courant number of 1. In Fig. 5.19 the results are shown. When too few cycles are used, large oscillations occur, eventually leading to a diverged simulation.

A grid study was done with the LEE. From the circular part of the original grid we removed after every point, one point (2.5% stretching), two points ($\approx$ 4% stretching) and three points (5.8% stretching). The results are shown in Fig. 5.20 along with the result obtained with the original mesh. The solution behaves the same as for the verification cases: as the grid is coarsened, the amplitude is reduced but the phase remains similar. In addition the solution with a finer mesh (0.5% stretching) is shown. The refined mesh has around 0.9 million cells, while the baseline mesh has around 0.6 million cells. The improvement is not worth the extra computational cost. The solution with the cross mesh is shown in Fig. 5.21. It is nearly identical to the result with the baseline circular mesh. Unlike with the DNS, the solution is not spoiled by reflections from the corners of the domain. Moreover, since the initial wave is not present in the hybrid method, internal reflections no longer occur.

In Fig. 5.22(a) the influence of the timestep of the DNS on the LEE simulation (via changes in the source data) is shown. There exists a phase difference between the results. This is most likely due to the small change in the rotation speed of the vortices. The vortices rotate slower when the simulation is carried out with a Courant number of unity

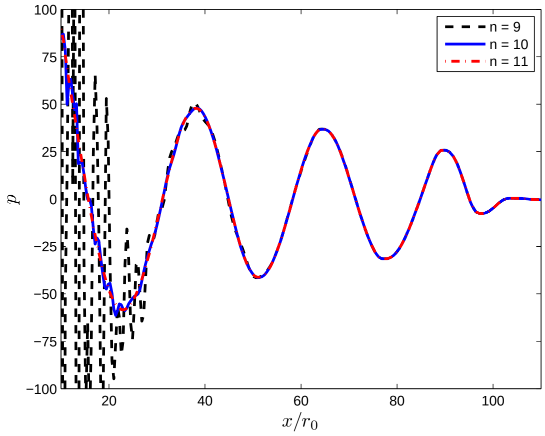


Figure 5.19: Influence of the number of subcycles on the LEE solution for p'. Flow data obtained at an initial Courant number of 1. $t = 2.05 \cdot 10^{-3}$ s.

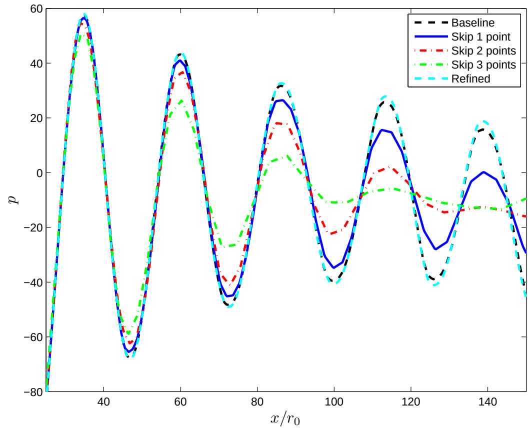


Figure 5.20: Gridstudy with the LEE (without $\bar{S}$). $t = 2.82 \cdot 10^{-3}$ s.

than with a Courant number of 0.2. This results in a phase lag.

The influence of skipping source data is checked. This is relevant for diskspace usage. Here we saved the acoustic source every timestep, with a timestep corresponding to an initial Courant number of unity. Early tests were done from skipping one data level up to five datalevels, while adjusting the number of subcycles accordingly. Practically no influence was visible, which is not so surprising if one realizes that almost five datalevels are used for representing a single degree of rotation of the vortices. In Fig. 5.22(b) the results when skipping 10 steps (1 datalevel per 2 degrees), 25 steps (1 datalevel per 5 degrees) and 100 steps (one source datalevel for every 20 degrees) are shown. As expected, the solution worsens. When skipping 100 steps, waves are along the x-axis still correctly send to the far field. However, the contourplot has lost all resemblance of a spiral. Though it appears that we may safely use an output interval of 10 for the present case. This information was used in the next test.

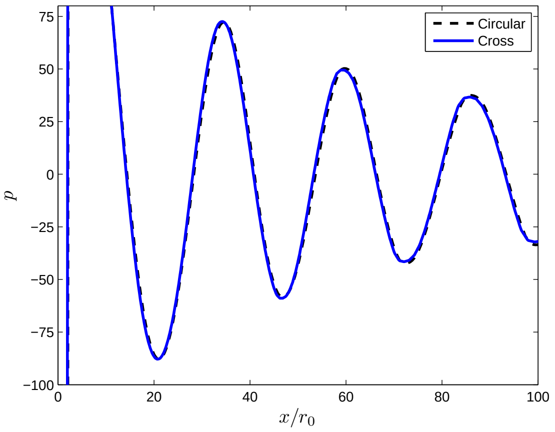
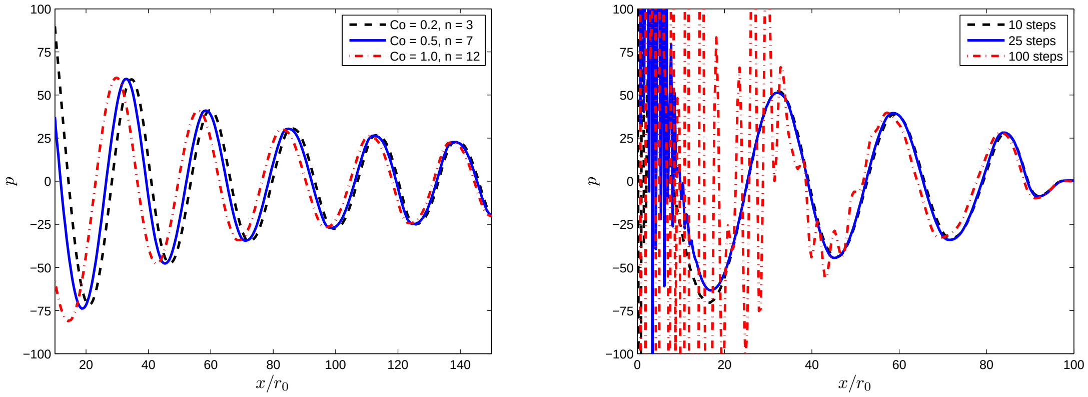


Figure 5.21: Different mesh types. LEE with $\bar{S}$ at $t = 2.82 \cdot 10^{-3}$ s.



(a) Different Courant numbers. $t = 2.82 \cdot 10^{-3}$ s.

(b) Skipping source data. $t = 2.05 \cdot 10^{-3}$ s.

Figure 5.22: Influence of the timestep and skipping source data on p'.

Lastly, it was investigated whether one could do the flow simulation on a smaller domain than the acoustic simulation. This could make the method more efficient.

Initially the `meshToMesh` class was used to interpolate the source data from the flow mesh to the acoustic mesh. This works for single core computations, but it seemed not so easy to make it work for parallel computations. The present approach is illustrated in Fig. 5.23 for a situation where the domain is decomposed and distributed over 4 processors. We distinguish between “source” cells (flow mesh) and “target” cells (acoustic mesh). From each processor, local source cell index and cell center data is extracted. This data is combined into a “total” cell list, which would correspond to the situation before the mesh was decomposed. Each processor now loops over this list. If for a given source cell, the source cell coordinates lie within the boundaries of the *local* target mesh, then the corresponding target cell is searched for and the indices are linked. Currently, the flow mesh needs to be a subset of the acoustic mesh (i.e. cell centers need to coincide), otherwise a cell can not be found and consequently the source will later not be mapped for that cell. This procedure is done once. The actual data mapping is done every timestep during the time loop. We tested this with a flow domain that extends around $20r_0$ from the center (to prevent boundary influence), which reduced the number of cells with 75%. Identical results were found.

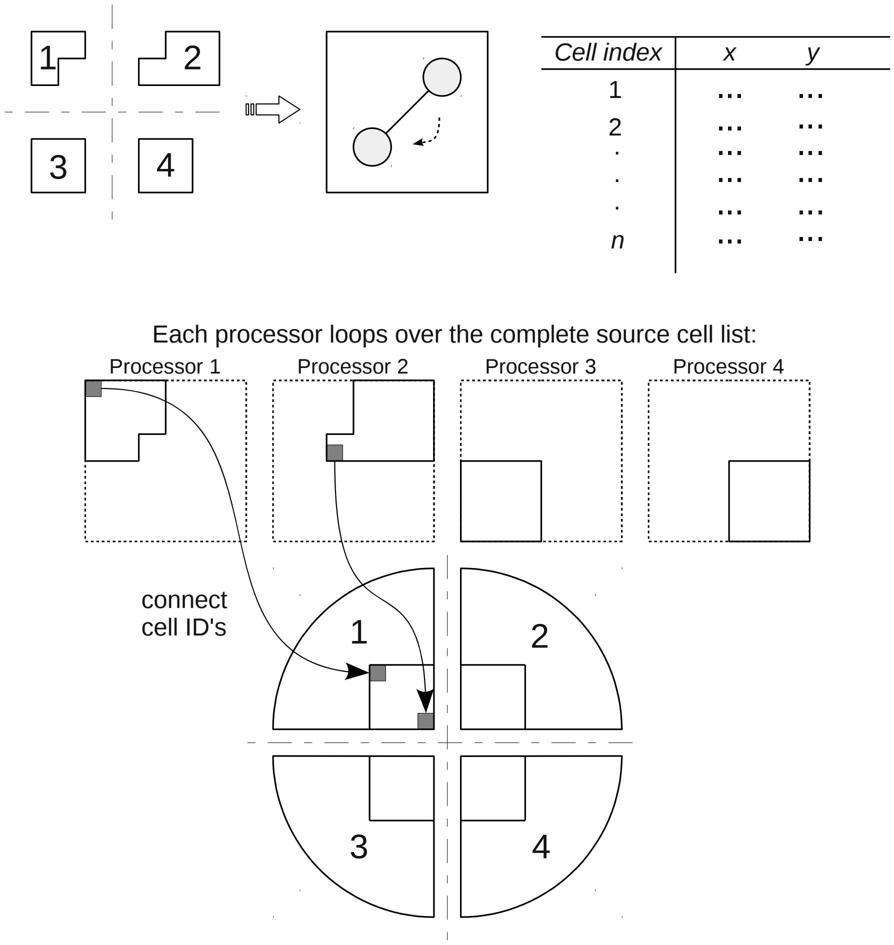

1
2
3
4
Cell index
x
y
1
2
.
.
.
n
...
Each processor loops over the complete source cell list:
Processor 1
Processor 2
Processor 3
Processor 4
connect
cell ID's
1
2
3
4

Figure 5.23: Connecting the flow mesh with the acoustic mesh. This is done outside the time integration loop, while the mapping is done during the time march.

Chapter 6

Noise from a beveled trailing edge

In this chapter results of a beveled trailing edge noise field are obtained using the solver presented in this thesis. The aim is to see whether the solver produces sensible results for such a realistic low Mach number, high Reynolds number flow problem and to identify possible issues with the method that were not apparent during the validation phase. Investigations of trailing edge noise currently receive much attention, especially from wind turbine and aircraft industries, making it a relevant case.

6.1 Flow simulation

An incompressible flow simulation using a Large Eddy Simulation (LES) methodology over a flat plate with an asymmetric beveled 25° trailing edge is conducted, see Fig. 6.1. The numerical set-up of the flow simulation is based on that of [32], with notable differences being the freestream velocity (here 35 m/s instead of 10 m/s), the plate length (0.18 m instead of 0.36 m and the corresponding Reynolds number ($4.7 \cdot 10^5$ instead of $2.68 \cdot 10^5$). Furthermore, the mesh is considerably coarser in the spanwise direction, leading to a flow mesh of around $1 \cdot 10^5$ cells. The inlet velocity is prescribed using a recycling method to generate a turbulent boundary layer at the inlet. The reader is refered to [32] for detailed descriptions.

The LES is run for 0.1 s ($\approx$ 10 flow through times) to establish a converged mean flow, see Fig. 6.2. It can be seen that the flow separates at the kink of the top side. This is due to the adverse pressure gradient that the boundary layer experiences. Consequently, a shear layer is formed (a recirculation region is visible close to the slanted surface). On the bottom side, the flow separates at the trailing edge. The vortex shedding of the instantaneous velocity field is visualized in Fig. 6.3 by means of the Q-criterion. The frequency of the shedding is approximated by measuring the streamwise distance in the wake between two maxima of the sinusoidal wave. This gives $f = \frac{u_\infty}{\Delta x} = \frac{35}{0.08} \approx 438$ Hz, where it is assumed that the convection speed is close to the freestream value. Based on the thickness of the plate ($h = 20$ mm), the Strouhal number becomes $St = \frac{fh}{u_\infty} \approx 0.25$, which is in good agreement with earlier studies [13, 31].

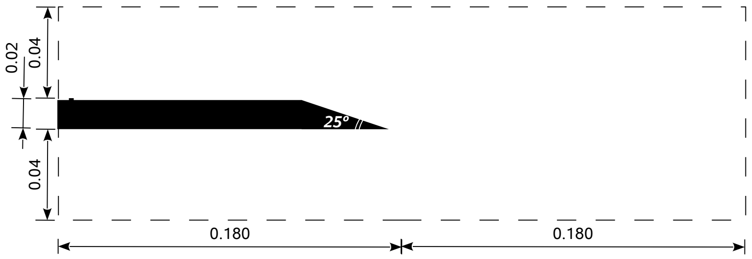

0.02
0.04
0.04
25°
0.180
0.180

Figure 6.1: Schematic of the flow domain. Distances in meter.

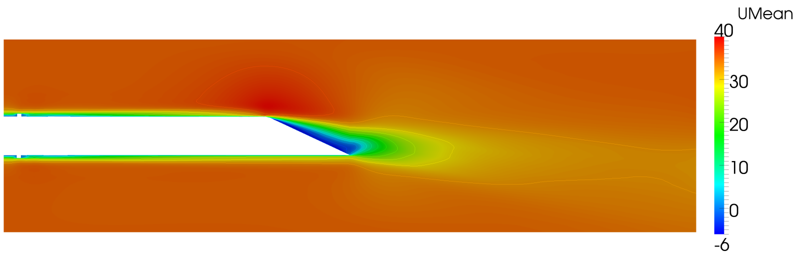


Figure 6.2: Time-average of the x-velocity component.

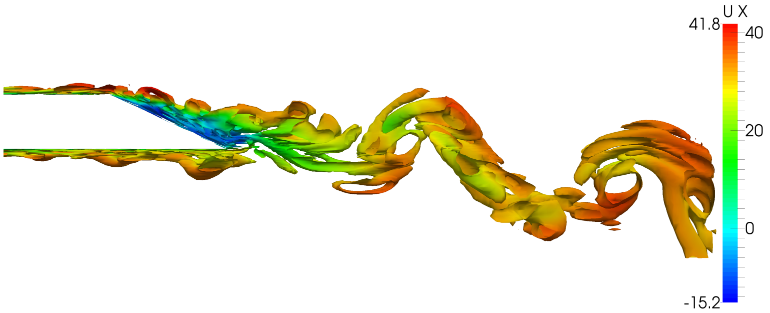

U X
41.8
40
20
0
-15.2

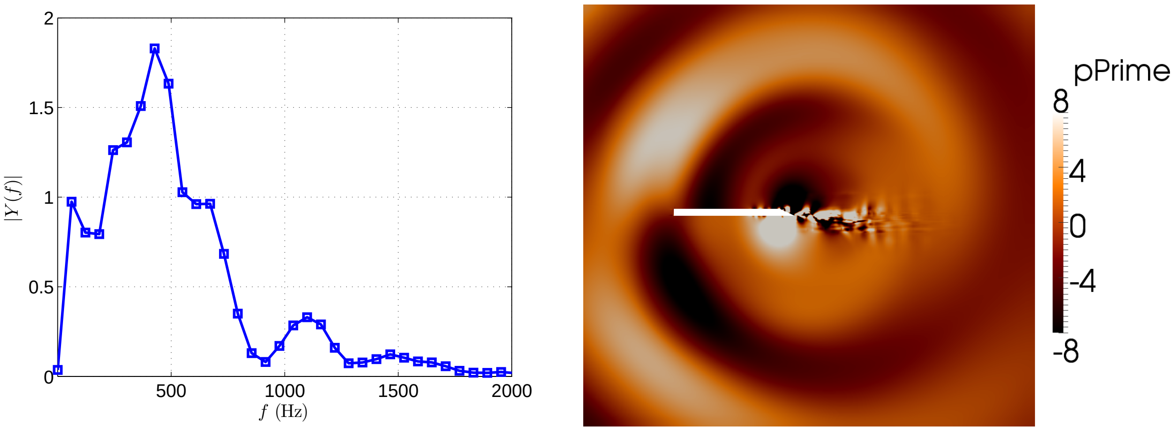
Figure 6.3: Snapshot of the $Q = 1{\cdot}10^5$ iso-surface. Colored by the instantaneous streamwise velocity component.

6.2 Acoustic simulation

The acoustic variables are calculated on a larger domain that has a height of 0.63 m w.r.t. the trailing edge. The mesh is progressively coarsened starting from the x-location 0.18 m (downstream of the plate), because the convective outlet B.C. alone was not good enough to fully prevent reflections from the strong pressure fluctuations in the wake.

The acoustic and flow simulation are now done simultaneously, with an one-on-one mapping of the acoustic source onto the acoustic mesh. The flow calculation is restarted with the velocity and pressure solutions obtained at 0.1 s as initial field. The previously obtained mean statistics for velocity and pressure serve as background flow in the acoustic part of the calculation. For the present case, the acoustic simulation would however diverge when the terms collected into $\mathbf{H}$ were at the left-hand-side of the LEE (Eq. (3.14)). In order to still account for part of these refraction effects, $\mathbf{H}$ is modeled as a source term. This approach was also taken in [18] to preserve numerical stability.

An impression of the obtained acoustic field is shown in Fig. 6.4(b). It can be seen that strong waves are directed upstream. A probe that records the acoustic pressure history was placed two chords above the trailing edge. The resulting one-sided frequency spectrum of the acoustic signal is shown in Fig. 6.4(a). The shedding frequency of the wake is expected to appear as a peak in the frequency spectrum. Although a peak around 430 Hz can be seen, there is a range of important frequencies around this peak. This is possibly due to the lack of samples as well as insufficient resolution in the flow simulation.



(a) Frequency spectrum.

(b) Snapshot of p'.

Figure 6.4: Acoustic solution.

Chapter 7

Conclusions and recommendations

The primary objective of the thesis was to develop an open-source tool that can predict the acoustic field from (incompressible) flow simulations, correctly accounting for one-way flow-acoustic interactions. With "developing", implementation, verification and validation is meant. For this purpose, an aeroacoustic solver based on the linearized Euler equations (LEE) was implemented in OpenFOAM. The thesis proposal suggested that a Riemann problem should be set-up for the calculation of the fluxes between cells. In the next two sections the conclusions and recommendations from the investigation of the above are given.

7.1 Conclusions

A standard LEE and a Riemann based LEE solver were implemented in OpenFOAM. The LEE implementations were both verified and the performance of the solvers was assessed. Compared to the standard implementation, the Riemann based solver performs better regarding dispersion, but is more diffusive. Good dispersive properties will be in particular important when noise frequencies need to be identified, but its diffusive character is mostly a disadvantage. The code works for structured as well as unstructured meshes.

A thorough investigation of the LEE based hybrid method applied to the motion of two co-rotating vortices was performed. We have shown that with the developed solver and the given flow data, the presented acoustic solutions are *the* solution. The flow simulation was found to be quite sensitive to the set-up and small differences become directly apparent in the acoustic results. Although consequently some discrepancies in the wavelength are present, we may safely say that the hybrid method is validated. One notable aspect for the LEE simulation is the required number of subcycles per flow timestep. With our explicit implementation, the simulation diverged when to few cycles were used. The importance of time integration was also seen with the Lighthill simulations, where seemingly good results were obtained with a few subcycles, while these were actually completely wrong.

Although the co-rotating vortices case is purely an academic case, we investigated how efficiency gains could be made. It was shown that it is possible with OpenFOAM to do the flow and acoustic calculations on different meshes and then transfer the source data. This can be done both during the CFD calculation, as well as afterwards if source data is saved. The data is mapped without interpolation. This was applied to the last case where trailing edge noise from a beveled flat plate was investigated. Preliminary results from this case indicate that it is possible to apply the code on such a complex case.

7.2 Recommendations

In this thesis the focus was implementation of the solver and testing it on classical benchmark cases. Possible future investigations are the following:

- The solver needs to be tested on progressively more realistic cases. Some suggestions are a flat plate, forward facing step and NACA 0012 airfoil. The quality of the acoustic simulation depends greatly on the input of the flow simulation, so it is wise to start with a case where the flowfield can be determined with confidence (and preferably without a too large mesh). One could compare the obtained results with those of e.g. Curle's analogy.
- The transfer of source data from the flow mesh to the acoustic mesh needs to be improved. Currently, the flow domain must be a subset of the acoustic domain so that for low Mach number cases an unnecessary fine grid is used near walls for the acoustic simulation.
- Further efficiency gains with the hybrid method are possible. When considering a 3D turbulent flow case, often the acoustic is calculated in a 2D plane using spanwise averaged data. Afterwards the sound pressure levels are then corrected to correspond to 3D wave propagation. This approach greatly increases the efficiency of the acoustic simulation, because the mesh contains a lot less cells. It would thus be worth to investigate whether such a spanwise averaging code (that works in parallel) can be implemented efficiently.

References

[1] C. Bailly and D. Juvé. Numerical solution of acoustic propagation problems using linearized euler equations. *AIAA Journal*, 38:22–29, 2000.

[2] C. Bailly, P. Lafon, and S. Candel. A stochastic approach to compute noise generation and radiation of free turbulent flows. In *AIAA Paper 95-092*, pages 669–674, June 1995.

[3] T. Barth and D. Jespersen. The design and application of upwind schemes on unstructured meshes. *AIAA-89-0366*, 1989.

[4] C.P.A. Blom. *Discontinuous Galerkin Method on Tetrahedral Elements for Aeroacoustics.* PhD thesis, University of Twente, 2003.

[5] C. Bogey. *Calcul direct du bruit aerodynamique et validation de modeles acoustiques hybrides.* PhD thesis, Ecole Central de Lyon, 2000.

[6] C. Bogey and C. Bailly. Three-dimensional non-reflective boundary conditions for acoustic simulations: far field formulation and validation test cases. *Acta Acustica*, 88:463–471, 2002.

[7] C. Bogey, C. Bailly, and D. Juvé. Computation of flow noise using source terms in linearized Euler's equations. *AIAA Journal*, 40:235–243, 2002.

[8] T. Colonius and S.K. Lele. Computational aeroacoustics: progress on nonlinear problems of sound generation. *Progress in Aerospace Sciences*, 40:345–416, 2004.

[9] J. Delfs. Basics of aeroacoustics, 2013.

[10] P. di Francescantonio. A new boundary integral formulation for the prediction of sound radiation. *Journal of Sound and Vibration*, 202:491–509, 1997.

[11] R. Ewert and W. Schröder. Acoustic pertubation equations based on flow decomposition via source filtering. *Journal of Computational Physics*, 188:365–398, 2003.

[12] M.E. Goldstein. An exact form of Lilley's equation with a velocity quadrupole/temperature dipole source term. *Journal of Fluid Mechanics*, 443:231–236, 2001.

[13] M. Greenway and C. Wood. The effect of a beveled trailing edge on vortex shedding and vibration. *Journal of Fluid Mechanics*, 61:323–335, 1973.

[14] J.C. Hardin, J.R. Ristorcelli, and C.K.W. Tam, editors. *ICASE/LaRC Workshop on benchmark problems in computational aeroacoustics (CAA)*, NASA conference publication 3300, May 1995.

[15] O. Inoue and N. Hatakeyama. Sound generation by a two-dimensional circular cylinder in a uniform flow. *Journal of Fluid Mechanics*, 471:285–314, 2002.

[16] H. Jasak. *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows.* PhD thesis, Imperial College London, 1996.

[17] Jr. John D. Anderson. *Fundamentals of aerodynamics.* McGraw-Hill, 2006.

[18] L. Kapa. *Numerical prediction of noise production and propagation.* PhD thesis, Université Libre de Bruxelles, 2011.

[19] P. Lafon. *Aéroacoustique des écoulements subsoniques et transsoniques.* PhD thesis, Ecole Centrale de Lyon, 2011.

[20] R.J. LeVeque. *Finite-Volume Methods for Hyperbolic Problems.* Cambridge, 2004.

[21] M.J. Lighthill. On sound generated aerodynamically, part 1: General theory. *Proceedings of the Royal Society of London*, A211:564–587, 1952.

[22] B.E. Mitchell, S.K. Lele, and P. Moin. Direct computation of the sound from a compressible co-rotating vortex pair. *Journal of Fluid Mechanics*, 285:181–202, 1995.

[23] T.J. Poinsot and S.K. Lele. Boundary conditions for direct simulations of compressible viscous flows. *Journal of computational physics*, 101:104–129, 1992.

[24] A. Powell. Theory of vortex sound. *Journal of the Acoustical Society of America*, 36:177–195, 1964.

[25] S. Rienstra and A. Hirschberg. An introduction to acoustics, 2013.

[26] C. Tam. Computational aeroacoustics: An overview of computational challenges and applications. *International Journal of Computational Fluid Dynamics*, 18:547–567, 2004.

[27] C.K.W. Tam and J.C. Webb. Dispersion-relation-preserving finite-difference schemes for computational acoustics. *Journal of Computational Physics*, 107:262–281, 1993.

[28] K.W. Thompson. Time dependent boundary conditions for hyperbolic systems i. *Journal of Computational Physics*, 68:1–24, 1987.

[29] K.W. Thompson. Time dependent boundary conditions for hyperbolic systems ii. *Journal of Computational Physics*, 89:439–461, 1990.

[30] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics.* Springer-Verlag, 2009.

[31] W.C.P. van der Velden, S. Probsting, A.T. de Jong, and A.H. van Zuijlen. Numerical and experimental investigation of a beveled trailing edge flow and noise field.

[32] W.C.P. van der Velden, A.H. van Zuijlen, A.T. de Jong, and H. Bijl. On the aeroacoustic properties of a beveled plate. In *E3S Web of Conferences*, 2015.

[33] M. Wang, J.B. Freund, and S.K. Lele. Computational prediction of flow-generated sound. *Annual Review of Fluid Mechanics*, 38:483–512, 2006.

[34] M. Wang, S.K. Lele, and P. Moin. Computation of quadrupole noise using acoustic analogy. *AIAA Journal*, 34:2247–2254, 1996.

Appendix A

Green’s function

Since in this thesis multiple times reference is made to a “Green’s function”, a small clarification is provided here. The concept of a Green’s function for application to an acoustic analogy is briefly outlined, following the reasoning of Delfs [9].

Consider an inhomogeneous wave equation such as Lighthill’s equation (Eq. (2.1)) in free field conditions:

$$L\{p'\} = S(\mathbf{x}, t). \tag{A.1}$$

L is the linear differential operator (in this case: $1/c_0^2(\partial^2/\partial t^2) - \nabla^2$) and $S(\mathbf{x}, t)$ the source term. In order to solve Eq. (A.1) a simplified problem is considered:

$$L\{\mathcal{G}\} = \delta(\mathbf{x} - \xi)\delta(t - \tau). \tag{A.2}$$

$\mathcal{G}$ is Green’s function and it is defined as the acoustic field generated at an observer’s location $\mathbf{x}$ at time t due to an impulse of Delta functions at position ξ and time τ. Multiplying Eq. (A.2) with the source term and integrating over a volume and time yields

$$\begin{aligned}\int\limits_{V}\int\limits_{-\infty}^{t'} L\{\mathcal{G}\}S(\xi, \tau)\mathrm{d}\tau\mathrm{d}V(\xi) &= \int\limits_{V}\int\limits_{-\infty}^{t'} \delta(\mathbf{x} - \xi)\delta(t - \tau)S(\xi, \tau)\mathrm{d}\tau\mathrm{d}V(\xi) \\ &= S(\mathbf{x}, t).\end{aligned} \tag{A.3}$$

The upper limit on the time integration t' indicates that at time t at the observer location p' only depends on past information. Therefore $\iint L\{\mathcal{G}\}S(\xi, \tau)\mathrm{d}\tau\mathrm{d}V(\xi)$ must be equivalent to $L\{p'\}$. Since L is linear and acts on $\mathbf{x}$, the operator can be taken outside of the integral so that

$$p' = \int\limits_{V}\int\limits_{-\infty}^{t'} \mathcal{G}(\mathbf{x}, t; \xi, \tau)S(\xi, \tau)\mathrm{d}\tau\mathrm{d}V(\xi). \tag{A.4}$$

For a given source the solution can thus be obtained by convolution of the source term with the Green’s function over space and time. The Green’s function itself is known from the solution to the point source problem given by Eq. (A.2). It turns out that the Green's function for a wave equation with an arbitrary source is:

$$\mathcal{G}_0(\mathbf{x}-\xi, t-\tau) = \frac{\delta(t-\tau-\|\mathbf{x}-\xi\|/c_0}{4\pi\|\mathbf{x}-\xi\|}. \tag{A.5}$$

$\|\mathbf{x}-\xi\|$ is the distance of the source to the observer. Since we did not specify any boundary conditions, the Green's function is called a free field Green's function, $\mathcal{G}_0$, as opposed to "tailored" Green's functions. Different Green's functions exist for different differential operators. Upon inserting Eq. (A.5) into the formal solution Eq. (A.4) the following is found:

$$\begin{aligned} p' &= \int\limits_{V}\int\limits_{-\infty}^{\infty} \frac{\delta(t-\tau-\|\mathbf{x}-\xi\|/c_0}{4\pi\|\mathbf{x}-\xi\|} S(\xi,\tau)\mathrm{d}\tau\mathrm{d}V(\xi) \qquad \text{(A.6)} \\ &= \frac{1}{4\pi}\frac{\partial^2}{\partial x_i \partial x_j}\int\limits_{V} \frac{T_{ij}(\xi, t-\|\mathbf{x}-\xi\|/c_0)}{\|\mathbf{x}-\xi\|}\mathrm{d}V(\xi), \end{aligned}$$

where the latter solution is specific for the Lighthill analogy.

Appendix B

Linearization of the Euler equations

In this appendix *one* way to derive the linearized Euler equations of [7] is shown. This is a specific form of the LEE. Various other forms of the LEE exists and are all applicable to linear aeroacoustics. For a very complete discussion on the linearization of Euler equations, the reader is referred to the PhD dissertation of Blom [4].

B.1 Governing equations of fluid flow

B.1.1 Navier-Stokes equations

The most general set of equations that govern fluid motion are the Navier-Stokes equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = S_c \tag{B.1a}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij} - \tau_{ij}) = S_{m,i} \tag{B.1b}$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_j}(\rho H u_j - u_i \tau_{ij} + q_j) = S_e, \tag{B.1c}$$

where ρ is the density, u the velocity, p the pressure, τ_{ij} the viscous part of the stress tensor, E the total energy, H the total enthalpy, q the heat flux due to heat conduction and S_c, S_m and S_e are the source terms in the continuity, momentum and energy equation, respectively.

The total energy E is the sum of the internal energy and specific kinetic energy and the total enthalpy H is given by:

$$E = e + \frac{1}{2}u_k u_k, \tag{B.2}$$

$$H = E + \frac{p}{\rho}. \tag{B.3}$$

τ_{ij} is given by:

$$\tau_{ij} = \mu\Big(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\Big) + \delta_{ij}\lambda \mathrm{div}\mathbf{U}. \tag{B.4}$$

For the last term often Stokes' hypothesis ($2\mu + 3\lambda = 0$) is utilized. μ is a function of temperature. Fourier's law can be used as constitutive relation for the heat flux q:

$$q_i = -k\frac{\partial T}{\partial x_i}, \tag{B.5}$$

with the thermal conductivity coefficient $k = k(T)$. Lastly, the source terms: S_c may represent mass injection, S_m body forces and S_e work from forces introduced with S_m or a heat source.

The system given by Eqs. (B.1) has 7 unknowns (ρ, u, E, p and T), but only 5 equations. To close the system, 2 extra thermodynamic relations are used:

$$\rho = \rho(T, p) \qquad e = e(T, p). \tag{B.6}$$

For a calorically perfect gas these become:

$$\rho = \frac{p}{RT} \qquad e = c_v T, \tag{B.7}$$

with R the gas constant and c_v the specific heat at a constant volume (being reasonably constant for $T < 1000$ K [17]).

B.1.2 Euler equations

Since we are dealing in this thesis with *aero*acoustics, the medium of interest is air. Usually viscous effects and thermal conduction may be neglected for sound propagation through air. In that case Eqs. (B.1) reduce to the Euler equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = S_c, \tag{B.8a}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij}) = S_{m,i}, \tag{B.8b}$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_j}(\rho H u_j) = S_e. \tag{B.8c}$$

B.2 Linearized Euler equations

To quantify the broad range of noise levels sensed by the human ear, a logarithmic scale is used:

$$SPL = 10\log\Big(\frac{p_{eff}^2}{p_{ref}^2}\Big). \tag{B.9}$$

SPL stands for sound pressure level, p_{eff} is the effective sound pressure (root-mean-square of pressure fluctuation) and p_{ref} the reference pressure. If the reference pressure is $2 \cdot 10^{-5}$ Pa (threshold of hearing), then at a SPL of 120 dB (jet aircraft noise), the effective sound pressure is 20 Pa. This is a fluctuation of only 0.02% of the atmospheric pressure, which is about 10^5 Pa. It is therefore justifiable to linearize the Euler equations, i.e. we consider small perturbations around a steady mean flow:

$$f(\mathbf{x},t) = \bar{f}(\mathbf{x}) + f'(\mathbf{x},t), \qquad \bar{f} = \lim_{T\to\infty} \frac{1}{T} \int_{t_0}^{T+t_0} f(\mathbf{x},t)dt \tag{B.10}$$

$$\begin{aligned} \rho &= \rho_0 + \rho', \\ u_i &= u_{i0} + u_i', \\ p &= p_0 + p'. \end{aligned}$$

As already noted in [4], there are various ways to linearize the Euler equations. Here we start from the conservative form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x_j} = \mathbf{S}, \tag{B.12}$$

with

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u_j \\ \rho u u_j + \delta_{1j} p \\ \rho v u_j + \delta_{2j} p \\ \rho w u_j + \delta_{3j} p \\ \rho H u_j \end{bmatrix} \text{ and } \mathbf{S} = \begin{bmatrix} S_c \\ S_{m,1} \\ S_{m,2} \\ S_{m,3} \\ S_e \end{bmatrix}. \tag{B.13}$$

Perform now the following steps:

- Substitute the decomposition B.11 into the Euler equations. Neglect terms that are have a multiplication of two or more fluctuating quantities.
- Substitute the mean flow variables into the Euler equations. It is assumed that the mean flow satisfies the Euler equations. Temporal derivatives of the mean flow quantities are zero.
- Subtract the equations of step 2 from the obtained equations at step 1. The result are expressions in terms of the fluctuating variables.

For completeness, we will in addition to the decomposition of the hydrodynamic variables also decompose the original sources into mean and fluctuating parts. Below the three steps are shown.

Continuity. Full decomposition:

$$\frac{\partial \rho_0}{\partial t} + \frac{\partial \rho'}{\partial t} + \frac{\partial}{\partial x_j}(\rho_0 u_{0j} + \rho_0 u_j' + \rho' u_{0j} + \rho' u_j') = S_{c,0} + S_c'. \tag{B.14}$$

Mean flow equation:

$$\frac{\partial \rho_0}{\partial t} + \frac{\partial}{\partial x_j}(\rho_0 u_{0j}) = S_{c,0}. \tag{B.15}$$

Momentum. Full decomposition:

$$\frac{\partial \rho_0 u_{i0}}{\partial t} + \frac{\partial \rho_0 u_i'}{\partial t} + \frac{\partial \rho' u_{i0}}{\partial t} + \frac{\partial \rho' u_i'}{\partial t} + \frac{\partial}{\partial x_j}\Big(\rho_0 u_{i0} u_{j0} + \rho_0 u_i' u_{j0} + \rho' u_{i0} u_{j0}$$
$$+ \rho' u_i' u_{j0} + \rho_0 u_{i0} u_j' + \rho_0 u_i' u_j' + \rho' u_{i0} u_j' + \rho' u_i' u_j' + (p_0 + p')\delta_{ij}\Big)$$
$$= S_{m,i0} + S_{m,i}'. \quad \text{(B.16)}$$

Mean flow equations:

$$\frac{\partial \rho_0 u_{i0}}{\partial t} + \frac{\partial}{\partial x_j}(\rho_0 u_{i0} u_{j0} + p_0 \delta_{ij}) = S_{m,i0}. \quad \text{(B.17)}$$

Energy. Full decomposition:

$$\frac{\partial}{\partial t}\Big(\frac{p_0 + p'}{\gamma - 1} + \frac{1}{2}(\rho_0 u_{k0} u_{k0} + \rho_0 u_{k0} u_k' + \rho_0 u_k' u_{k0} + \rho' u_{k0} u_{k0}\Big) + \frac{\partial}{\partial x_j}\Big(\frac{\gamma(p_0 + p')}{\gamma - 1}\Big)$$
$$+ \frac{\partial}{\partial x_j}\Big(\frac{1}{2}(\rho_0 u_{k0} u_{k0} u_{j0} + \rho_0 u_{k0} u_{k0} u_j' + \rho_0 u_{k0} u_k' u_{j0} + \rho_0 u_k' u_{k0} u_{j0} + \rho' u_{k0} u_{k0} u_{j0})\Big)$$
$$= S_{e,0} + S_e'. \quad \text{(B.18)}$$

Mean flow equation:

$$\frac{\partial (\gamma - 1)^{-1} p_0 + \frac{1}{2}\rho_0 u_{k0} u_{k0}}{\partial t} + \frac{\partial}{\partial x_j}(\frac{1}{2}\rho_0 u_{k0} u_{k0} u_{j0}) = S_{e,0}. \quad \text{(B.19)}$$

For the energy equation we have omitted the terms of second order and higher orders on forehand to prevent the expressions from becoming too long. Subtracting now Eq. (B.15) from Eq. (B.14), Eq. (B.17) from Eq. (B.16) and Eq. (B.19) from Eq. (B.18) and simplefying yields:

$$\mathbf{U} = \begin{bmatrix} \rho' \\ \rho_0 u' + u_0 \rho' \\ \rho_0 v' + v_0 \rho' \\ \rho_0 w' + w_0 \rho' \\ (\gamma - 1)^{-1} p' + \frac{1}{2} u_{k0} u_{k0} \rho' + \rho_0 u_{k0} u_k' \end{bmatrix}, \quad \text{(B.20)}$$

$$\mathbf{F} = \begin{bmatrix} \rho_0 u_j' + u_{j0} \rho' \\ \rho_0 u_{j0} u' + \rho_0 u_0 u_j' + u_0 u_{j0} \rho' + \delta_{1j} p' \\ \rho_0 u_{j0} v' + \rho_0 v_0 u_j' + v_0 u_{j0} \rho' + \delta_{2j} p' \\ \rho_0 u_{j0} w' + \rho_0 w_0 u_j' + w_0 u_{j0} \rho' + \delta_{3j} p' \\ \frac{1}{2} u_{k0} u_{k0} (u_{j0} \rho' + \rho_0 u_j') + \rho_0 u_{k0} u_{j0} u_k' + \frac{\gamma}{\gamma - 1}(u_{j0} p' + p_0 u_j') \end{bmatrix} \quad \text{(B.21)}$$

and $\mathbf{S} = [S_c' \; S_{m,i}' \; S_e']^T$.

The above is a possible linearization of the Euler equations. Bogey and Bailly [7] use a different state vector for the LEE. They use e.g. only p' as state variable for the energy equation and place part of the gradients of the mean flow into a separate vector $\mathbf{H}$. Their form can be obtained by cleverly adding and subtracting multiplications of the mean and perturbed momentum and continuity equations such that terms cancel when the chain rule for differentiation is used. The result is:

$$\mathbf{U} = \begin{bmatrix} \rho' \\ \rho_0 u' \\ \rho_0 v' \\ \rho_0 w' \\ p' \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho' u_{j0} + \rho_0 u'_j \\ \rho_0 u_{j0} u' + \delta_{1j} p' \\ \rho_0 u_{j0} v' + \delta_{2j} p' \\ \rho_0 u_{j0} w' + \delta_{3j} p' \\ u_{j0} p' + \gamma p_0 u'_j \end{bmatrix}, \tag{B.22}$$

$$\mathbf{H} = \begin{bmatrix} 0 \\ \frac{\partial u_0}{\partial x_j} \rho_0 u'_j - \delta_{1i} \frac{\rho'}{\rho_0} \frac{\partial p_0}{\partial x_i} \\ \frac{\partial v_0}{\partial x_j} \rho_0 u'_j - \delta_{2i} \frac{\rho'}{\rho_0} \frac{\partial p_0}{\partial x_i} \\ \frac{\partial w_0}{\partial x_j} \rho_0 u'_j - \delta_{3i} \frac{\rho'}{\rho_0} \frac{\partial p_0}{\partial x_i} \\ (\gamma - 1)(p' \frac{\partial u_{j0}}{\partial x_j} - u'_j \frac{\partial p_0}{\partial x_j}) \end{bmatrix}, \tag{B.23}$$

$$\mathbf{S} = \begin{bmatrix} S'_c \\ S'_{m,1} - u_0 S_{m,1} - \frac{\rho'}{\rho_0}(S_{m,10} - u_0 S_{c,0}) \\ S'_{m,2} - v_0 S_{m,2} - \frac{\rho'}{\rho_0}(S_{m,20} - v_0 S_{c,0}) \\ S'_{m,3} - w_0 S_{m,3} - \frac{\rho'}{\rho_0}(S_{m,30} - w_0 S_{c,0}) \\ S'_e - S_{m,k0} u'_k - S'_{m.k} u_{k0} + u_{k0} u'_k S_{c,0} + \frac{1}{2} u_{k0} u_{k0} S'_c \end{bmatrix}. \tag{B.24}$$

The performed operations can be decoded from the resulting source term (e.g. from the momentum equations of the LEE given earlier: 1) u_{0i} times the continuity equation of the LEE was subtracted, 2) $\frac{\rho'}{\rho_0}$ times the continuity equation for the mean flow was subtracted, and 3) $u_{0i} \frac{\rho'}{\rho_0}$ times the momentum equation for the mean flow was added). Like in Eq. (B.23), Lafon [19] also uses for the vector $\mathbf{H}$ the term $-\frac{\rho'}{\rho_0} \frac{\partial p_0}{\partial x_i}$ for the momentum equations. This term may be replaced though for $\rho' u_{i0} \frac{\partial u_0}{\partial x_j}$, since the steady mean flow satisfies the Euler equations. Then Eqs. (B.22)-(B.23) are the same set of equations that Bogey and Bailly [7] use. The source term formulation of [7] is however chosen in analogy to the method of Lilley as *possible* aerodynamic sound source.

Appendix C

Inclusion of mean density variations

In case the the Mach number is high enough to make mean density variations of importance, the following modifications are suggested:

$$\mathbf{q} = \begin{bmatrix} \rho_0 u' \\ \rho_0 v' \\ \rho_0 w' \\ p' \end{bmatrix}, \tag{C.1}$$

$$\mathcal{W}^1 + \mathcal{W}^2 = \begin{bmatrix} (\rho_r u'_r - \rho_l u'_l) - n_x(\rho_r U'_r - \rho_l U'_l) \\ (\rho_r v'_r - \rho_l v'_l) - n_y(\rho_r U'_r - \rho_l U'_l) \\ (\rho_r w'_r - \rho_l w'_l) - n_z(\rho_r U'_r - \rho_l U'_l) \\ 0 \end{bmatrix}, \tag{C.2}$$

$$\mathcal{W}^3 = \begin{bmatrix} \frac{c_l n_x}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) + \frac{n_x}{c_l+c_r}(p'_r - p'_l) \\ \frac{c_l n_y}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) + \frac{n_y}{c_l+c_r}(p'_r - p'_l) \\ \frac{c_l n_z}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) + \frac{n_z}{c_l+c_r}(p'_r - p'_l) \\ \frac{c_l c_r}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) + \frac{c_r}{c_l+c_r}(p'_r - p'_l) \end{bmatrix}, \tag{C.3}$$

$$\mathcal{W}^4 = \begin{bmatrix} \frac{c_r n_x}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) - \frac{n_x}{c_l+c_r}(p'_r - p'_l) \\ \frac{c_r n_y}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) - \frac{n_y}{c_l+c_r}(p'_r - p'_l) \\ \frac{c_r n_z}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) - \frac{n_z}{c_l+c_r}(p'_r - p'_l) \\ -\frac{c_r c_l}{c_l+c_r}(\rho_r U'_r - \rho_l U'_l) + \frac{c_l}{c_l+c_r}(p'_r - p'_l) \end{bmatrix}, \tag{C.4}$$

where $c_l = \sqrt{\gamma p_l/\rho_l}$ and $c_r = \sqrt{\gamma p_r/\rho_r}$.