Spring 2023

The Linearized Euler Equations for Predicting Supersonic Jet Noise of a Rectangular Jet Using OpenFOAM

Patrick Good
Embry-Riddle Aeronautical University, goodp1@my.erau.edu

The Linearized Euler Equations for Predicting Supersonic Jet Noise of a Rectangular Jet Using OpenFOAM

By

Patrick Good

A Thesis Submitted to the Faculty of Embry-Riddle Aeronautical University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Aerospace Engineering

April 2023

Embry-Riddle Aeronautical University

Daytona Beach, Florida

The Linearized Euler Equations for Predicting Supersonic

Jet Noise of a Rectangular Jet Using OpenFOAM

By

Patrick Good

This Thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. Reda Mankbadi, Department of Aerospace Engineering, and has been approved by the members of the Thesis Committee. It was submitted to the Office of the Senior Vice President for Academic Affairs and Provost, and was accepted in the partial fulfillment of the requirements for the Degree of Master of Science in Aerospace Engineering.

THESIS COMMITTEE

Chair, Dr. Reda Mankbadi	Member, Dr. Vladimir Golubev
Member, Dr. Tasos Lyrintzis	Member, Dr. Seyyed Salehian
Graduate Program Coordinator, Dr. Hever Moncayo	Date
Dean of the College of Engineering, Dr. James W. Gregory	Date
Associate Provost of Academic Support, Dr. Christopher Grant	Date

To Luna, my dog, who was always under my desk until the end.

You will forever be missed.

ACKNOWLEDGEMENTS

First off I would like to thank my Advisor Dr. Mankbadi for giving me the opertunity to pursue this thesis. Before this thesis was even completed my whole career in Aeroacautics has begun due to this work. The fact that this work means something to the academic community and industry is an understatement. I could of never imagined at the begning what this would lead too. I would also like to thank Dr. Salehian for his time as well. He was a guiding force that kept me on track and provided crucial inputs in time of crises. His assistance was a vaulable asset for this work.

Seccondly, I would like to thank the entire Embry-Riddle Aeropace Engineering faculty and staff over the years of my time in their programs. There professionalism and eagerness to help is beyond top knocth and this degree would be impossible to complete with out them.

This work was difficult to complete. The equations solved and the methods available to solve them led to many long nights and frustration. This tested my abilities to think on a high level and problem solve in ways I have never personally expericend before. Being so, this allowed for a lot of personal growth howwever, with growth comes growing pains. With that I would like to take this last moment to thank my loveing partner Emily, family, and friends for there support. None of this would be possible with out you.

ABSTRACT

This thesis presents a Linearized Euler Equation (LEE) solver that was developed for OpenFOAM. OpenFOAM is an open-source Computational Fluid Dynamics (CFD) package that is widely used by industry and academia. The LEE's are a set of equations used in Computational Aeroacoustics (CAA). They can solve the acoustic solution directly and quickly. The solver developed, named leeFoam, was tested against analytical solutions to verify accuracy and then was utilized to predict the acoustics of a heated supersonic rectangular jet. The rectangular jet (RJET) results were compared against experimental data.

The results from this work show that OpenFOAM can be used as an effective platform to develop CAA specific solvers. The leeFoam solver successfully predicted acoustic solutions accurately and quickly. To aid in the production of accurate results several other tools were developed to go along with the leeFoam solver to enhance its capabilities. This included boundary dampening to prevent reflections and artificial viscosity to aid in stability. In all, this allowed the leeFoam solver to produce accurate CAA results allowing this code to be used by any professional looking to run the Linearized Euler Equations in their workflow.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

From the beginning of the turbojet era in the 1940's to the modern-day engineering marvel of today's jet engines, jet noise research still remains a high priority in the aerospace sector. Jet noise first became a problem for communities surrounding airports as they were suddenly experiencing much louder and lower frequency noise compared to previous propeller driven aircraft [1]. Today, with increasing population density more people have migrated around airport vicinities and regulators have been tasked with pushing the industry towards a quitter future.

The effects of jet noise and other forms of noise pollution have been well studied in the past 50 years. Researchers such as Dutchen have shown that:

> "Noise pollution not only drives hearing loss, tinnitus, and hypersensitivity to sound, but can cause or exacerbate cardiovascular disease; type 2 diabetes; sleep disturbances; stress; mental health and cognition problems, including memory impairment and attention deficits; childhood learning delays; and low birth weight" [2].

These are good reasons for regulators to implement action and for the aerospace industry to work on solutions.

The work presented in this thesis is meant to be a tool to assist the academic community and industry in formulating these aforementioned solutions. This tool is a computational code to solve noise generation and propagation utilizing the Linearized Euler Equations (LEE). Before we get into the details however, we must first understand the basics of jet noise and how it is produced.

1.1 Jet Noise Sources

Precise interpretation of the exact source or main contributing factor leading to jet noise still remains the holy grail for any jet noise researcher. The fact is, turbulent flows in and around jet

exhaust plumes are very complex. The current big picture understanding however, is that the noise source is within that jet plume turbulence [3].

Early versions of jet aircraft and modern-day military aircraft utilize low-bypass turbojet engines. This type of jet engine produces high jet exit velocities and high amounts of noise. The high jet velocity core mixes with the relatively low velocity surrounding atmosphere. This mixing region is where the noise is produced as complex turbulent structures are formed. Researchers have discovered most of the turbulence that contributes to the noise is concentrated within 2 potential core lengths of the jet and within this region exists a separation of scales [4]. See figure 1.1. Small scale structures are those smaller than the jet diameter while large scale structures are larger than the jet diamcheck1eter. The large-scale structures form the dominate noise source as they decay into smaller scale components through merging and cascading [4]. This breakdown of turbulent structures form pressure disturbances that propagate through the surrounding atmosphere as waves.

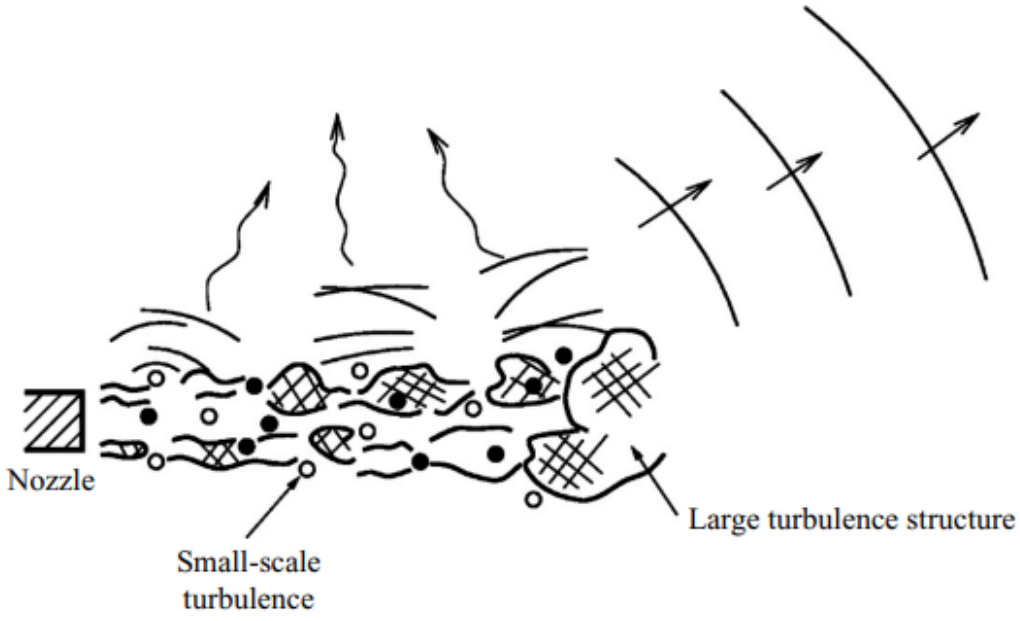


Figure 1.1 Jet Noise Structures [4]

Noise produced from a jet can be visualized using Schillerian imagery. Figure 1.2 shows a snapshot of a jet exhaust taken with Schillerian imagery. Note the complex turbulent structures in the exhaust and the propagating waves traveling outward away from the jet.

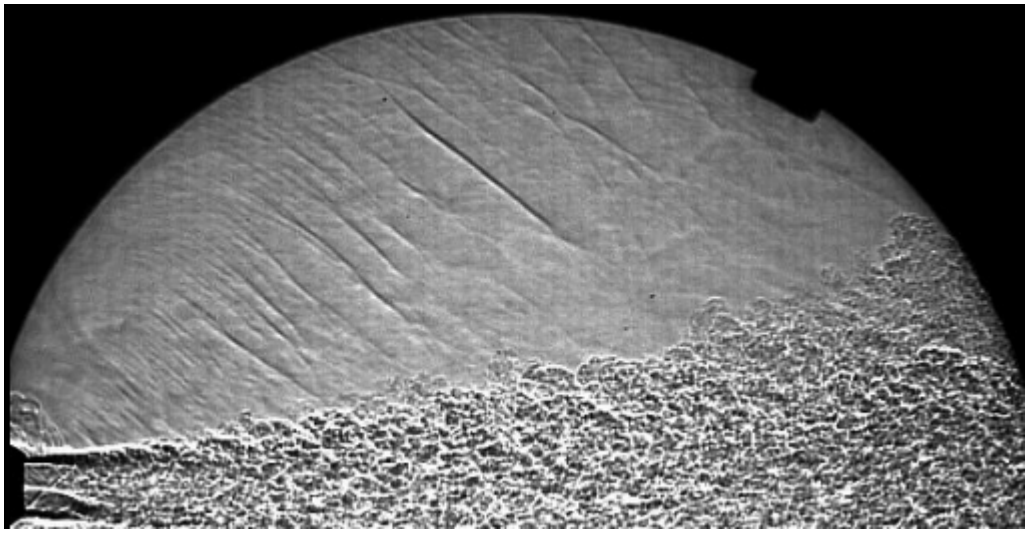


Figure 1.2 Schillerian Imagery of a Supersonic Jet [5]

1.2 Jet Noise Reduction

Over the years the industry has made advancements in reducing jet noise. Current advancements actively deployed on aircraft flying today are increased bypass ratios, chevrons, and acoustic treatment material. We will not get into the details of how these work as the scope of this thesis is not on noise reduction methods. However, the premises of these first two methods is to treat the problem of the jet core interacting with the surrounding atmosphere. The high-bypass ratio creates a cooler and slower moving jet which surrounds the hotter and faster moving core[6]. This provides a buffer zone for the noise radiating from the core and also helps with performance. Chevrons accelerate the mixing with the jet and the surrounding atmosphere producing a smaller noise source area thus, helping with noise but, with the cost of a performance penalty [7]. Figure 1.3 shows a GE 9X engine on the Boeing 787 with high-bypass ratio and chevrons.

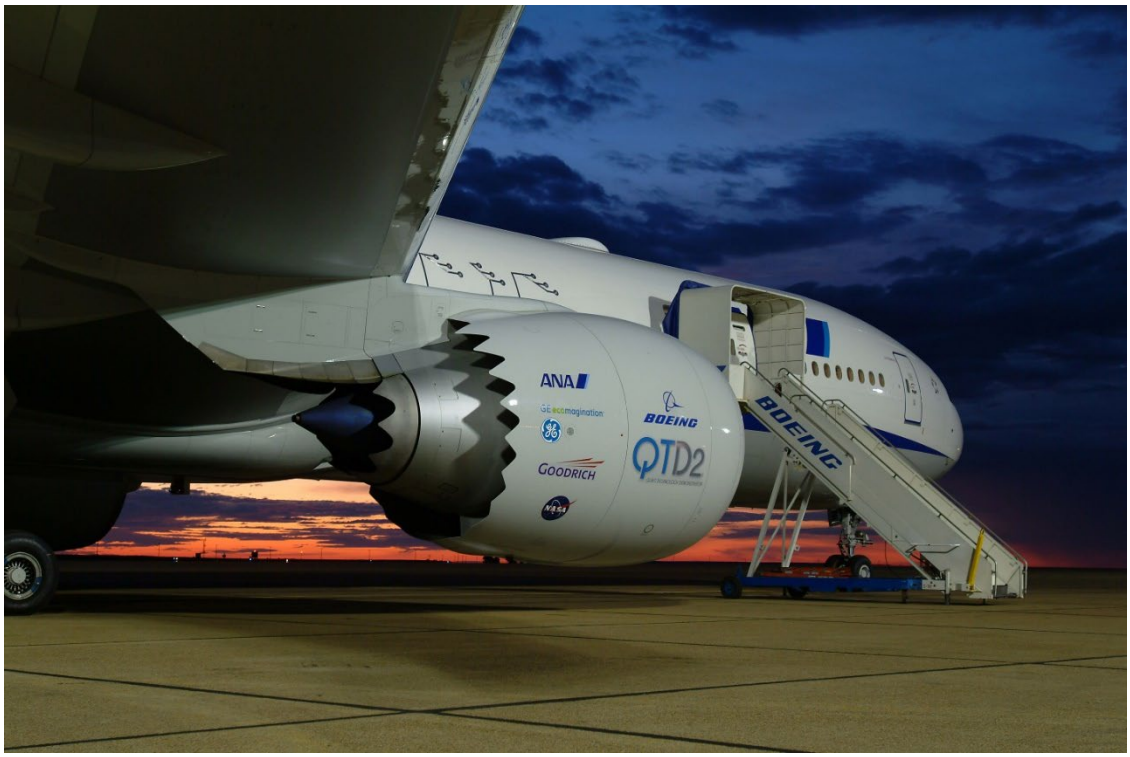


Figure 1.3 GE 9X on Boeing 787 (NASA)

Methods for designing and testing noise reduction techniques utilize experimental testing and numerical modeling. Experimental testing offers the most accurate way to determine whether a technique or device works. However, experimental testing especially at a full scale is expensive. This is a monetary and time expense. Numerical modeling offers a total lower cost advantage over experimental testing as multiple simulations can be conducted simultaneously. Resources to conduct numerical simulations are also relatively inexpensive. Since this thesis focuses on the numerical side, we will discuss the methods used to conduct computational aeroacoustics (CAA) simulations.

1.3 Computational Aeroacoustics (CAA)

There are three numerical approaches commonly mentioned in the literature. The first one is direct numerical simulation (DNS). DNS solves the compressible Navier-Stokes equations directly throughout the computational domain to predict noise generation and propagation [8]. DNS,

however, is highly computationally demanding, making it impractical even with today's advances in computational power. Another approach that has become more practical with today's resources is the use of Large-Scale Eddy Simulations (LES). In LES, large scale turbulent components are calculated, and the small scales are not captured [9]. LES has been used to compute noise sources in jet cases and has been proven in the literature to perform well. Figure 1.5 shows a numerical Scherlin image from a LES simulation. Compare this to figure 1.2. LES does however require fine numerical meshes along with low time stepping to remain accurate and stable, so it is usually only commonly used to compute the noise sources and near-field regions. To compute into the far-field with LES is impractical due to the high computational time required for the lather as adding the far-field drastically increases the mesh size. Also, the physics associated with sound propagation is much simpler than generation. So, performing LES in the far-field gives no benefit compared to taking a simpler approach. Typically, the far-field for LES simulations is calculated by a Lighthill's analogy [10,11]. Lighthill's papers really began the advancements in jet noise [12].

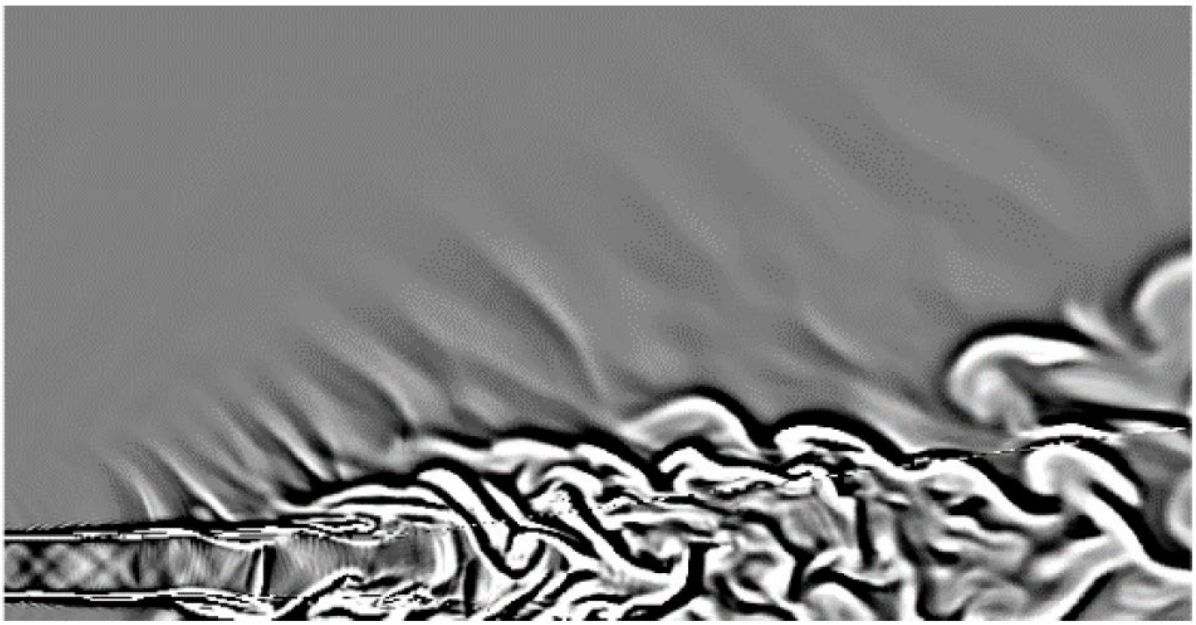


Figure 1.4 LES of Supersonic Jet [19]

An even faster approach compared to LES that also does not need Lighthill's analogy is the use of the Linearized Euler Equations (LEE). LEE has the benefit of solving for the acoustic solution directly. All other physics that are not related to the acoustic propagation are ignored thus saving computational time.

1.4 Linearized Euler Equations (LEE)

As mentioned, the LEE equations greatly improve time to solution for aeroacoustics. This is due to the nature of the equations. The LEE equations can be derived from the Navier-Stokes equations by first removing all viscous and non-linear effects. The large-scale structures in free shear flows are inviscid [8]. The large-scale turbulent structures however do have some non-linear properties but, these effects seem to be more important to subsonic jets [8]. Since we are interested in the sound generation and propagation by the breakdown of large-scale structures in supersonic jets this is a valid assumption.

The LEE's can be derived by first removing the viscous terms from the Navier-Stokes equations. This will lead us down to the Euler equations for compressible flow. The next step is to remove all nonlinear terms by linearizing the equations around a mean flow. This leads to a perturbation and mean flow component. Since sound can be described by a pressure perturbation in a medium, solving for these perturbations directly leads to the acoustic solution. For the medium, a mean flow must be given. By definition, this flow is stagnant in time as it is a time averaged flow. So, a time average solution or a RANS solution can be used as the mean flow. More details on this will come in a later section but, Equations 1-3 describe the governing equations, continuity, momentum, and energy respectively. A full derivation is shown in appendix A.

$$\frac{\partial \rho'}{\partial t} + \nabla \cdot (\rho' \boldsymbol{U} + \rho \boldsymbol{u}') = S_c \tag{1.1}$$

$$\frac{\partial \boldsymbol{u}'}{\partial t} + \left(\left[\boldsymbol{u}' + \frac{\rho'}{\rho} \boldsymbol{U} \right] \cdot \nabla \right) \boldsymbol{U} + (\boldsymbol{U} \cdot \nabla) \boldsymbol{u}' + \frac{1}{\rho} \nabla p' = S_m \tag{1.2}$$

$$\frac{\partial p'}{\partial t} + \boldsymbol{U}' \cdot \nabla P + p' \gamma (\nabla \cdot \boldsymbol{U}) + \boldsymbol{U} \cdot \nabla p' + \gamma P (\nabla \cdot \boldsymbol{U}') = S_e \tag{1.3}$$

There have been many LEE solvers developed by many researchers and even some commercial codes exist. However, these codes are not available for a mass audience. Also, in the Aeroacoustics field research into the supersonic rectangular jet has been a hot topic in recent years and LEE has not been utilized. So, this Thesis has two main purposes and hypothesis that corelate together. First, can LEE be implemented into OpenFOAM? There are some challenges to implementing such as code that will be discussed however, the benefits of doing so are massive. Second, can an OpenFOAM LEE code predict supersonic rectangular jet noise? LEE has been used in the past to predict jet noise as shown by the Mankbadi, 1998 reference [8]. If the OpenFOAM LEE solver is accurate, then it is hypothesized to be possible.

2 The leeFoam Solver

As mentioned, the LEE solver for this Thesis was developed in OpenFOAM. OpenFOAM (Open-source Field Operation and Manipulation) is an open-source Finite-Volume (FVM) CFD package written in C++ originally released in 2004. Due to the opensource nature of OpenFOAM it is relatively easy to write custom solvers that can utilize pre-existing code inside the package. For example, the leeFoam solver written for this thesis can run on thousands of CPUs utilizing OpenFOAM's built in parallelization tools and while benefiting from built in numerical methods inside OpenFOAM.

2.2 Numerical Schemes Available

OpenFOAM has many numerical schemes available for a user's arsenal. The following sections provide a brief description of the schemes tested for the leeFoam solver and comments on their performance. In a later section different schemes will be tested against each other and compared for accuracy on a verification case.

2.2.1 Temporal Discretization Schemes

The leeFoam solver, like any LEE code, is naturally transient in time thus OpenFOAM's transient time schemes are used. The code has the ability to run fully explicitly however, it has been found to run with greater stability semi-implicitly. The continuity and momentum equations are implicit while p' is calculated explicitly in this setup. The schemes available are Euler, Crank-Nicolson, and Backward. The Euler scheme is the most stable but is 1^{st} order accurate. The Backward scheme offers full 2^{nd} order accuracy, but it can become too oscillatory causing it to ruin any accuracy gain. Crank-Nicolson offers a blend between pure 2^{nd} and 1^{st} order that is set with a blending factor. The schemes are shown below, and the performance of these schemes is shown in a later section.

The Euler temporal discretization scheme is shown in equation 2.1. The scheme calculates the derivative of the current time φ from the previous time φ^0 [13].

$$\frac{\partial \varphi}{\partial t} = \frac{\varphi - \varphi^0}{\Delta t} \tag{2.1}$$

The backward scheme shown in equation 2.2 is true 2nd order accurate and it uses three-time levels $\varphi, \varphi^0,$ and φ^{00} to calculate the current time [13].

$$\frac{\partial \varphi}{\partial t} = \frac{3\varphi - 4\varphi^0 + \varphi^{00}}{2\Delta t} \tag{2.2}$$

The Crank-Nicolson scheme [14] uses the midpoint between the current and old-time levels to make the Euler scheme 2nd order accurate. OpenFOAM offers a blending factor that corresponds to the Euler scheme if set to 0 and pure Crank-Nicolson if set to 1 as shown in equation 2.3 [13].

$$\frac{\partial \varphi}{\partial t} = (1 + \lambda)\frac{\varphi - \varphi^0}{\Delta t} + \lambda[A|b]^0\varphi^0 \tag{2.3}$$

Another important concept in the temporal discretization is the Courant Number (Co#). The Co# is dimensionless and describes how long a particular particle spends in a cell. A value of 1 means the particle spends each new time step in the next cell. Anything above 1 means the particle is skipping cells. This is important when determining accuracy and stability as skipping cells can lead to discontinuities. For the leeFoam solver the Acoustic Courant Number is used (ACo#) as shown in equation 2.4. The speed of sound is c and Δl_{min} is the minimum cell length. Typically, the user will reference the max ACo# thus the Δl_{min} is the smallest cell length in the domain.

$$ACo = \frac{c\Delta t}{\Delta l_{min}} \tag{2.4}$$

2.2.2 Gradient Discretization Schemes

OpenFOAM offers two main gradient schemes Gauss Linear and Least-Squares. Both are 2nd order accurate, however, the Least-Squares method can become oscillatory with the leeFoam solver thus causing stability issues. The Gauss Linear scheme uses standard Gauss integration from the FVM as shown in equation 2.5 and the face values φ_f are interpolated from the cell centers linearly [15].

$$\nabla\varphi = \lim_{\Delta V \to 0} \frac{1}{\Delta V} \int_S \varphi dS = \frac{1}{V} \sum_f S_f \varphi_f \tag{2.5}$$

For the Least-Squares method [15] the gradient is calculated using a weighted difference between cell centers and neighboring cells. This helps reduce the error between cells. This is shown in equation 2.6 then the gradient is calculated in equation 2.7.

$$G = \sum_f w_{id}^2 \Delta d \Delta d \tag{2.6}$$

$$\nabla\varphi = \frac{1}{V} \sum_f w_{id}^2 G^{-1} \cdot \Delta d(\varphi_N - \varphi_P) \tag{2.7}$$

OpenFOAM does have gradient limiters to help prevent oscillations however some initial testing was performed with the leeFoam solver, and it was found limiting the gradient added allot of diffusion error.

2.2.3 Divergence Discretization Schemes

OpenFOAM has many divergence schemes available to choose from. The most popular schemes are Gauss Upwind (1st order accurate), Gauss Linear (2nd order accurate), and the TVD scheme Gauss Limited-Linear which uses a limiter that varies from 1st to 2nd order accurate depending on the conditions. The TVD scheme Gauss Limited-Linear with a limiter set to 1.0 was found to work

well with the leeFoam solver. The Gauss Upwind scheme does provide the most stability to the solver, however, the 1st order accuracy creates noticeable diffusion in the solution. The pure Gauss Linear scheme works well for simple cases, however cases with strong mean flow gradients such as with a jet tend to become too oscillatory. TVD schemes help solve this issue as they apply a limiter that limits towards upwind in strong gradients.

The TVD scheme can be visualized with a Sweby diagram shown in figure 2.1. In order for schemes to be classified as a TVD it must remain in the shaded area [13] Schemes that lie on or towards to upwind side of the diagram tend to be the most stable but also the most diffusive. The Sweby diagram also offers another parameter when comparing TVD Schemes. If the scheme lies on the β = r line before reaching linear then the scheme is symmetric, and the gradient limiter is applied evenly. If not, then uneven limiting can lead to distortion [13].

.

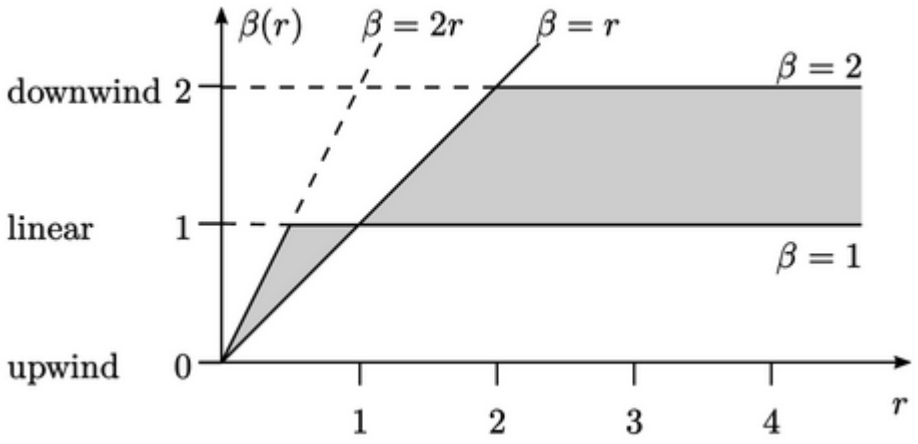


Figure 2.1 Sweby Diagram

Figure 1.2 shows the Sweby diagram of the Limited-Linear scheme as well as another popular scheme in OpenFOAM the Minmod scheme. Notice how the Minmod scheme is symmetric but

also limits more to the upwind. Some initial tests were performed with this scheme on the rectangular jet and found it did well. However, it was noticeably more diffusive than Limited-Linear. Limited-Linear is not symmetric as can be seen but no distortion was noticed in any test and thus was chosen for its accuracy and robustness. The performance of these schemes is shown in a later section.

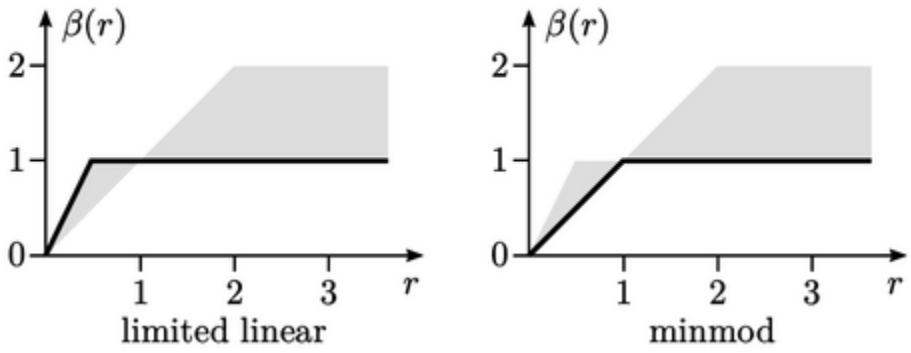


Figure 2.2 Sweby Diagram with Limited-Linear and Minmod

2.3 Additional Numerical Tools Available

The LEE equations are prone to stability issues as they are linear in nature with no viscous dissipation present thus spurious instabilities can grow uncontrollably. These Instabilities can form in the domain or boundaries [8]. To counteract boundary treatment issues and stability problems in the domain two additional tools were developed for the leeFoam solver. An Acoustic Relaxation Term (ART) can be added to the momentum equation and acts as artificial viscous dissipation for the entire domain. For the boundaries, a Boundary Dampening term can also be activated to prevent issues such as reflections and stability problems. These tools will be discussed in detail in this section.

2.3.1 Boundary Dampening

The built in outflow boundary code available from OpenFOAM that works best with the leeFoam solver is the *acusticwaveTransmissive* boundary condition. The *acusticwaveTransmissive*

boundary condition works by solving for the flux term across the boundary with a given advective velocity **U**. See equation 2.8 [16] This advective velocity for the acoustic boundary treatment is the speed of sound at the boundary.

$$\frac{D\emptyset}{Dt} = \frac{\partial \emptyset}{\partial t} + \boldsymbol{U} \cdot \nabla \emptyset = 0 \tag{2.8}$$

$$\frac{D\emptyset}{Dt} \approx \frac{\partial \emptyset}{\partial t} + \boldsymbol{U}_n \cdot \frac{\partial \emptyset}{\partial \boldsymbol{n}} = 0 \tag{2.9}$$

The *acusticwaveTransmissive* boundary in OpenFOAM is not a perfect boundary treatment and not all boundary codes can produce zero reflections however, the OpenFOAM implementation does have a setback. The advection velocity is assumed to be perpendicular to the boundary face. See equation 2.9 [16] This causes an issue when the incoming wave is not perpendicular to the boundary face. When the code was first tested issues persisted where the non-normal component of the incoming wave was reflected into the solution causing an entire failure as a result. This is an issue that can be resolved with future development in OpenFOAM as other commercial codes do employ more advanced boundary conditions.

With the built in OpenFoam outflow boundary condition not preforming in an optimal way, one solution is to kill any waves before they reach the boundary in order to prevent reflections. This can be done with grid stretching or with applying artificial dampening in set zones before the boundaries. Grid stretching is tricky with the leeFoam solver and will be discussed later. Boundary damping is the preferred method.

Artificial dampening works by applying a sink to the governing equations. The sink can be controlled by a coefficient that includes strength and location applied to the computational domain.

Equation 2.10 describes how the dampening is applied to the momentum equation. v is the dampening coefficient and is described in equation 2.11.

$$S_m = -\nu \boldsymbol{u}' \tag{2.10}$$

$$\nu = wf(blendFactor) \tag{2.11}$$

$$blendFactor = \left(\frac{1}{r_2^2 - r_1^2}\right)(R - r_1^2) \tag{2.12}$$

Equation 2.12 shows the blending factor. The blending factor is what controls the strength of the dampening coefficient as a ratio between 0 and 1 with 1 being the strongest. The ratio is controlled by varying the radial distance between the two R values. This creates a "sponge zone" that is placed before the boundary to absorb any incoming waves. An illustration of this is shown in figure 1.3. This "sponge zone" absorbs incoming waves thus preventing any boundary condition issues.

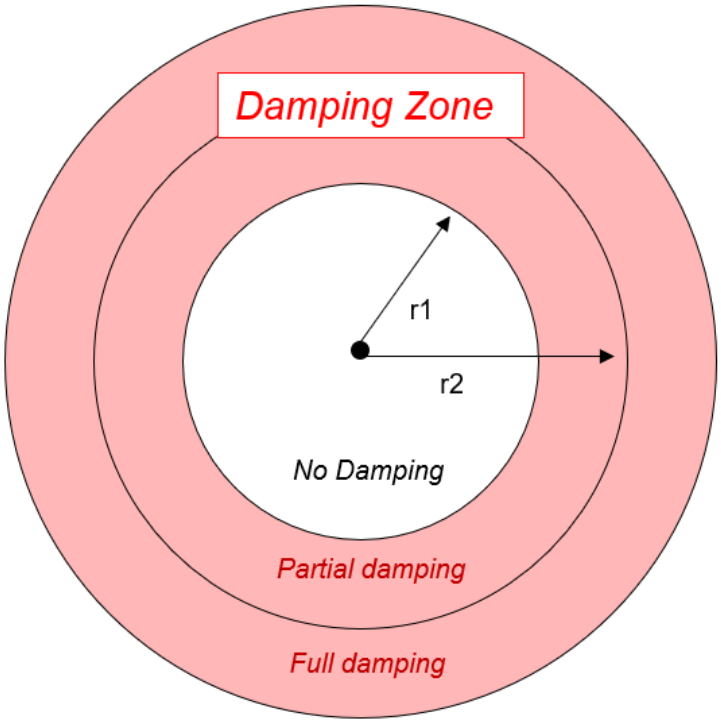


Figure 2.3 Boundary Dampening Sponge Zone

2.3.2 Acoustic Relaxation Term

To counteract instabilities not caused by the boundaries but inside the domain an acoustic relaxation term was added in the governing momentum like in equation 2.13 where $\boldsymbol{\sigma}$ is the ART relaxation factor [17]. This can be turned on or off and some cases do not require it. For the rectangular jet case it was used and a relaxation factor $\boldsymbol{\sigma}$ was chosen that provided dampening to unwanted high frequency noise while having minimal effect to the frequency being tested. Further details of ART being implemented are in the jet section.

$$S_m = \nabla[\sigma(\nabla \cdot \boldsymbol{u}')] \tag{2.13}$$

2.4 Code Verification Test

Like most new CFD codes the leeFoam solver was benchmarked and compared for accuracy against known solutions. Presented in this thesis and in the next few sections the leeFoam solver is compared to analytical solutions that are available. The solver is first tested against a 1-D wave with zero mean flow then against a Gaussian pulse. The Gaussian Pulse is a common benchmark for CAA codes. In this case, it is tested with a uniform mean flow. Unfortunately, an analytical benchmark for a non-uniform mean flow was not available. However, for the jet case we are comparing the code to experimental data so that will suffice.

2.4.1 1-D Wave

The 1-D wave test conducted for this work was a simple sine wave convecting through a stagnant men flow. The goal of the test is to verify a disturbance properly convects through the domain which, in this case, is a sinusoidal oscillation at the inlet. The analytical solution follows the form in equation 2.14. Figure 2.4 shows the results.

$$f(t) = \sin\left(\frac{2\pi t}{\lambda}\right) \tag{2.14}$$

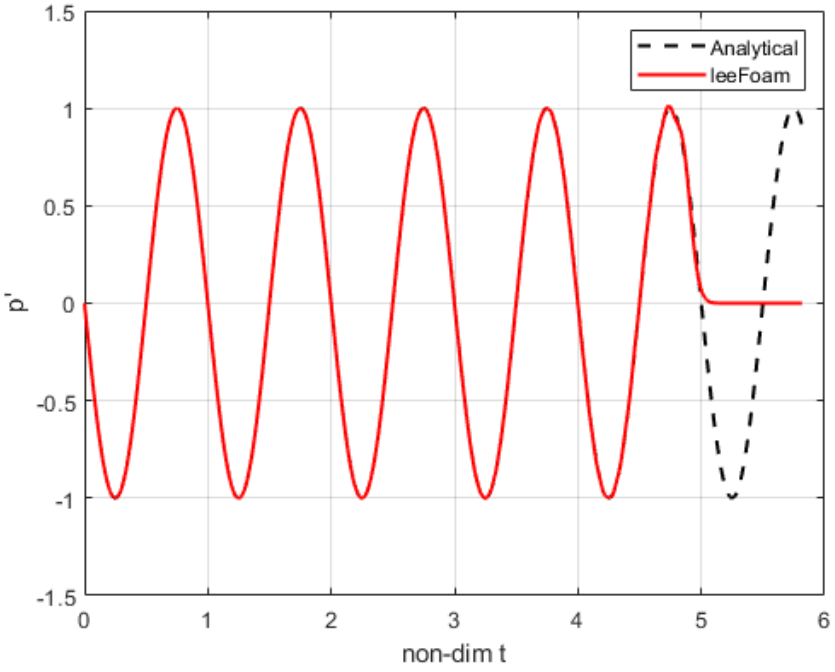


Figure 2.4 1-D Wave test

As can be seen the leeFoam solver convects the disturbance without any dissipation or dispersion errors. The numerical schemes for this test were of pure 2nd order accurate in both time and space.

2.4.2 Gaussian Pulse in Uniform Mean Flow

The next verification is the 2-D Gaussian Pulse in a uniform mean flow. It is important to note OpenFOAM naturally runs in 3-D. So, the Gaussian Pulse is technically calculated in 3-D. However, the mesh is extruded by 1 block and boundary conditions are set thus making the case behaves like a pure 2-D case. This is a normal procedure noted in the OpenFOAM documentation.

The initial conditions and solution are from the work of Tam and Webb, 1992 [18] and is shown in equations 2.15 and 2.16. The parameters of those equations are shown in table 2.1.

$$p'(x,t0) = \epsilon e^{-\alpha(x^2+y^2\,z^2)} \tag{2.15}$$

$$p'(x,t) = \frac{\varepsilon}{\beta}\int_0^{\infty} \delta^2 e^{-\frac{\delta^2}{4\alpha}} \cos(t\delta)\, j_0(\delta n) d\delta \tag{2.16}$$

Table 2.1 Gaussian Pulse Parameters

Parameter	**Symbol**	**Value**
Amplitude	ε	1
Pulse Half-Width	b	9
-	α	$\ln(2)/b^2$
-	n	$\sqrt{(x-Mt)^2 + y^2}$
-	β	$2\alpha\sqrt{\pi\alpha}$
-	j_0	$\sin(\delta n)/\delta n$
Inflow Mach Number	M	0.2

The results of this test are presented in figures 2.5 and 2.6. The numerical schemes used for this test are of pure 2nd order accurate. As can be seen the leeFoam solver convects the pulse without any dissipation or dispersion errors. Next, we will look into how OpenFOAM's various numerical schemes affect the Gaussian Pulse solution.

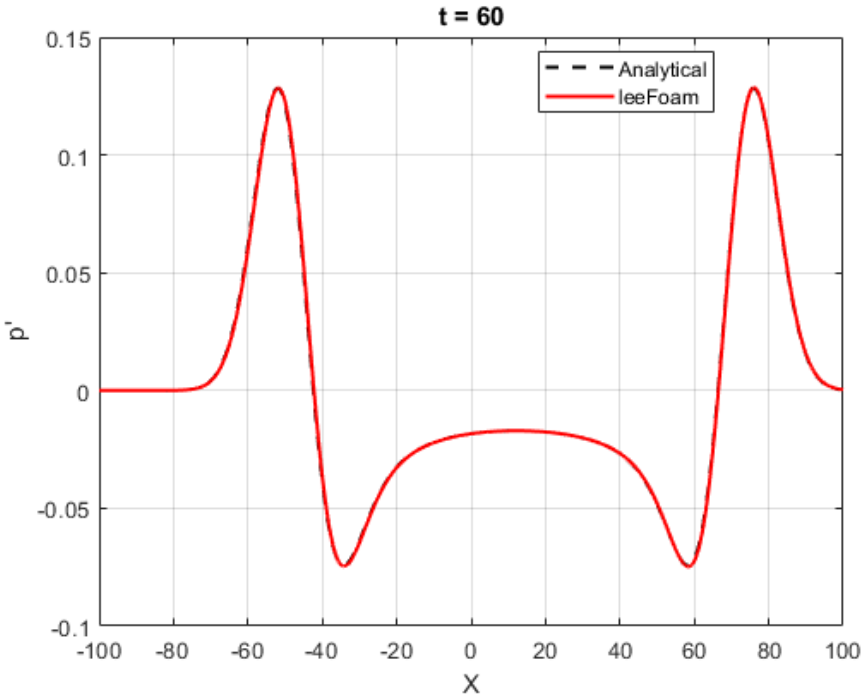
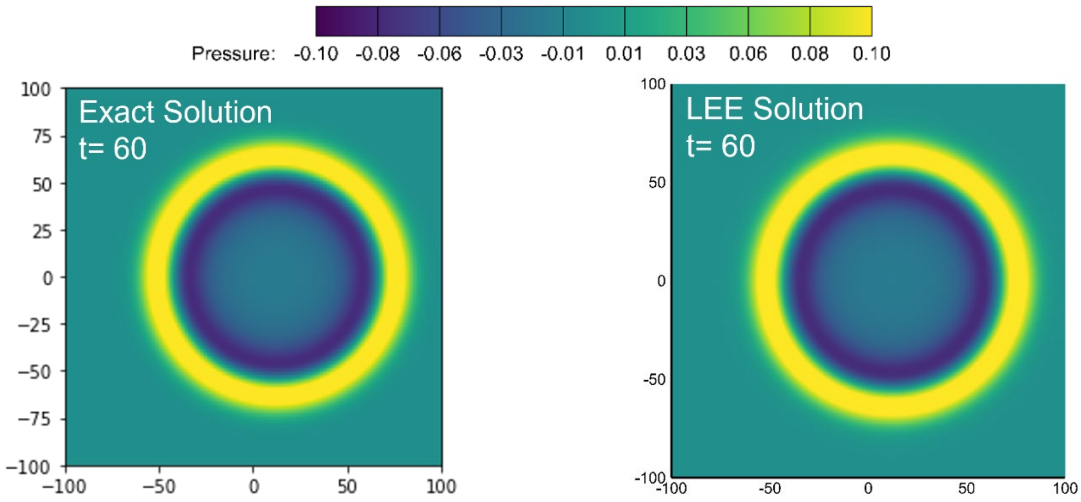


Figure 2.5 Gaussian Pulse



(a) (b)

Figure 2.6 Gaussian Pulse contours

2.5 Numerical Schemes Test on Gaussian Pulse

This section will demonstrate the leeFoam solvers performance while using various numerical schemes on the Gaussian Pulse case. A grid independence study was also conducted. The result of this study provides insight on how the leeFoam solver performs and what schemes should be used for cases such as with the rectangular jet.

2.5.1 Grid Independence

A common practice for any CFD case is to eliminate errors due to the numerical grid used. To do this it is common to refine either the entire domain or certain sections and check for differences in the solution. For example, if refining the grid causes the solution to change sufficiently then the solution is not grid independent and the accuracy should be questioned. The grid should then be further refined until there is no significant change in solution. This was performed for the Gaussian Pulse as shown in figure 2.7. The grid was refined on a domain of length L = 400 by varying the number of cells from N = 100 to N = 800. Figure 2.8 shows the change in relative error between the different grid sizes. Ultimately the grid of N = 800 was used in all other shown Gaussian Pulse cases.

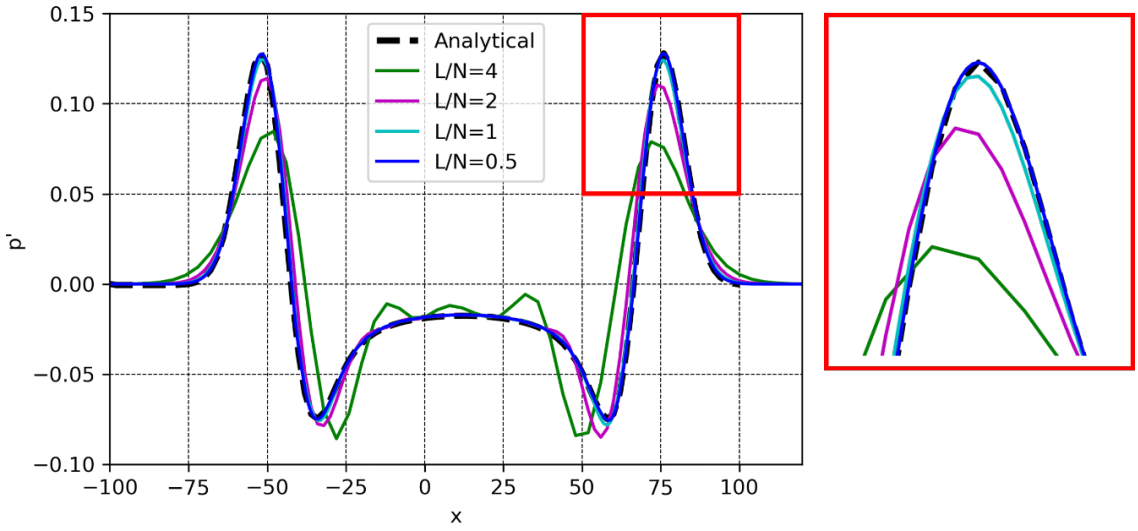


Figure 2.7 Grid Independence Study

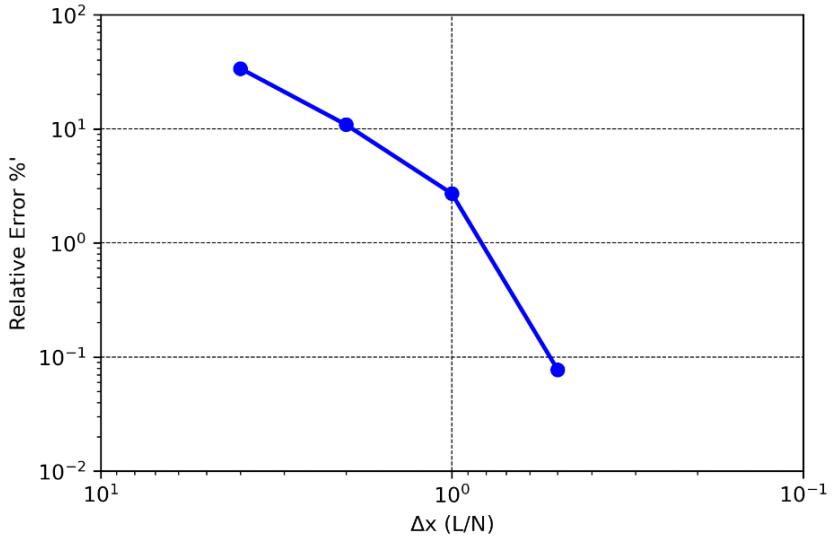


Figure 2.8 Grid Independence Study Relative Error

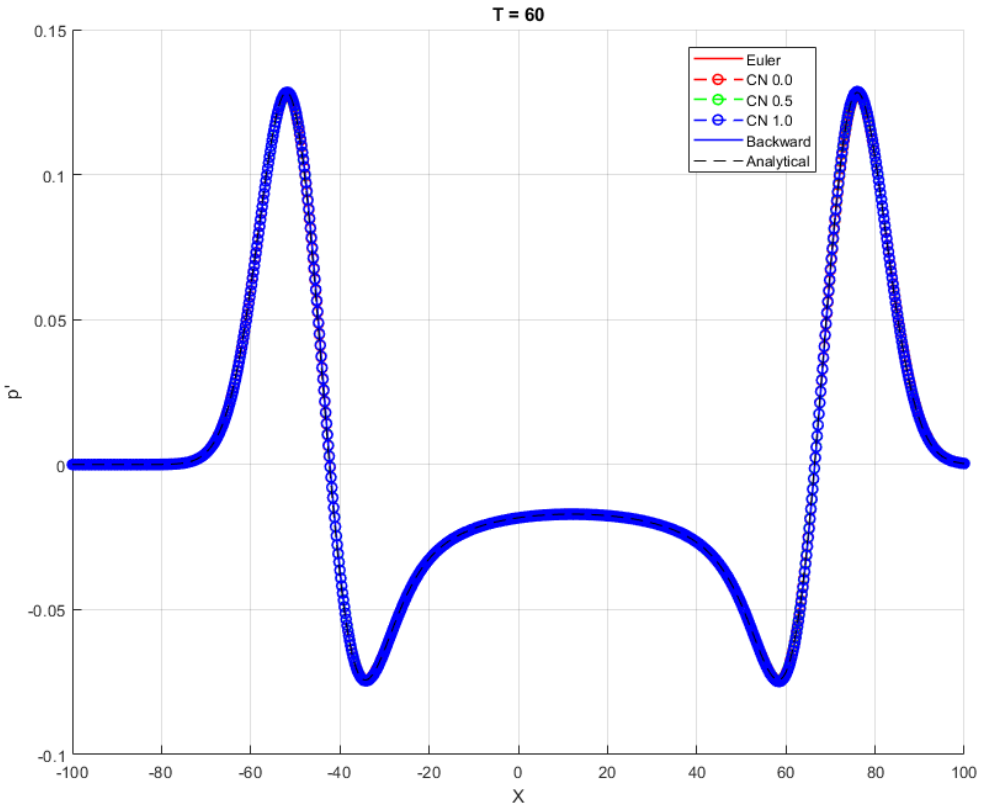
2.5.2 Temporal Discretization Schemes Test

This section tests the performance of the temporal discretization schemes available to the leeFoam solver shown in figure 2.9. The relative errors are shown in table 2.2. As can be seen there is minimal effect on the time schemes chosen when run at the same Acoustic Co#. It is interesting however as the Euler and Crank-Nicolson 0.0 schemes can run at CO#'s much higher while still

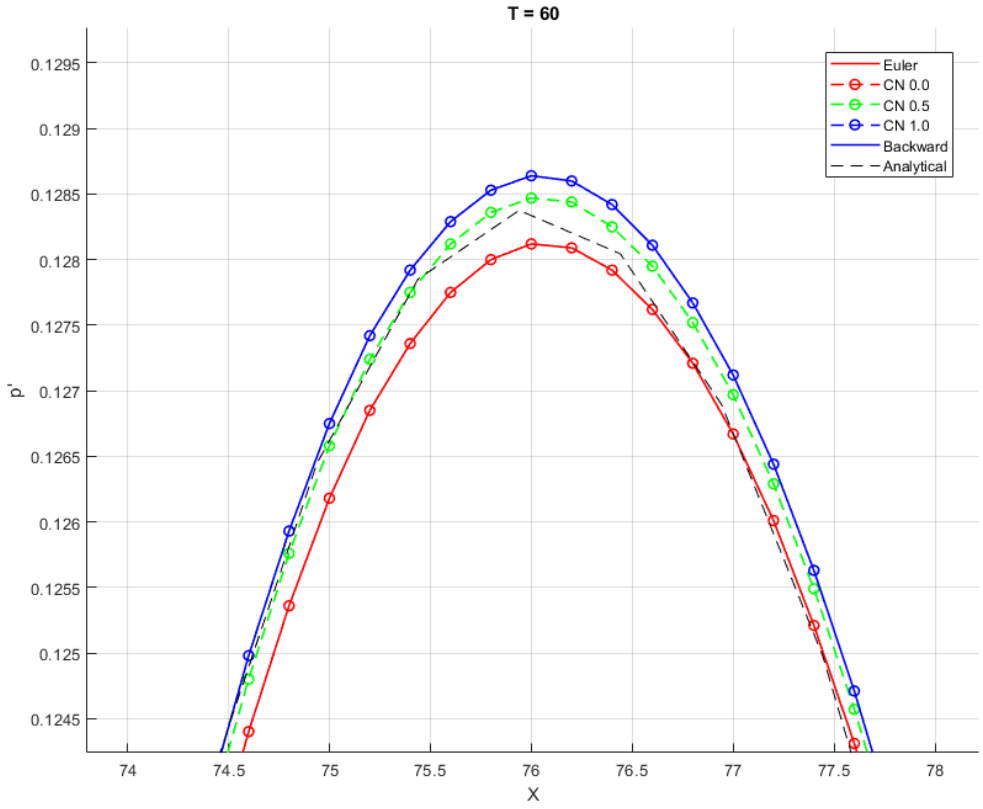
maintaining accuracy. The other schemes failed to produce a solution as they became unstable at ACo#'s above 0.027.

Table 2.2 Temporal Schemes Error

Scheme	**Acoustic Co#**	**% Relative Error**
Euler / Crank-Nicolson 0.0	1.092	0.0739
Euler / Crank-Nicolson 0.0	0.027	0.0717
Crank-Nicolson 0.0	0.027	0.0717
Crank-Nicolson 0.5	0.027	0.0745
Crank-Nicolson 1.0	0.027	0.0750
Backward	0.027	0.0750



(a)



(b)

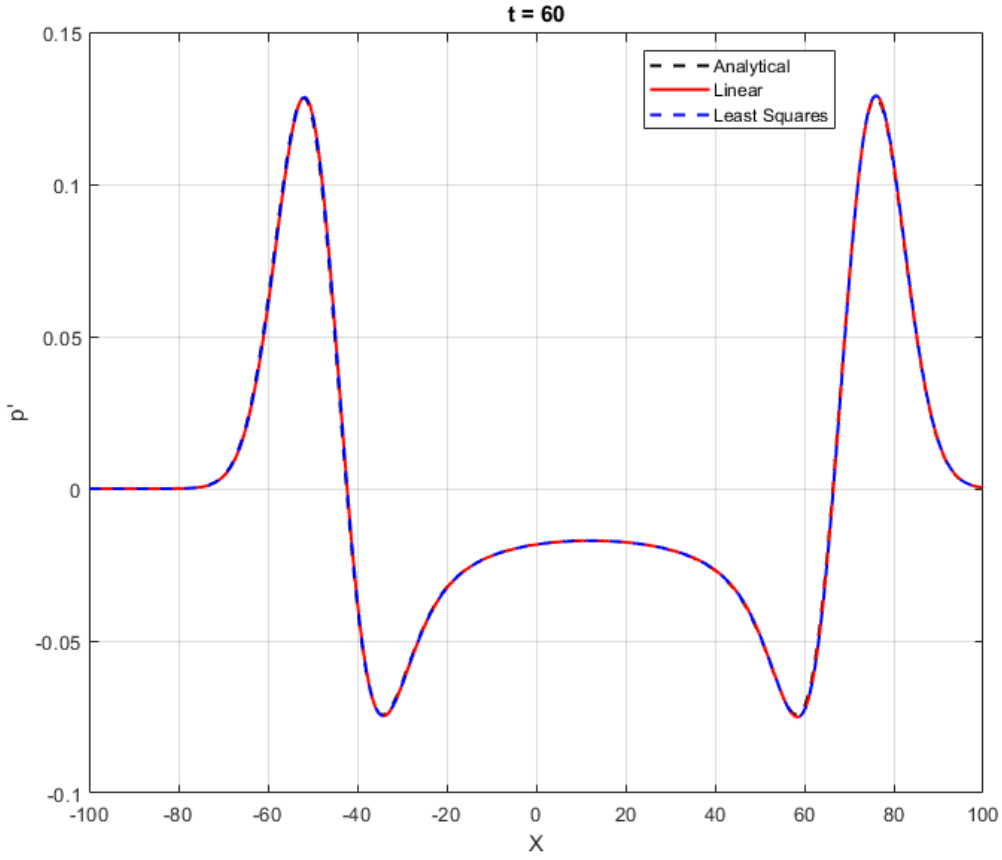
Figure 2.9 Temporal Schemes Test

2.5.3 Gradient Discretization Schemes

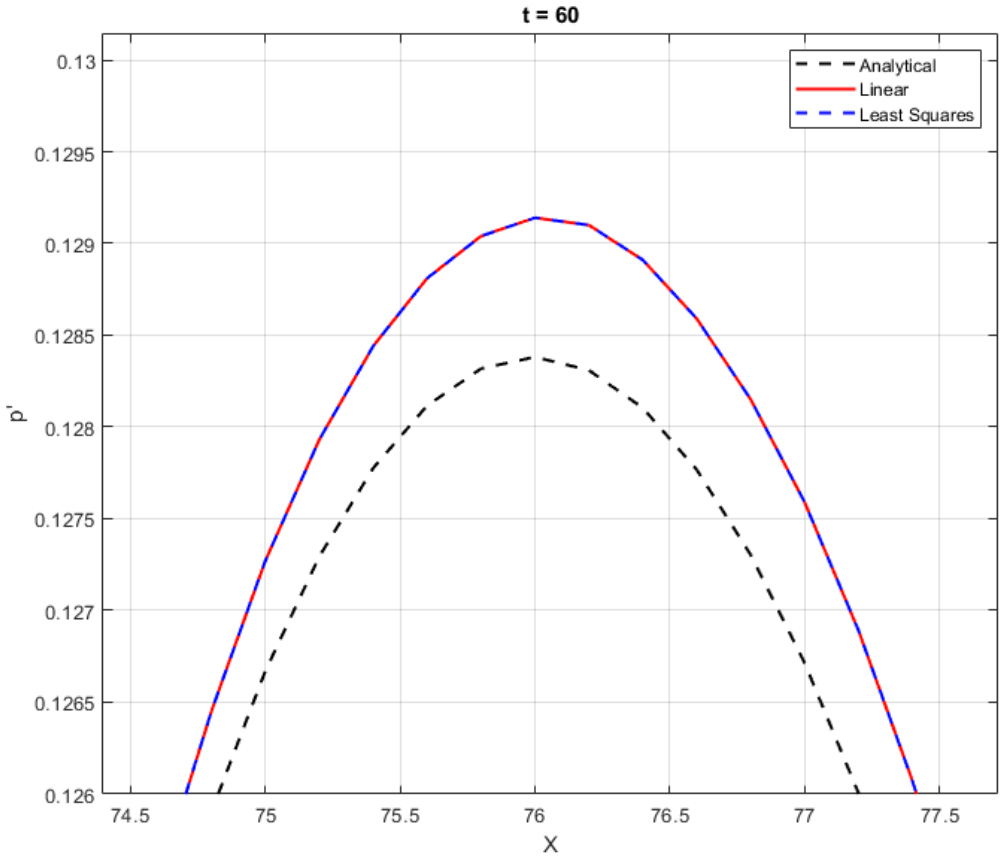
This section tests the performance of the gradient discretization schemes available to the leeFoam solver shown in figure 2.9. The relative errors are shown in table 2.3. As can be seen there is no difference between the two and both are 2^{nd} order accurate schemes. The OpenFOAM documentation claims the least squares method is more accurate but can be oscillatory. This was not noticed in the Gaussian Pulse while this was noticed later in the rectangular jet.

Table 2.3 Gradient Schemes Error

Scheme	**% Relative Error**
Linear	0.0755
Least Squares	0.0755



(a)

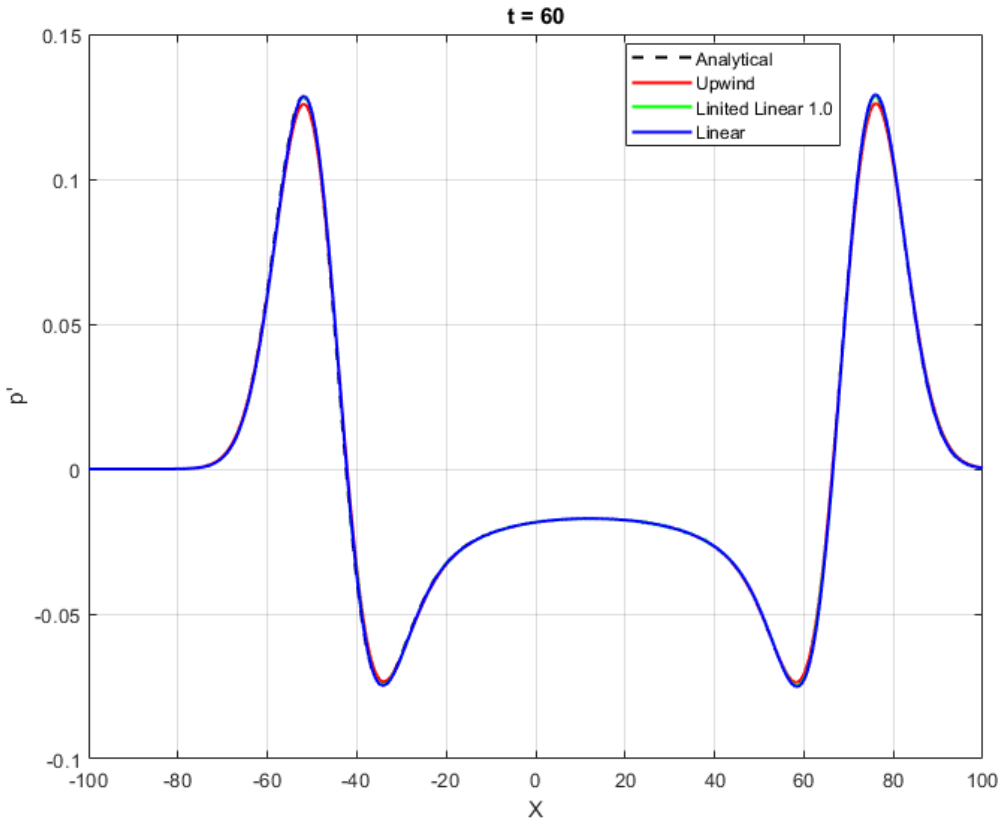


(b)

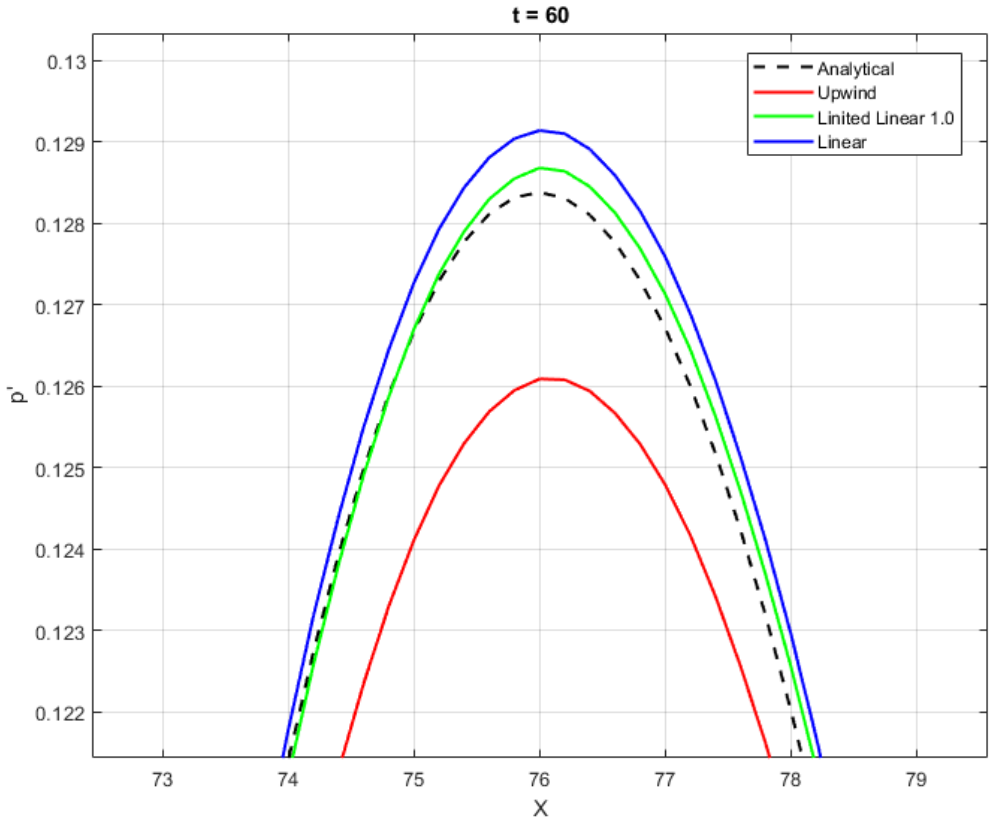
Figure 2.10 Gradient Schemes Test

2.5.4 Divergence Discretization Schemes

This section tests the performance of the divergence discretization schemes available to the leeFoam solver shown in figure 2.10. The relative errors are shown in table 2.4. As can be seen the 1st order accurate Gauss Upwind scheme does affect the solution by showing diffusion error and this would be expected. There is minimal difference between the Gauss Linear and the TVD Gauss Limited Linear 1.0 Scheme.



(a)



(b)

Figure 2.11 Divergence Schemes

Table 2.4 Divergence Schemes Error

Scheme	**% Relative Error**
Upwind	0.1050
Limited Linear 1.0	0.0751
Linear	0.0750

3 The Heated Supersonic Rectangular Jet

This chapter discusses the leeFoam solver use on a heated supersonic rectangular jet. Supersonic jet aircraft either military or upcoming civilian more commonly use rectangular jet design as airframe geometry is rather strict for this purpose [19]. For the civilian side, future air travel at supersonic speeds has a massive obstacle to overcome. Noise regulations are increasingly becoming more stringent and jet noise still poses a massive problem for such aircraft.

The leeFoam solver's ability to produce an estimate of the expected near and far field noise levels quickly and affordably can assist a designer in the aeroacoustics design phase. For example, the cases presented in this chapter on average took between 4-6 hours of CPU time on parallel clusters with less than 4 nodes. This is significantly faster than LES methods. LES methods as will be seen offer a more complete analysis, but the LEE approach can serve as an initial general solution or accompany traditional LES methods.

3.1 Geometry

Figure 3.1 shows the geometry of the rectangular jet nozzle used in this study. This is based on experimental studies by Mora et al. [5]. The design operating mach number of this 2:1 aspect ratio rectangular converging-diverging (C-D) nozzle is Mach 1.5. The temperature ratio (TR) and pressure ration (NPR) are set to design experimental condition by Mora et al. Figure 3.1 shows a drawing of the nozzle and table 3.1 presents the design parameters.

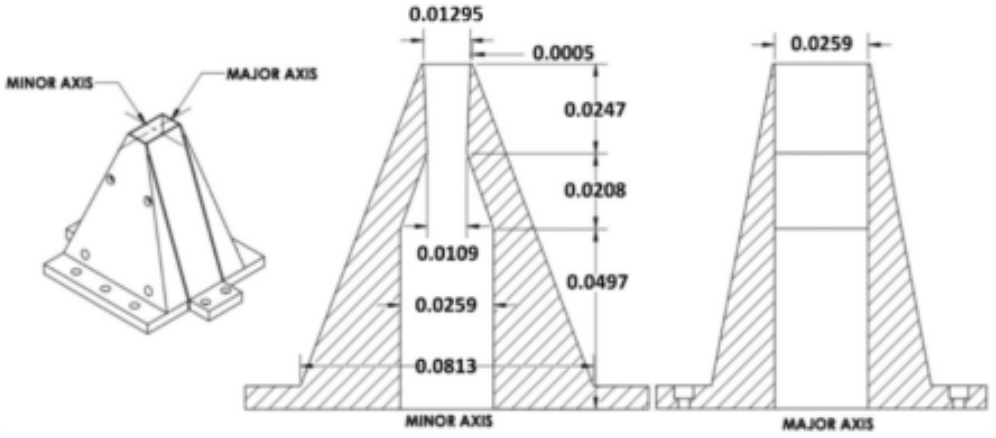


Figure 3.1 Rectangular Converging-Diverging (C-D) Nozzle Geometry [5]

Table 3.1 Rectangular Converging-Diverging (C-D) Nozzle Parameters

Parameter	**Value**
Exit Minor Axis (W)	12.95 [mm]
Exit Major Axis (H)	25.91 [mm]
Equivalent Diameter (D)	20.65 [mm]
Design Mach Number	1.5
Area Ratio	1.18
Nozzle Pressure Ratio (NPR)	3.67
Nozzle Temperature Ratio (TR)	3.0

3.2 Numerical Grid

The numerical grid used in this study follows an acoustic grid approach. The grid is fine enough to properly capture the mean flow structure then expands to meet the minimum points per wavelength (PPW) of 15 cells/wavelength in the far field. In the literature 7 to 15 cells/wavelength is the standard operating practice [20]. Caution must be taken in the expanding process to not exceed a cell volume ratio of 1.015 to neighboring cells. Large cell volume ratios were observed in initial testing to cause wave distortion and internal reflections.

The grid itself extends up to 50 jet diameters axially and 60 jet diameters radially. This is a notably much larger domain than used for LES cases however, with LEE we are capturing the far-field

noise in the domain itself instead of projecting it with FW-H. This is one of the main benefits of performing LEE. Also, the grid in this study has half of the total cells due to the courser nature compared to the LES study from Salehian and Mankbadi. With the simpler calculations involved in LEE and low cell counts this allows the leeFoam solver to produce results within a fraction of the required time compared to LES.

Figure 3.2 illustrates a schematic of the numerical domain with table 3.2 showing the grid parameters. Figure 3.3 shows the 3D numerical grid. Figures 3.4 and 3.5 show the numerical grid sliced in the major and minor planes respectively.

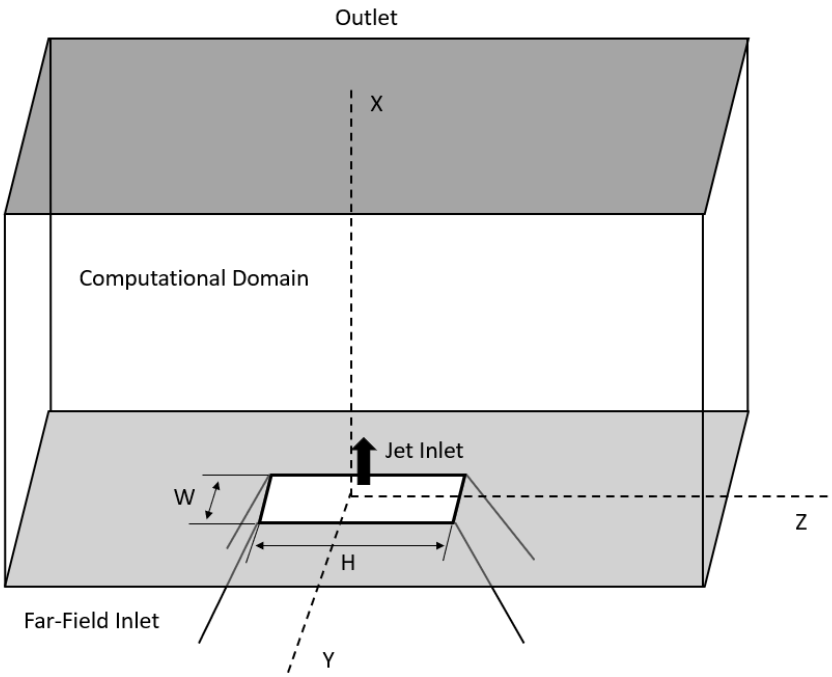


Figure 3.2 Schematic of Numerical Domain

Table 3.2 Grid Parameters

Parameter	**Value**
X-Dir	50D
Y-Dir	55D
Z-Dir	55D
Cell Count	41 million (60 million for St = 0.2 case)

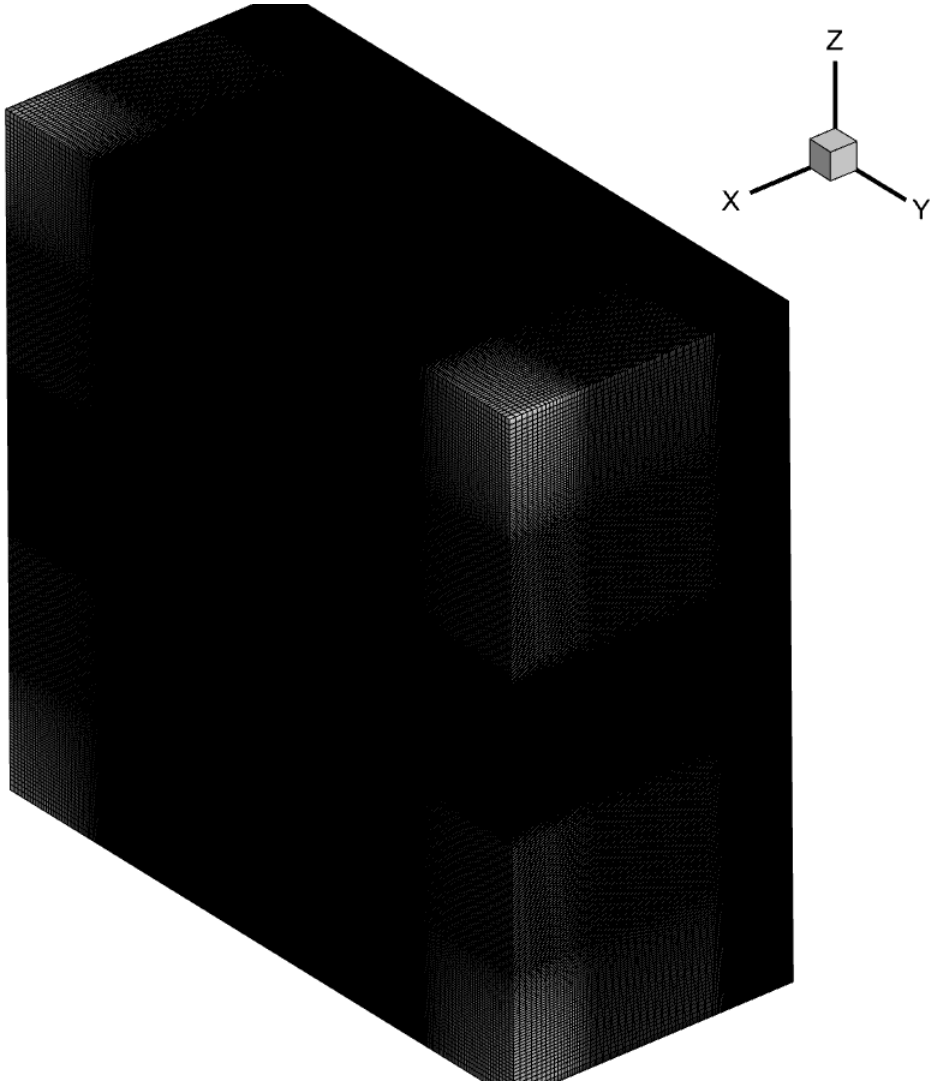


Figure 3.3 3D Numerical Grid

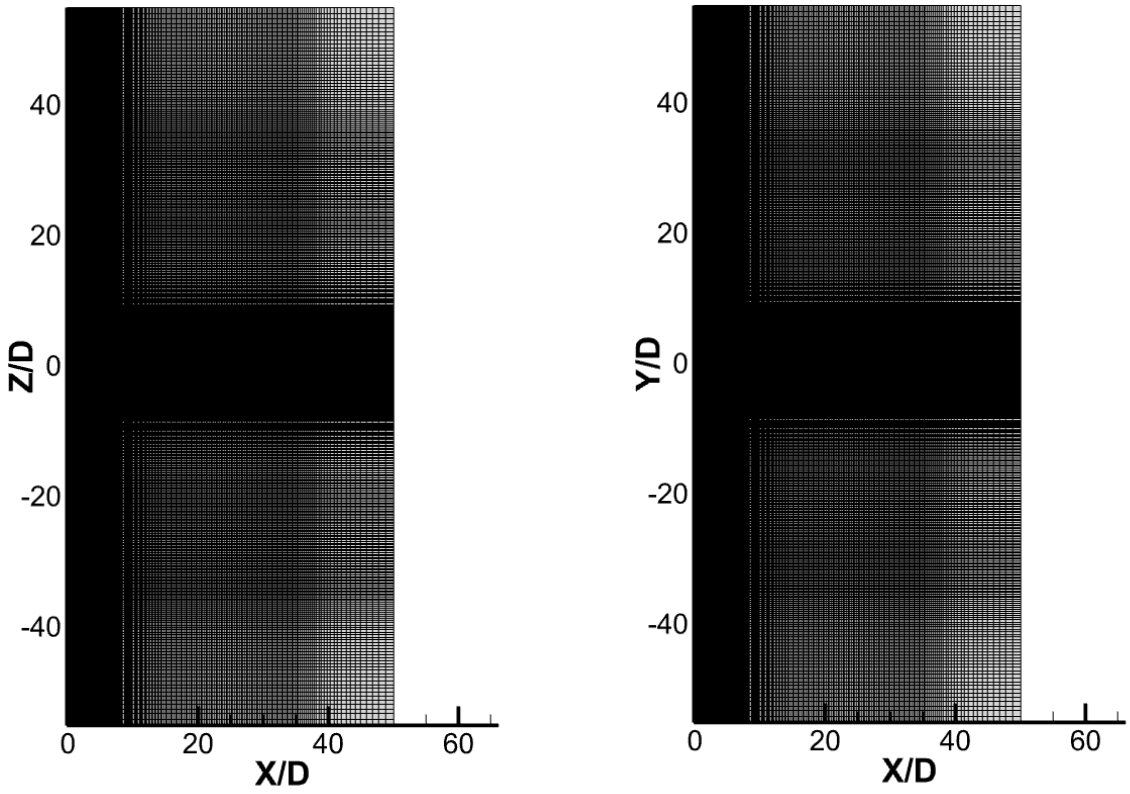


Figure 3.4 Major and Minor Planes Numerical Grid

3.3 Mean Flow

The LEE's require a mean flow to be given which can be from an analytical or a verified RANS solution. For the rectangular jet case, a RANS solution was given from the Salehian and Mankbadi paper. The authors did not present their RANS solution in the paper as their main focus was on LES results but, since RANS is commonly run before transitioning to LES it was available. The provided RANS solution was run further to confirm convergence and accuracy was compared to mean flow data provided by an experiment. See figure 3.5.

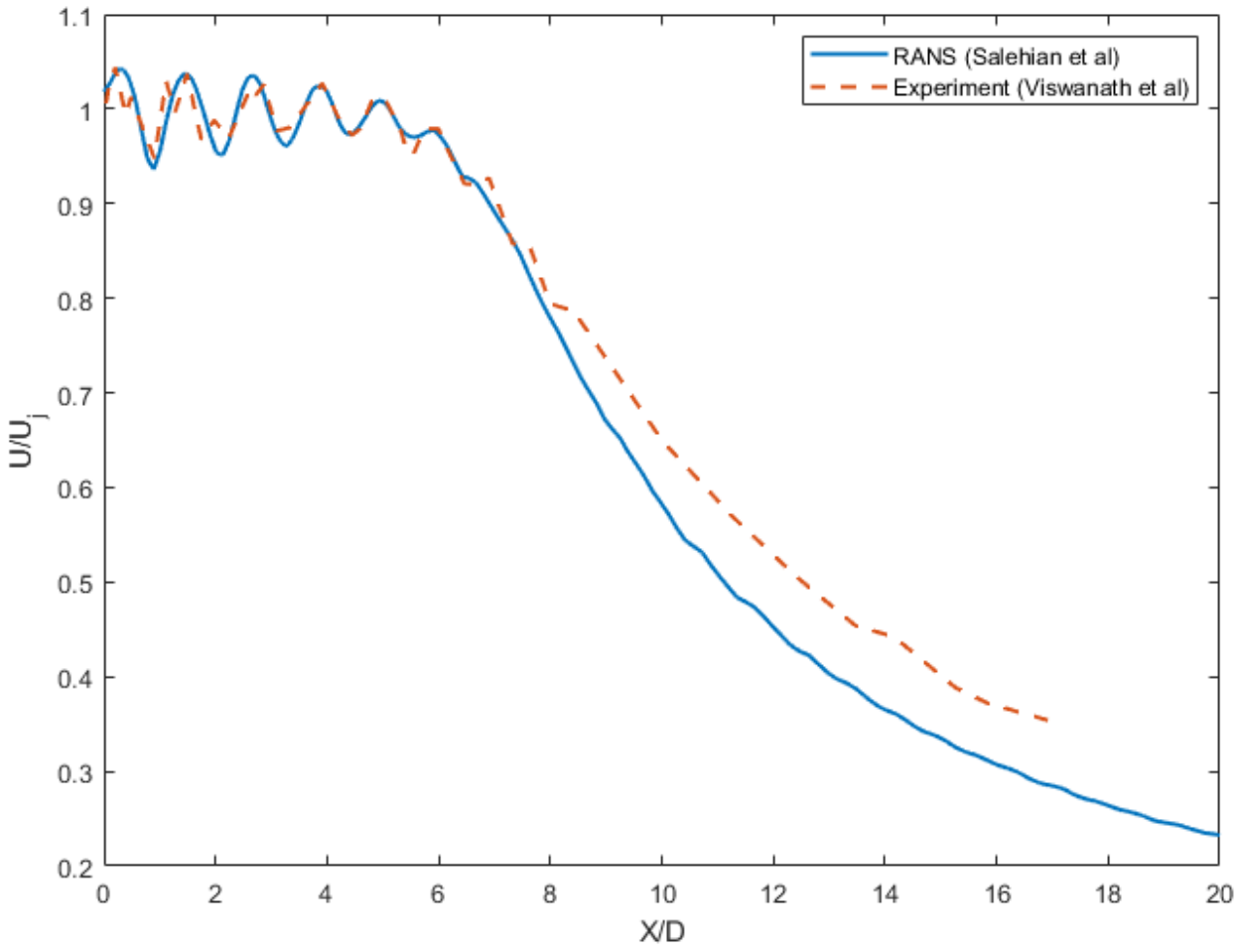


Figure 3.5 Jet Core Mean Flow Axial Velocity along Jet Centerline vs Experiment [19,24]

The rectangular jet is unique compared to traditional round jets. The jet core emerging from the nozzle is initially rectangular but then transitions to a round state. This can be seen in the 3D mean flow figures 3.6 and 3.7.

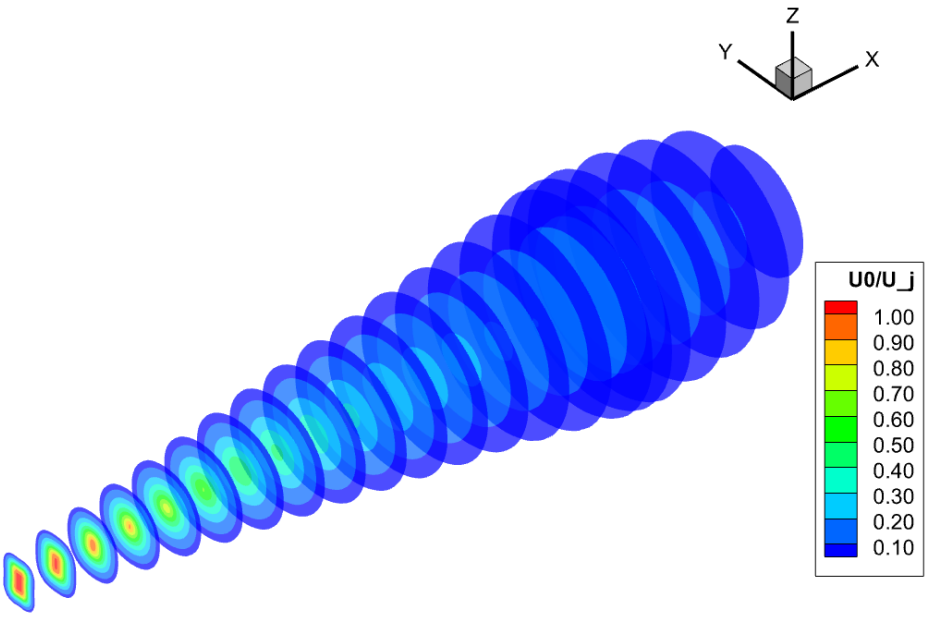


Figure 3.6 Jet Velocity Mean Flow

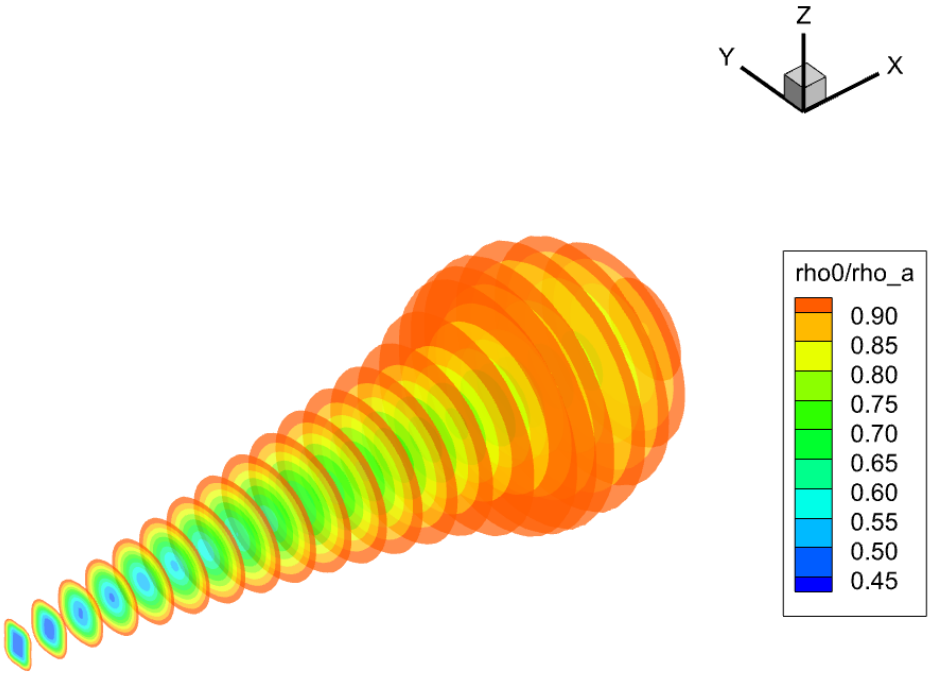


Figure 3.7 Jet Density Mean Flow

Figures 3.8 to 3.11 show the u, v, w mean flow quantities along the major and minor planes.

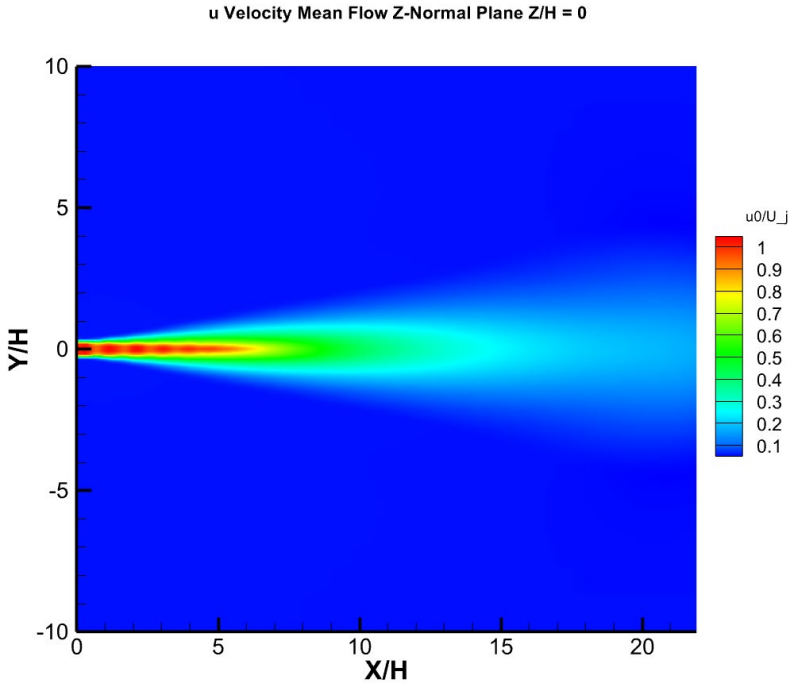


Figure 3.8 u Velocity Mean Flow along Major Plane

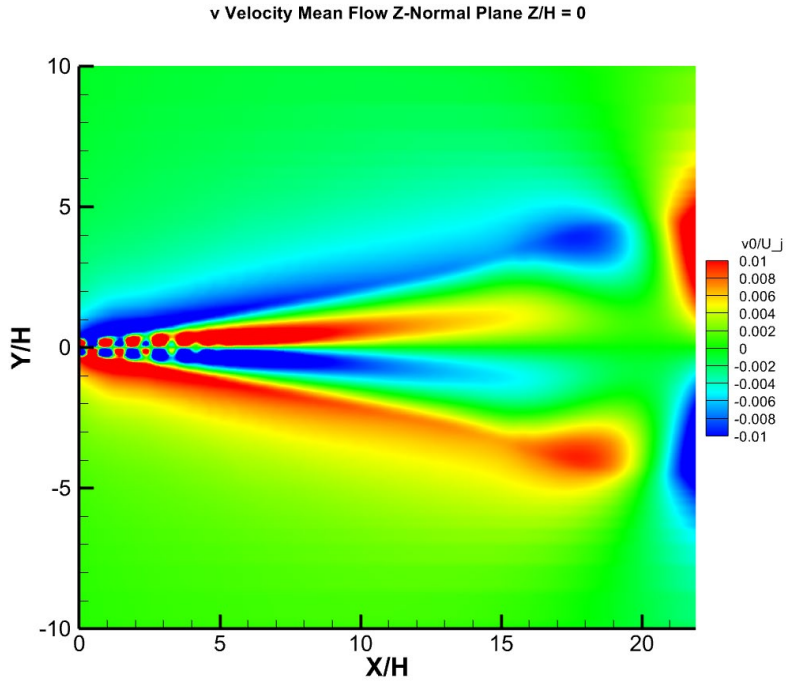


Figure 3.9 v Velocity Mean Flow along Major Plane

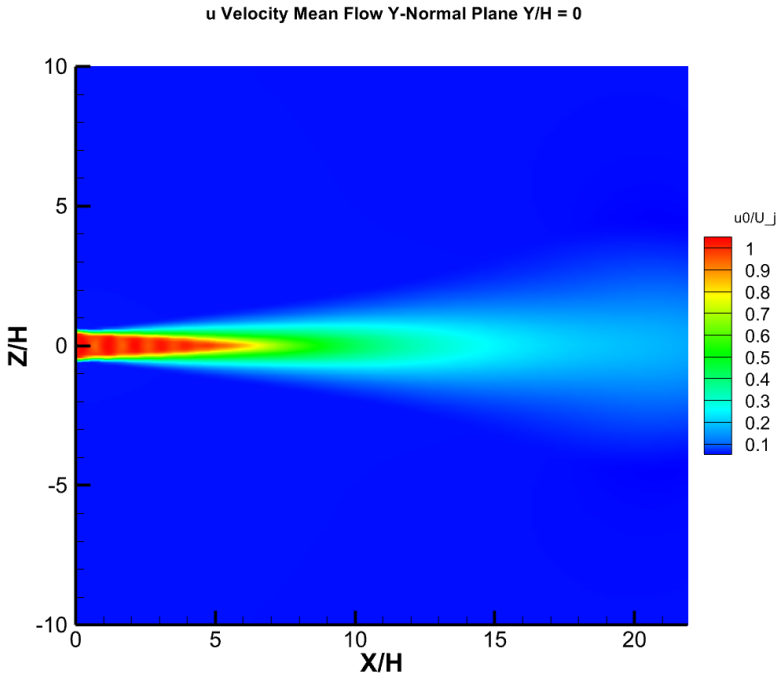


Figure 3.10 u Velocity Mean Flow along Minor Plane

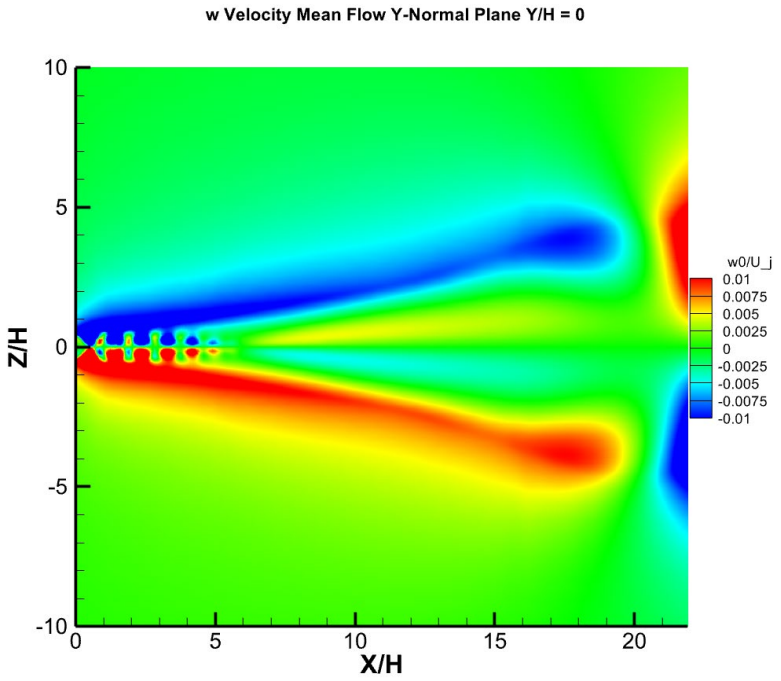


Figure 3.11 w Velocity Mean Flow Along Minor Plane

Figures 3.12 and 3.13 show the jet density mean flow in the major and minor planes respectably.

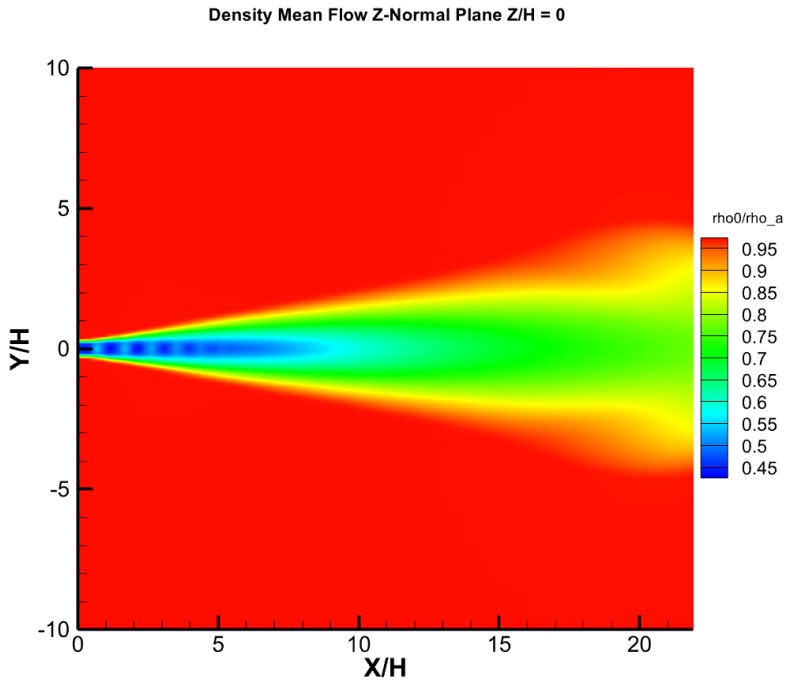


Figure 3.12 Density Mean Flow Along Major Plane

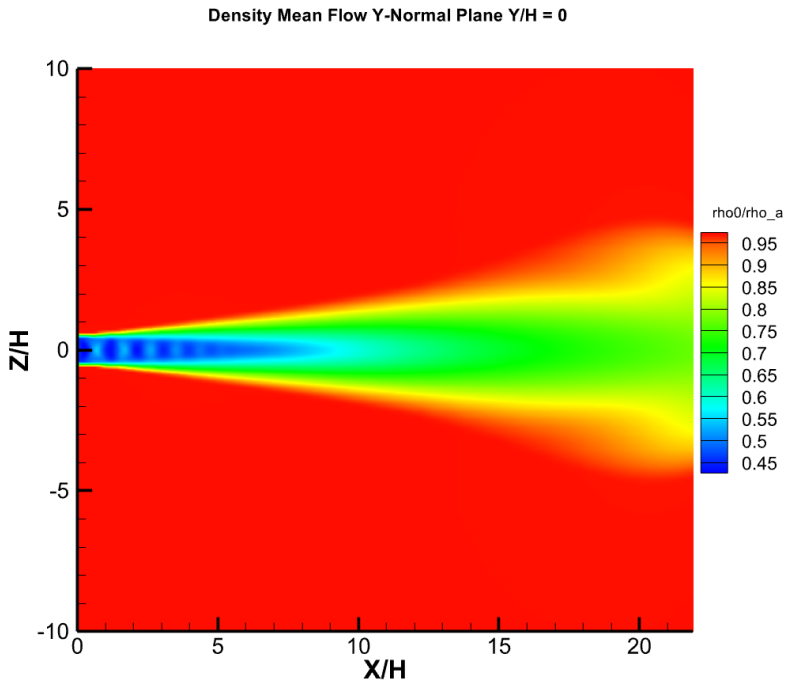


Figure 3.13 Density Mean Flow Along Minor Plane

Figures 3.14 and 3.15 show the jet pressure mean flow in the major and minor planes respectably.

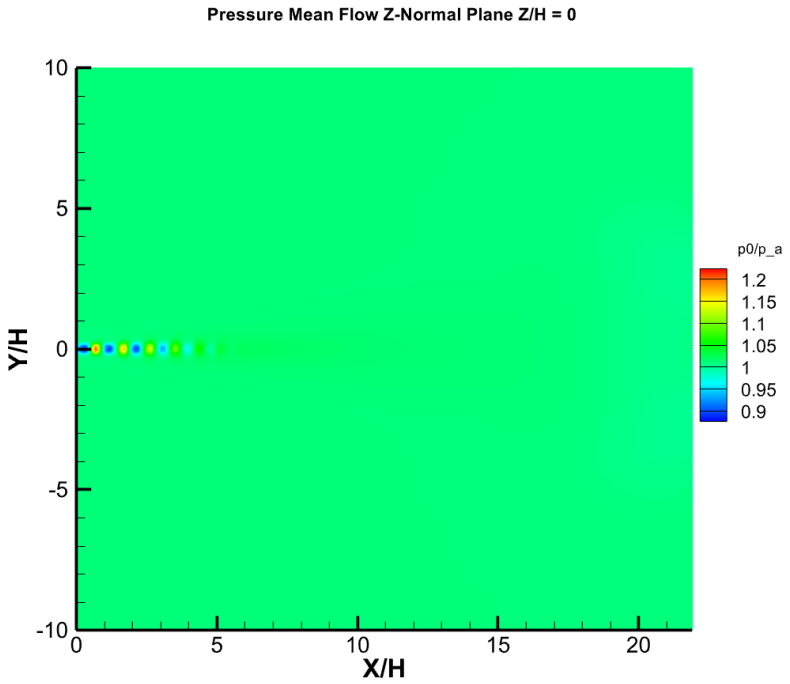


Figure 3.14 Pressure Mean Flow Along Major Plane

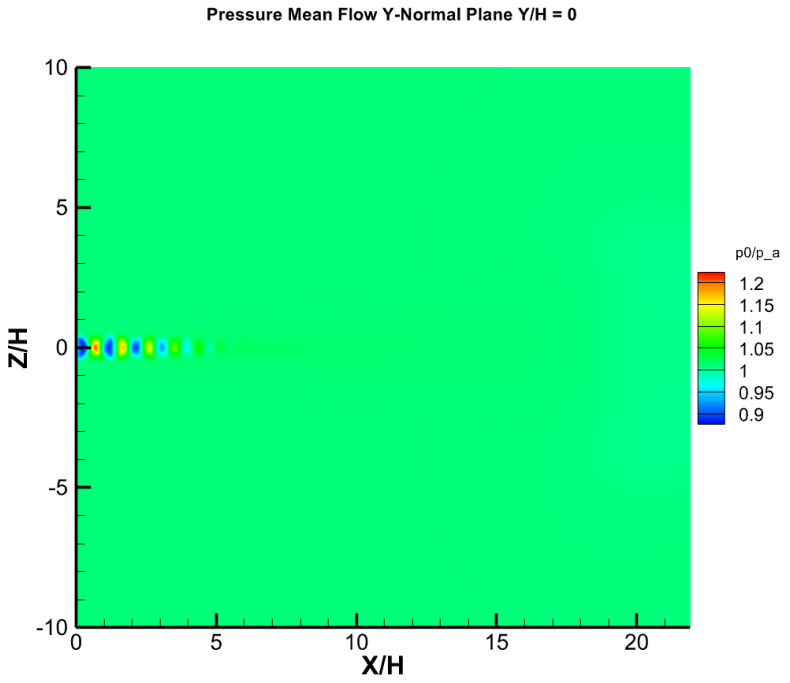


Figure 3.15 Pressure Mean Flow Along Minor Plane

3.4 Numerical Methods

OpenFOAM has several finite-volume schemes available to choose from. This was one of the benefits of choosing OpenFOAM as the developing environment for this code as the numerics are already present. No work was required to code any custom numerical schemes but, it was required to select the proper numerical schemes. As it was important to understand how each available scheme affects the code and the rectangular jet case.

3.4.1 Finite-Volume Schemes

The schemes chosen for the rectangular jet case are outlined in table 3.3 below. These schemes were chosen based on their best ability to maintain wave properties and provide stability. The details of these schemes were presented in chapter 2. Note the use of TVD schemes. TVD schemes limit towards upwind in steep gradients. With sharp mean flow gradients in the jet core terms dealing with the mean flow benefit from this upwind limiting approach. This helps provide stability with small amounts of numerical diffusion.

For temporal discretization the Crank-Nicolson 0.9 with a ACo# of 0.1 was used. This is not quite 2^{nd} order accurate, but it does provide stability at the slight expense to accuracy. It is notable that most CAA codes, especially using Finite- Difference (FDM) utilize high order schemes that can achieve 4^{th} order accuracy. However, when utilizing OpenFOAM's 2^{nd} order schemes it is shown the solver can remain accurate and stable. Other test cases were run with 1^{st} order Euler temporal schemes and the ACo# was able to be much higher but being closer to pure 2^{nd} order is preferred.

Table 3.3 Finite-Volume Numerical Schemes for the Rectangular Jet

	Scheme	**Order**
Time	Crank-Nicolson 0.9	≈2nd
Gradient	Gauss Linear	2nd
Divergence	Limited Linear 1.0	2nd

3.4.2 Boundary Conditions

The boundary conditions (BC) play an important role especially in acoustics simulations. It is very important reflections from the outflow boundaries are minimized as much as possible. Reflections can interact with the simulation producing unrealistic results or even instabilities causing a simulation to fail. For an LEE code this is a magnified problem as spurious modes can be introduced at the boundaries [8].

The inflow boundary condition acts as the input disturbance to the leeFoam solver. The location is shown in figure 3.16. In this case a direct sinusoidal disturbance is used as to simulate an instability wave type disturbance.

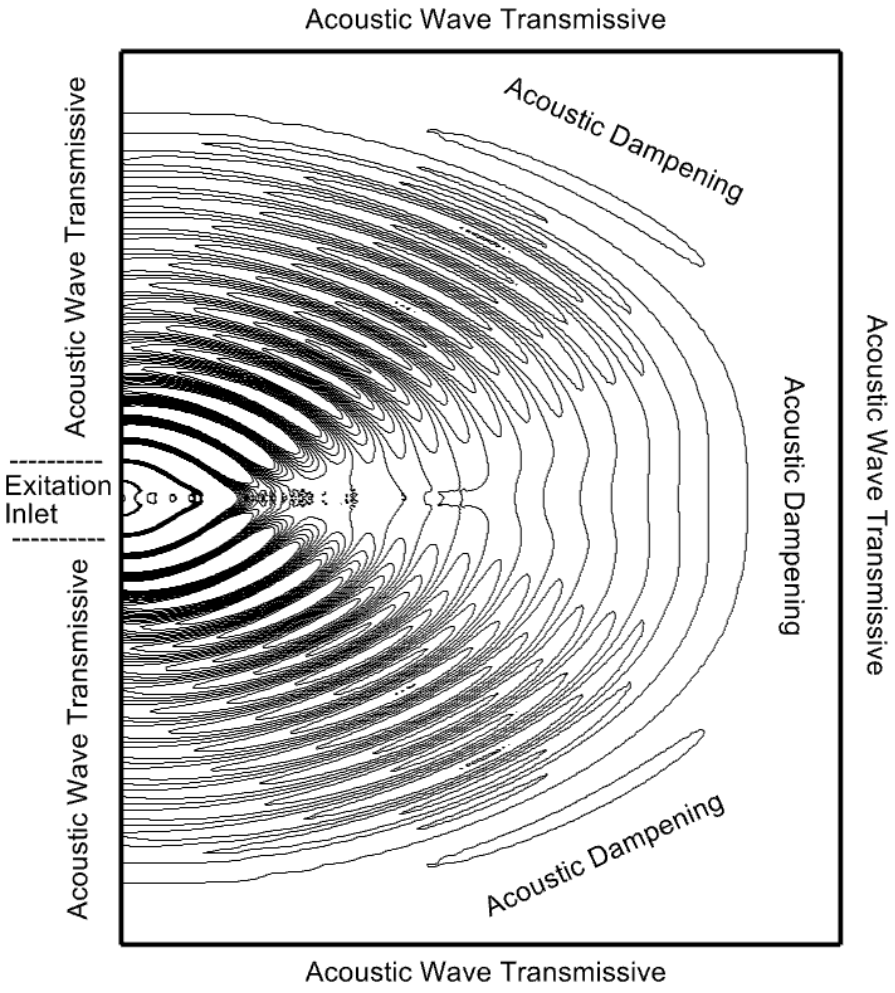


Figure 3.16 Inflow Boundary location

To produce the disturbance needed for the 3D jet LEE simulation the inflow boundary is an oscillating type of boundary that varies u' with time. This boundary is custom coded into OpenFOAM and follows equations 3.1-3.3. Equation 3.1 describes the sinusoidal oscillation. A fraction of the jet exit velocity from the mean flow is varied with time at a frequency described by equation 3.2. The frequency is dependent on the Strouhal Number and jet exit parameters. This Strouhal Number chosen for this simulation is based on the peak acoustic frequency determined by Salehian and Mankbadi [20] and is approximately St = 0.1.

$$u' = u_{ex}(0, y, t)) = \epsilon \, Sin(wt) \, \bar{u}(0, y) \tag{3.1}$$

With,

$$\omega = St \left(\frac{4\pi U_j}{H}\right) \tag{3.2}$$

$$\epsilon = 0.1 \tag{3.3}$$

The boundary layer at the inlet is also modeled in the disturbance in order to prevent any discontinuous gradients. A discontinuous gradient in the disturbance can cause a discontinuity that leads to a solution failure event. The boundary layer is modeled by a hyperbolic tangent function that is applied to the mean flow parameter as shown in equation 3.4 below. Table 3.4 provides parameters for the input disturbance.

$$\bar{u}(y^+, z^+) = \frac{\bar{u}}{2}\left(1 + \tanh\left(\frac{\eta_x}{4\theta^+}\left(\frac{\eta_x}{|y^+ + z^+|} - \frac{|y^+ + z^+|}{\eta_x}\right)\right)\right) \tag{3.4}$$

Table 3.4 Inflow Boundary Condition Parameters

Parameter	**Value**
St	0.05 to 0.2 (by case)
$\boldsymbol{\epsilon}$	0.075
H	20.65 [mm]
$\boldsymbol{U_j}$	750 [m/s]

For this simulation the outflow boundaries use the built in *acusticwaveTransmissive* in OpenFOAM. The locations are shown in figure 3.16. The *acusticwaveTransmissive* boundary condition was discussed in chapter 2. Boundary dampening was also applied as was discussed in chapter 2 and is shown in figure 3.17.

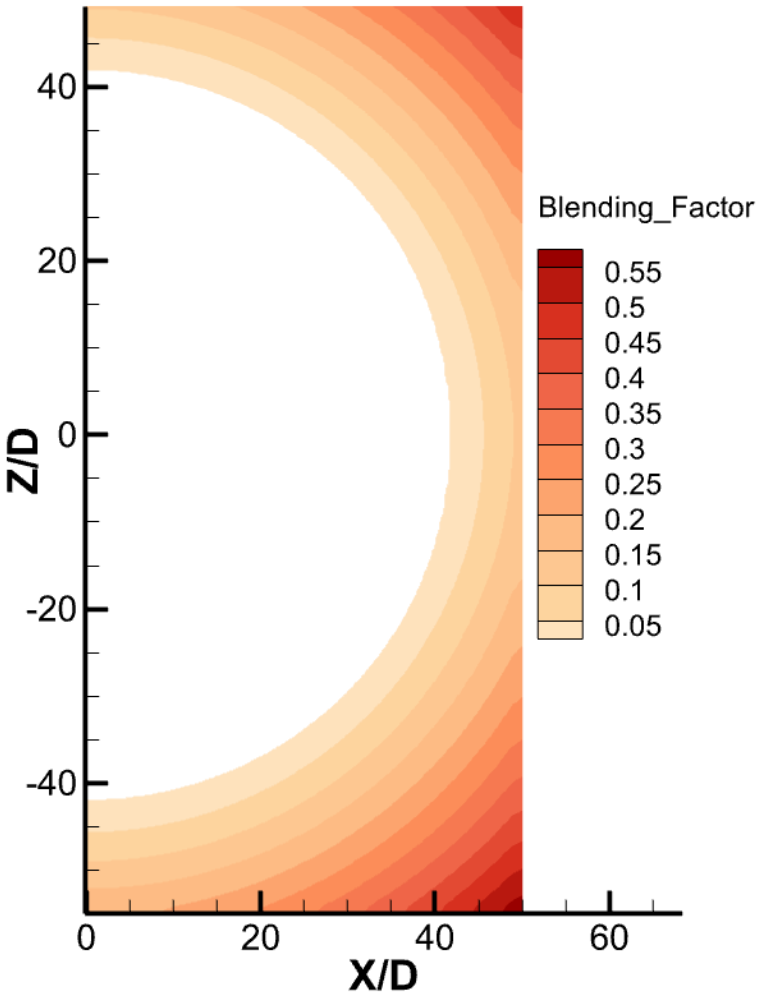


Figure 3.17 Dampening Sponge Zone

3.4.3 Acoustic Relaxation Term

As mentioned earlier an acoustic relaxation term (ART) can be applied to the LEEs in order to dampen/kill unwanted numerical noise. These spurious frequencies can develop due to grid issues, numerical schemes, or time stepping. ART has been shown to provide decent frequency selective properties in the rectangular jet case while minimizing diffusion error [17]. Figure 3.18 demonstrates ART's effects as frequency is increased. These tests were run experimentally on the

1D wave test with reasonably similar mean flow conditions as the jet core. The shock formations are omitted thus to keep the test simple and achieve the overall objective.

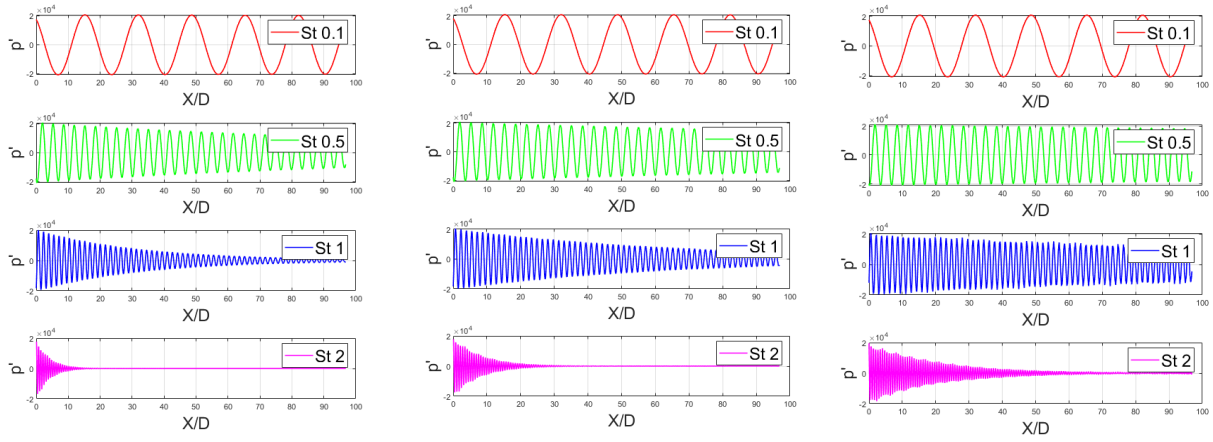


Figure 3.18 ART Tested Experimentally

The ART coefficient can also be determined analytically as shown in the Bernicke paper [17]. The derivation comes from the imaginary component of the 1-D wave equation solution in Fourier space [17]. See equation 3.4. By setting a fixed ART coefficient we can plot the resulting frequency vs damping as shown in figure 3.19. Notice how the damping increases exponentially as the frequency increases. The goal is to maximize the damping at the higher frequencies while maintaining little to no diffusion for the wanted frequencies. ART in this case was chosen to be 0.04.

$$k_i(\omega,\sigma) = -\frac{\sqrt{2}}{2}\sqrt{\frac{\omega^2(\sqrt{\omega^2\sigma^2+1}-1}{c_0^2+\omega^2\sigma^2}} \tag{3.4}$$

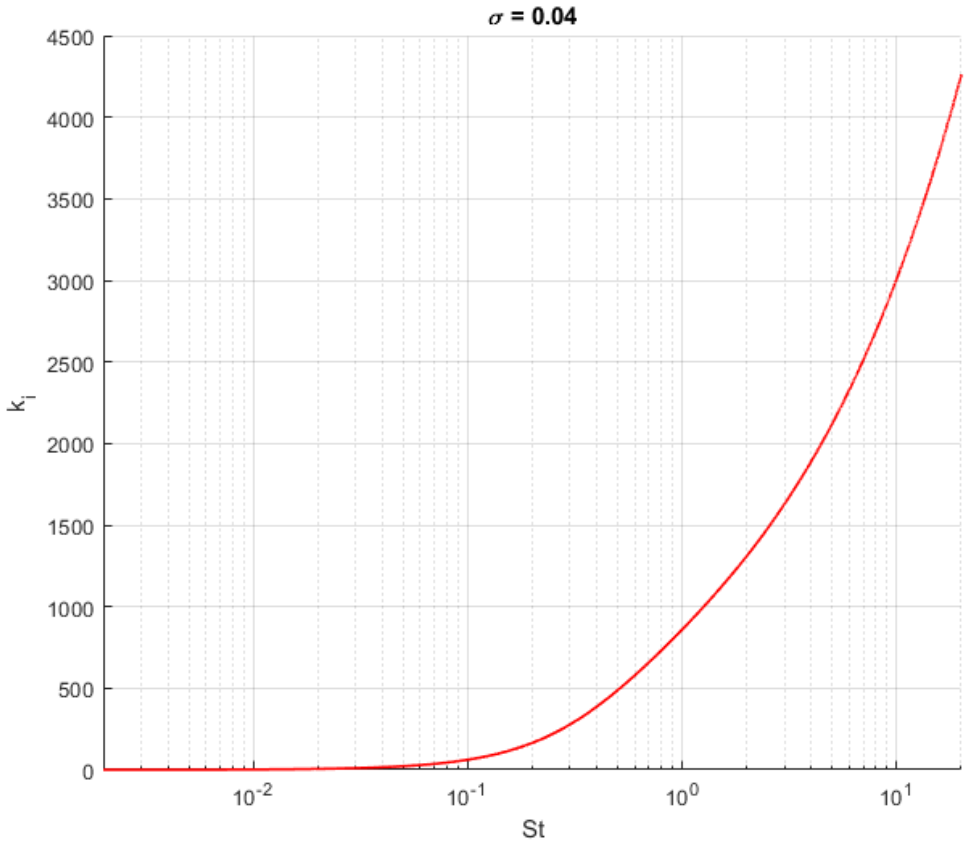


Figure 3.19 ART Analytical Coefficient

4 Rectangular Jet Results

This chapter will present the results from the rectangular jet case. The still images for the major and minor planes along with Root Mean Square contours, OASPL contours, Directivity, and convergence criteria will be presented. Limited discussion will be provided as the main concluding arguments are presented in Chapter 5 Conclusions.

Sections 4.1 and 4.2 split the still images between the major and minor planes with the data previously mentioned. Figure 4.1 shows an illustration of the locations of the major and minor planes in the domain. Sound Pressure Level (SPL) contours are calculated with equation 4.1.

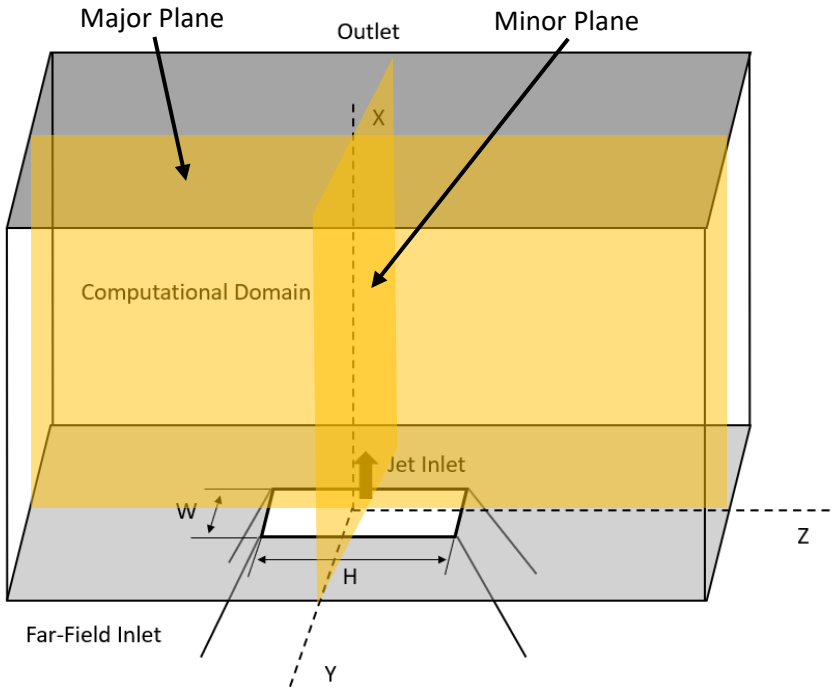


Figure 4.1 Illustration of Major and Minor Planes

$$SPL = 20\log_{10}\left(\frac{p'_{RMS}}{2E-05}\right) \tag{4.1}$$

4.1 Sliced Snapshots in Major Planes

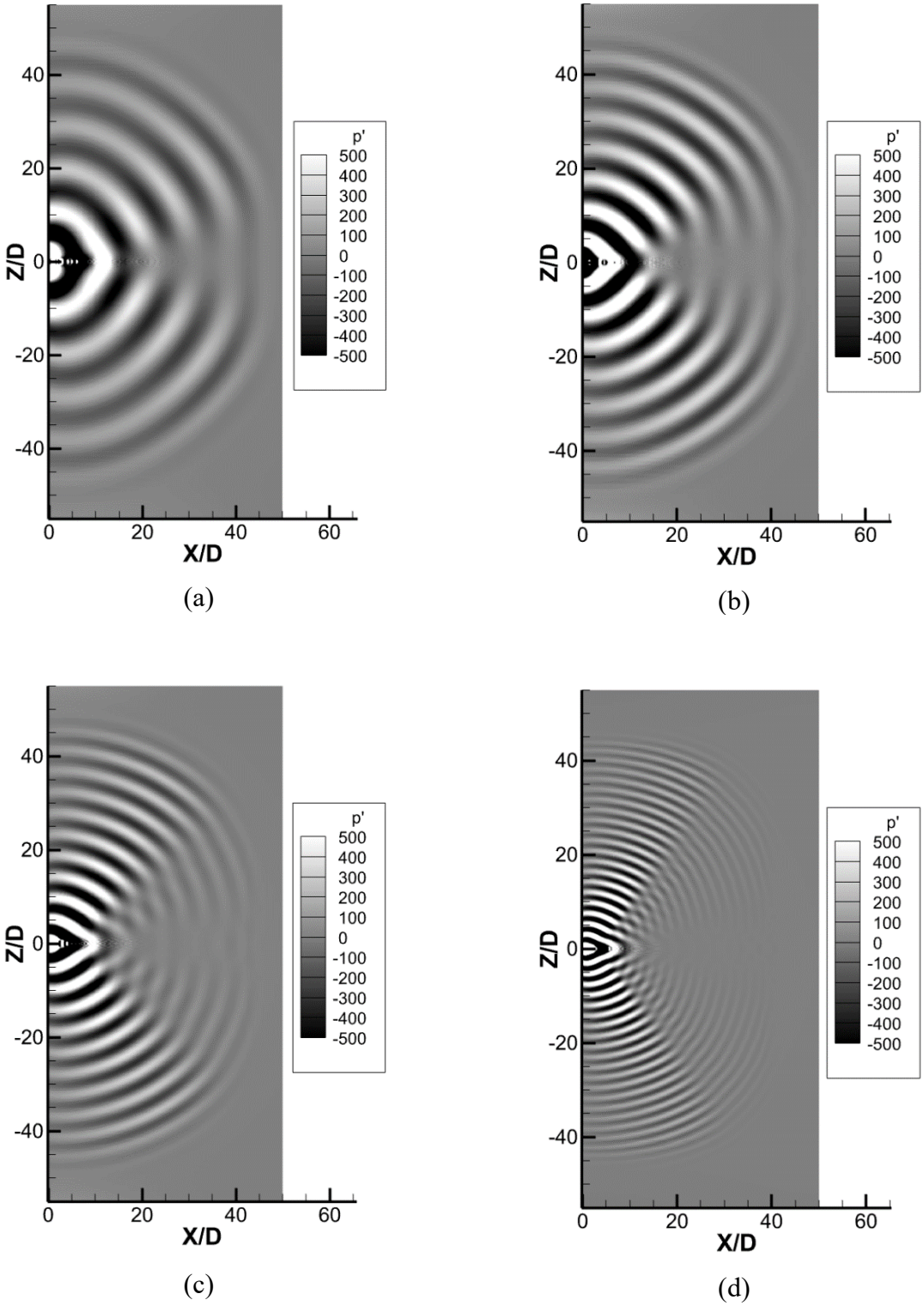


Figure 4.2 p' contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

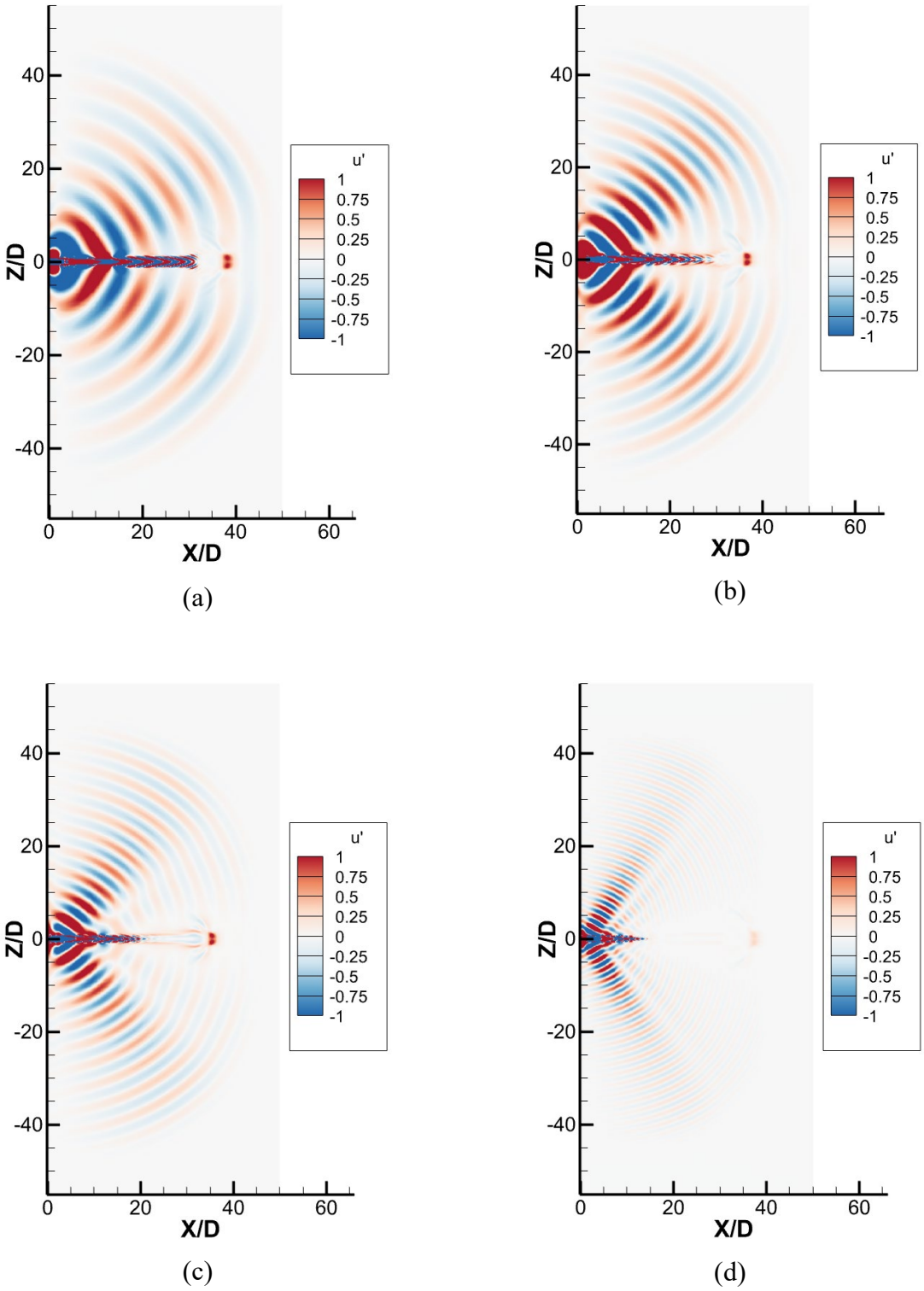


Figure 4.3 u' contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

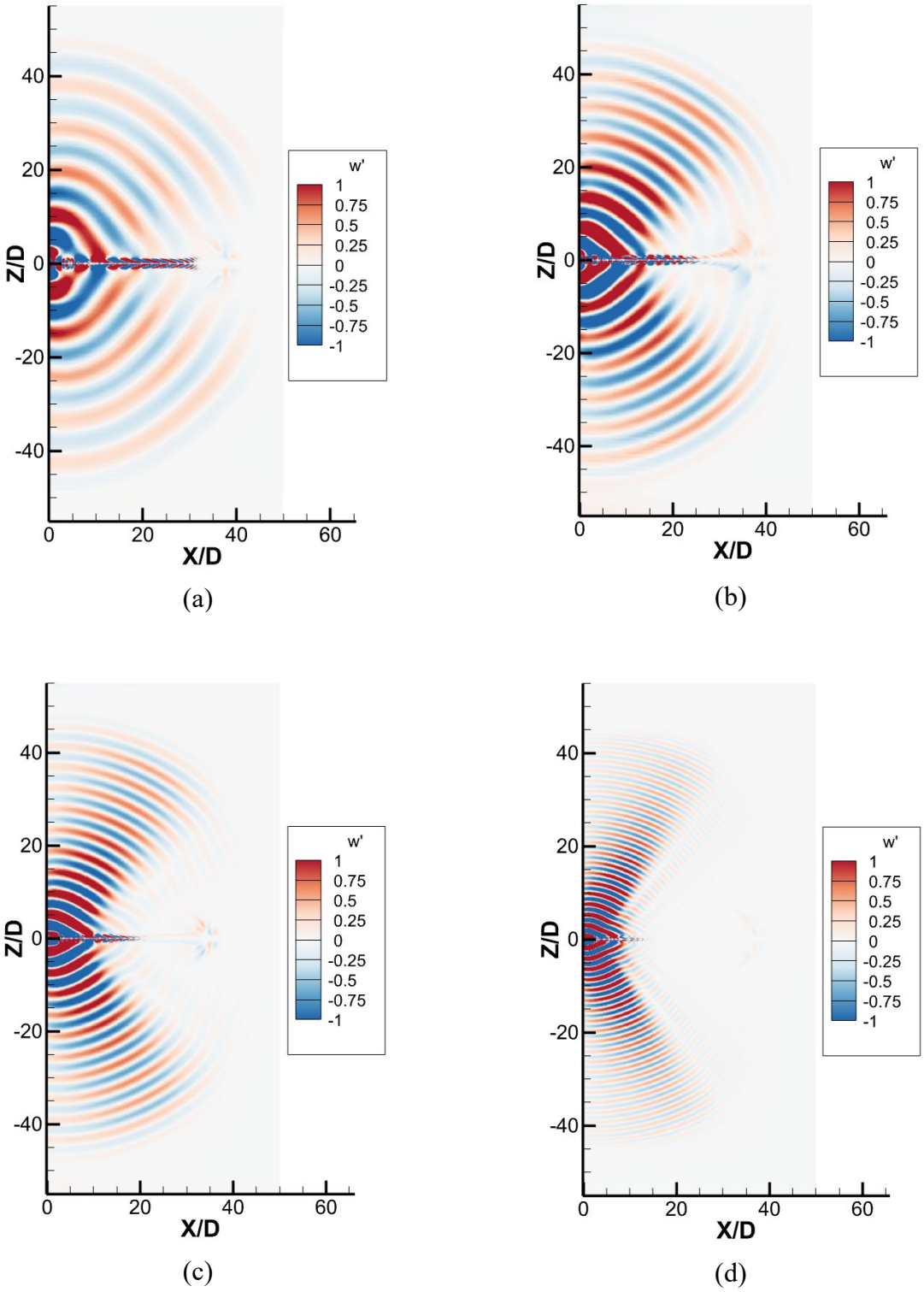


Figure 4.4 w’ contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

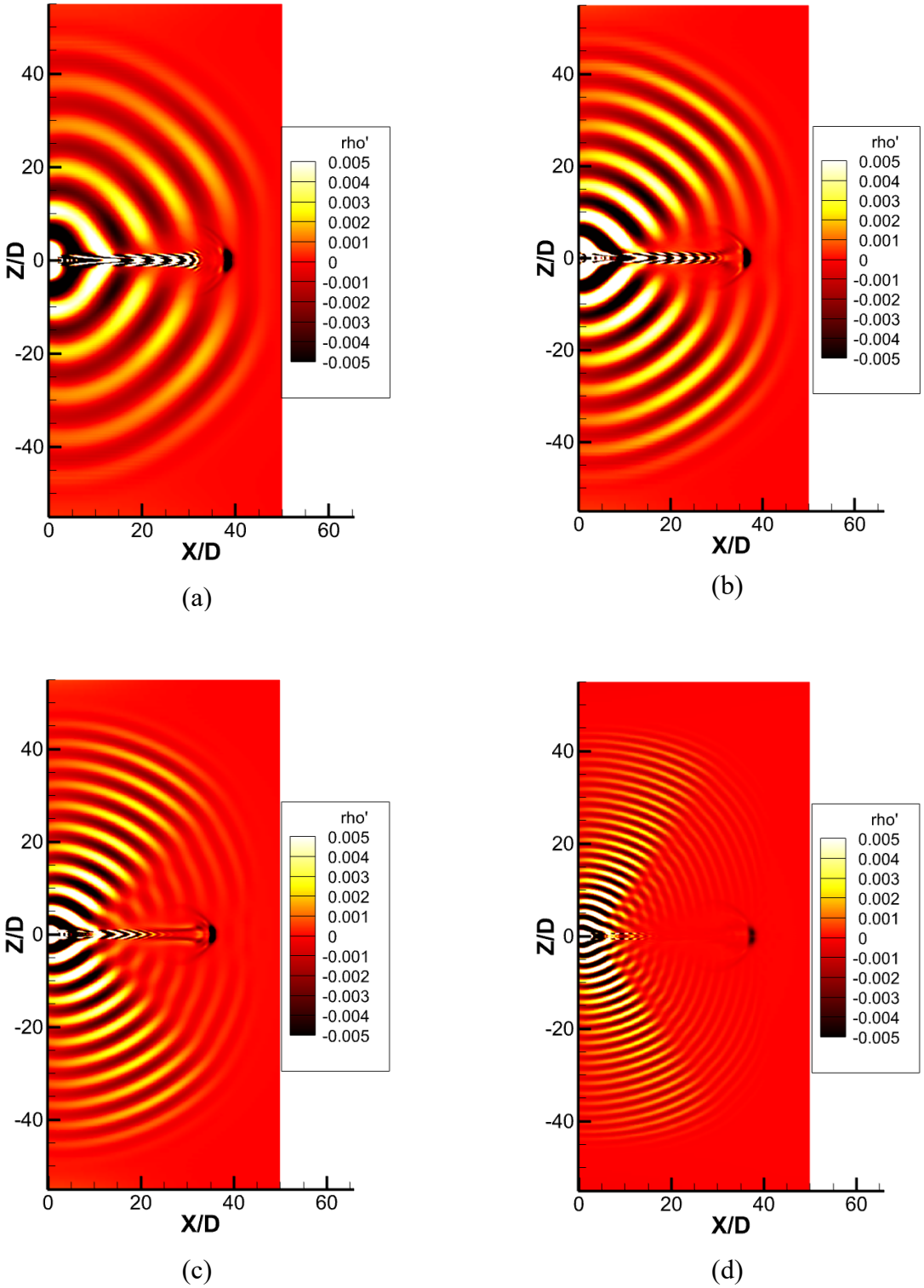


Figure 4.5 rho' contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

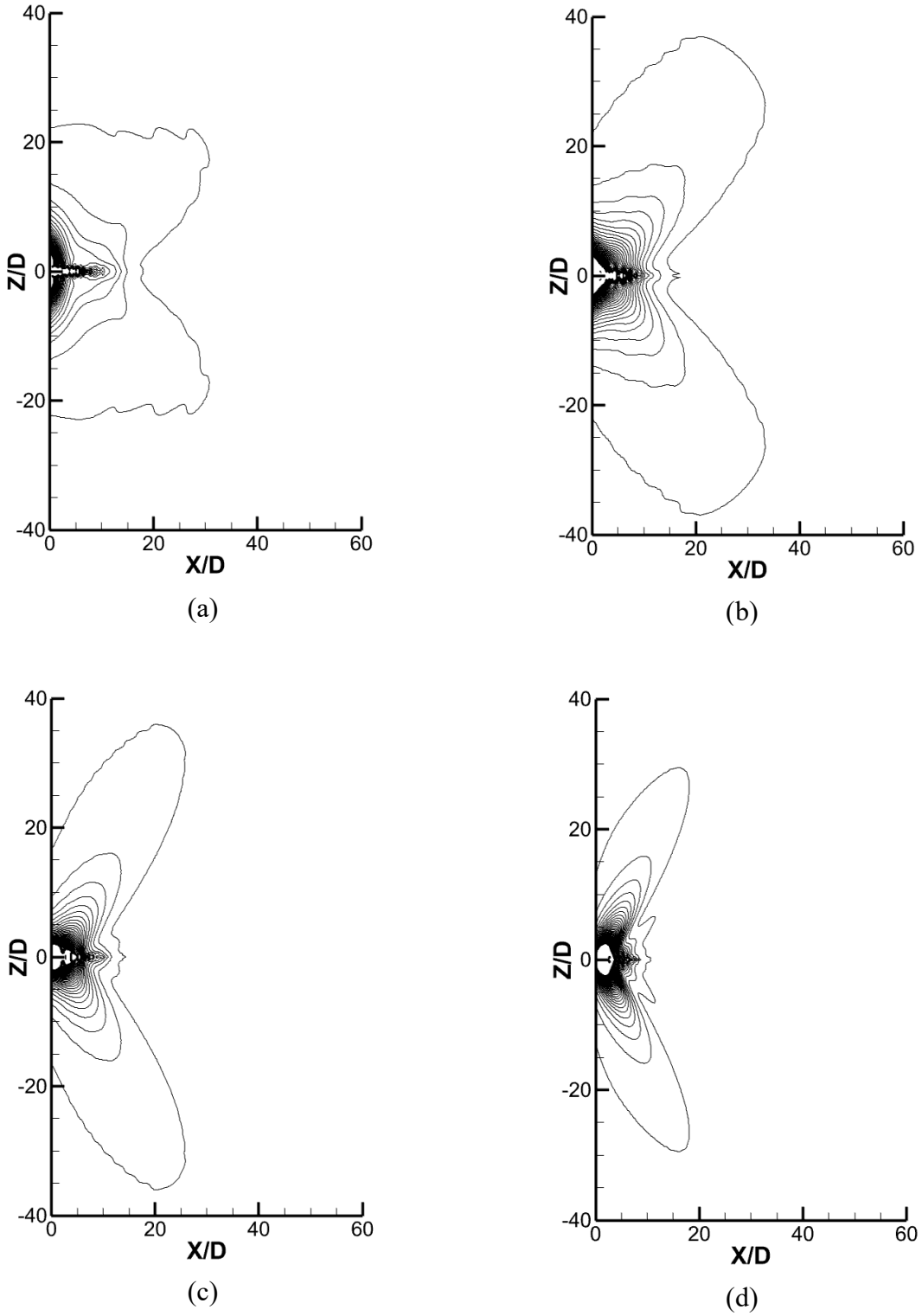


Figure 4.6 p’ rms contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

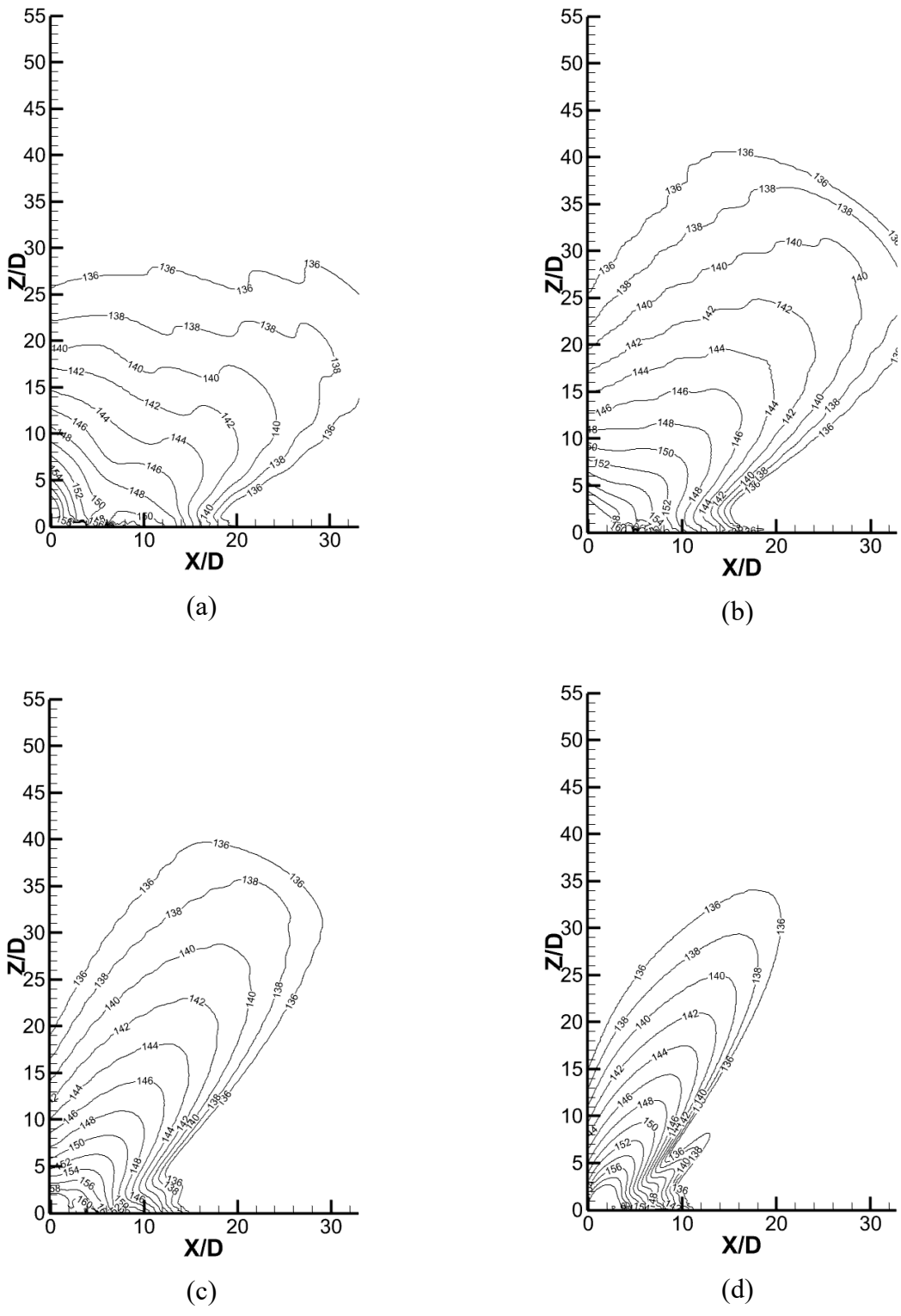


Figure 4.7 SPL contours major plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

4.2 Sliced Snapshots in Minor Planes

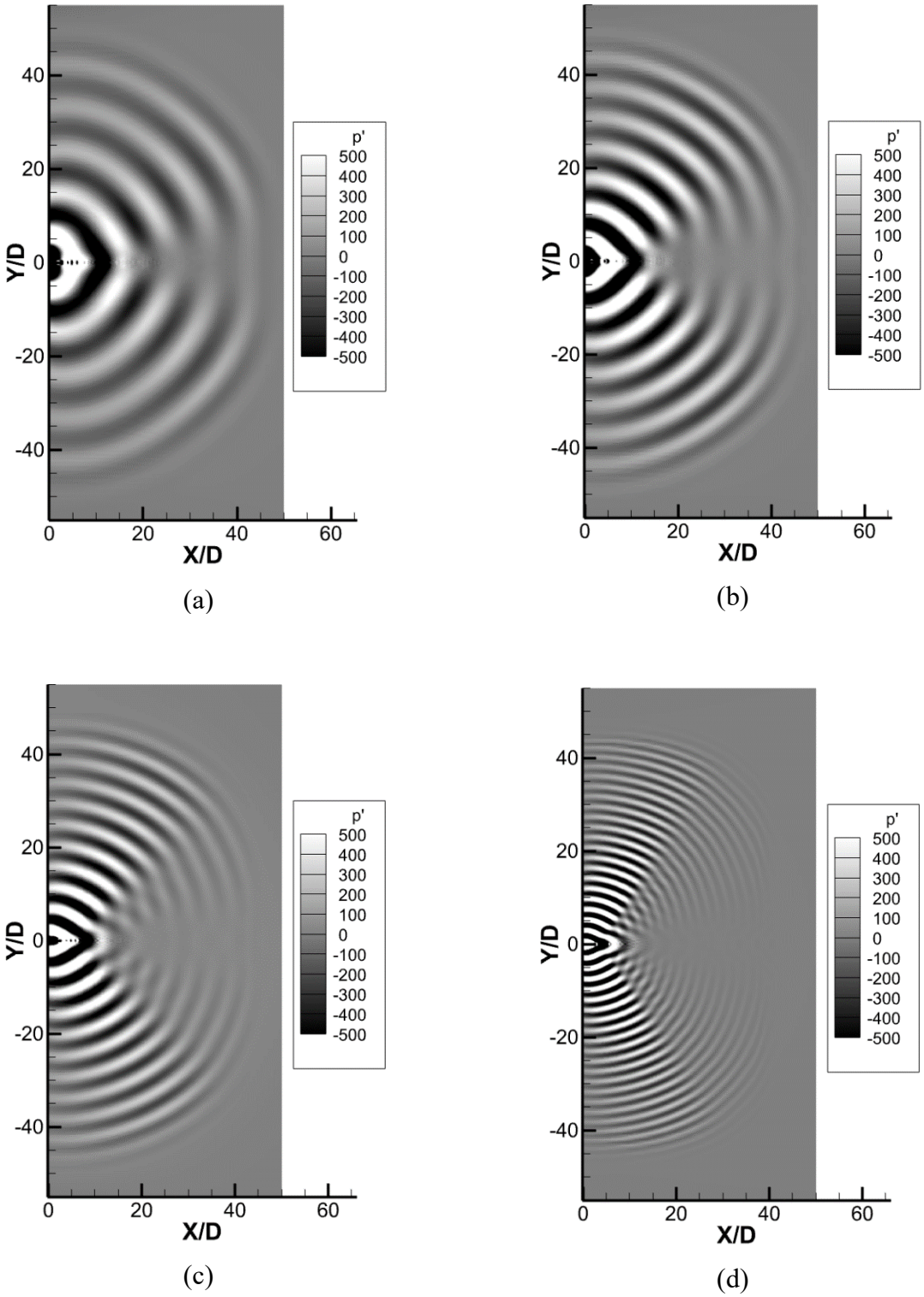


Figure 4.8 p' contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

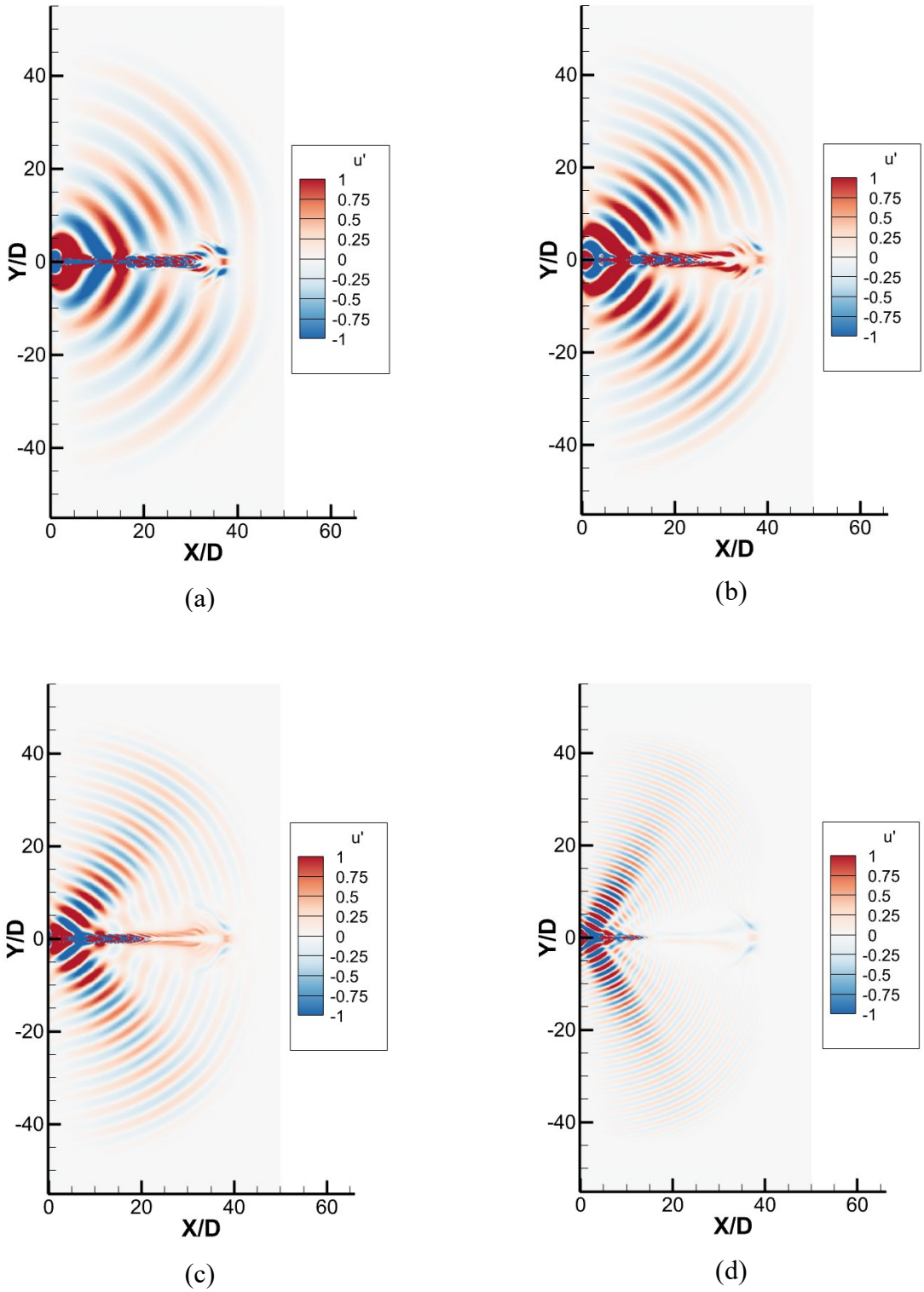


Figure 4.9 u’ contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

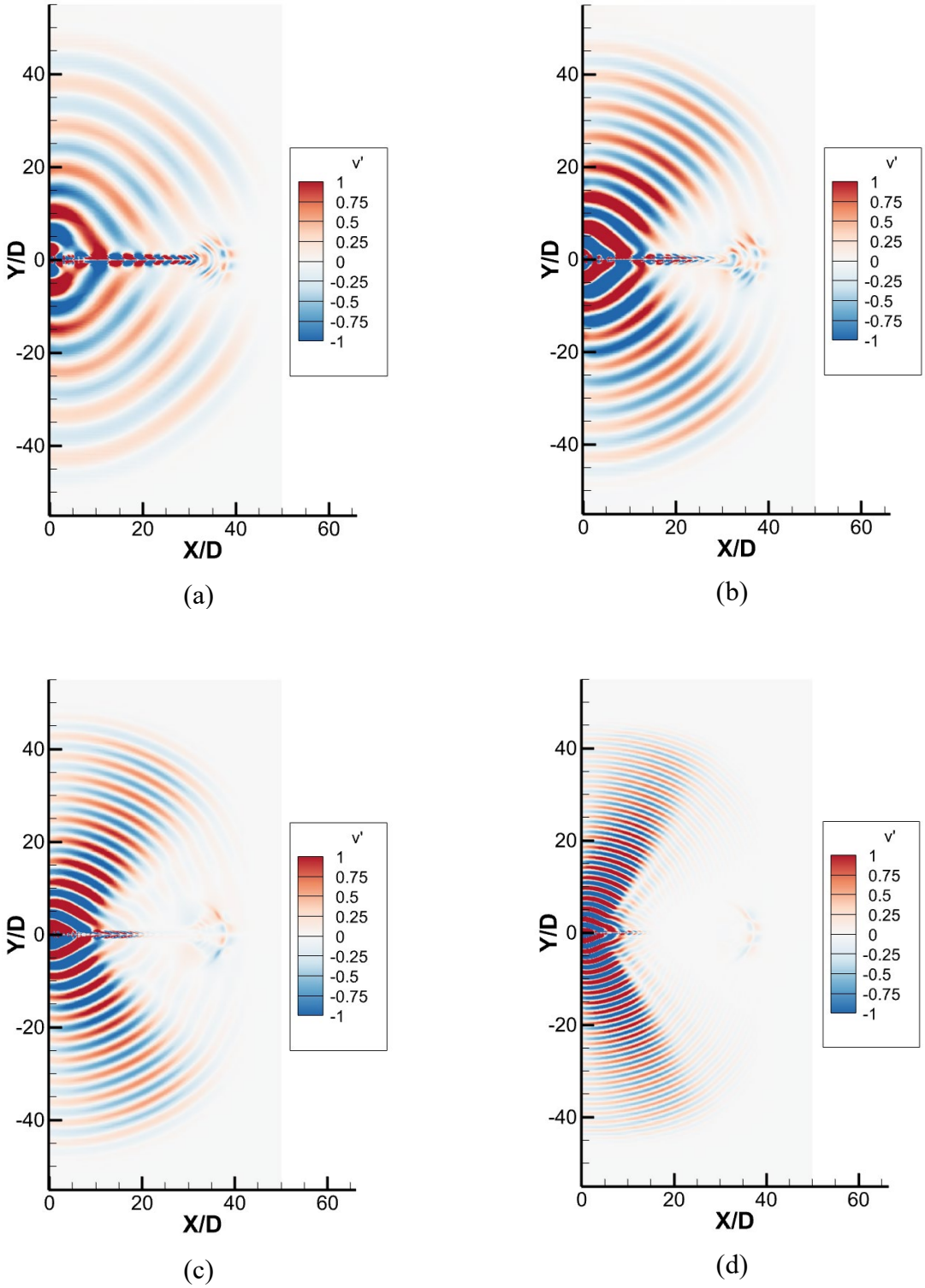


Figure 4.10 v' contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

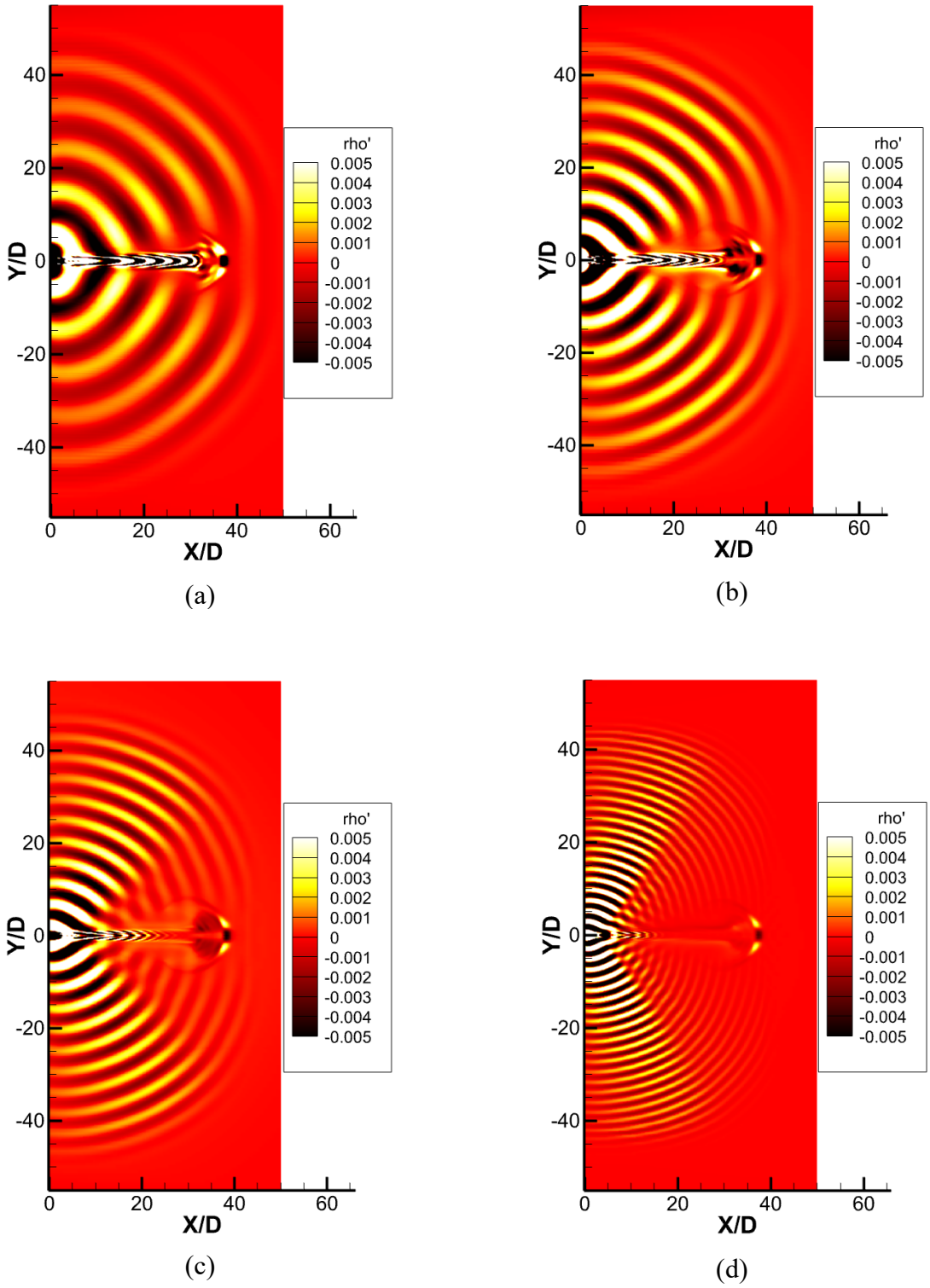


Figure 4.11 rho' contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

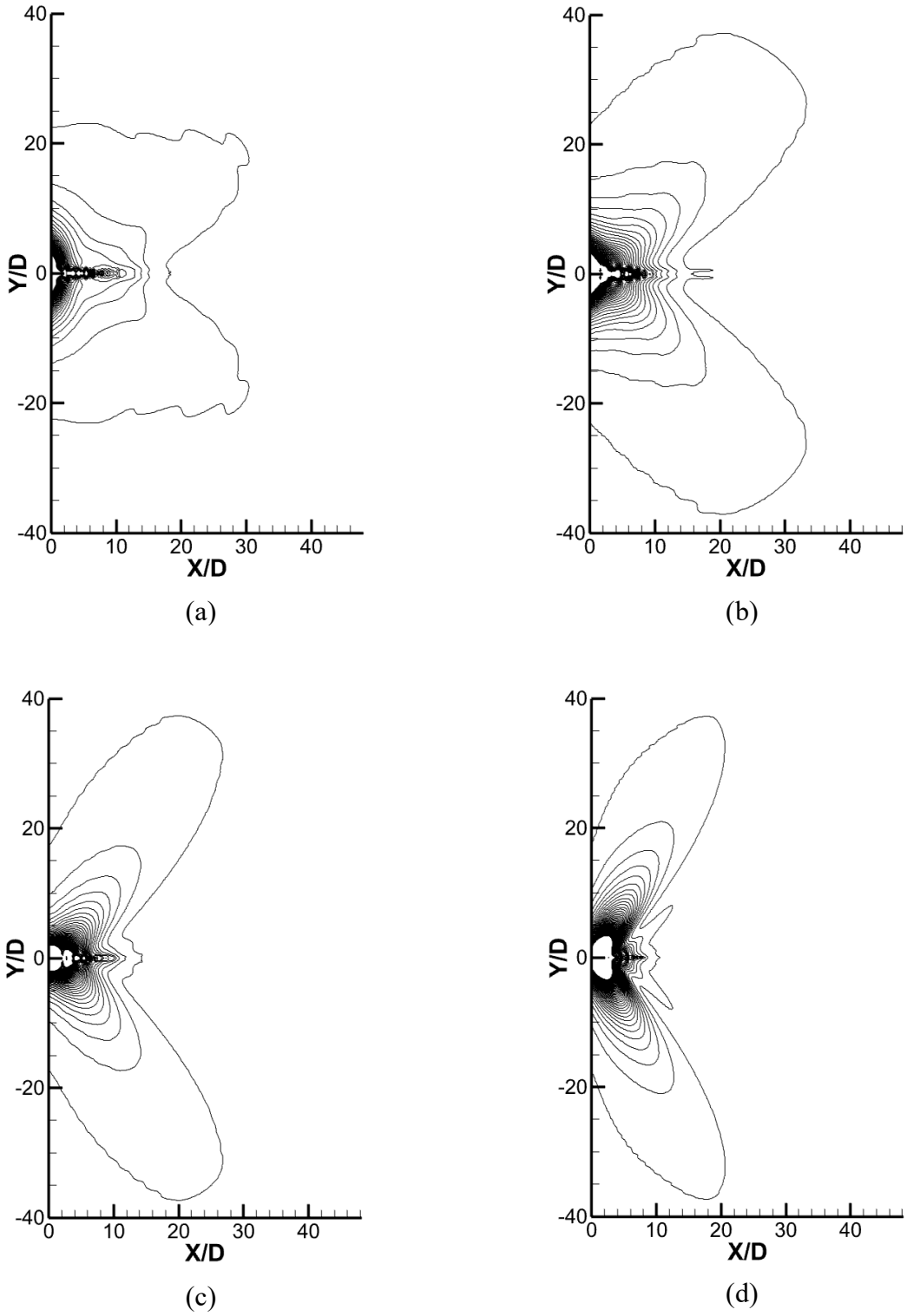


Figure 4.12 p’ rms contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

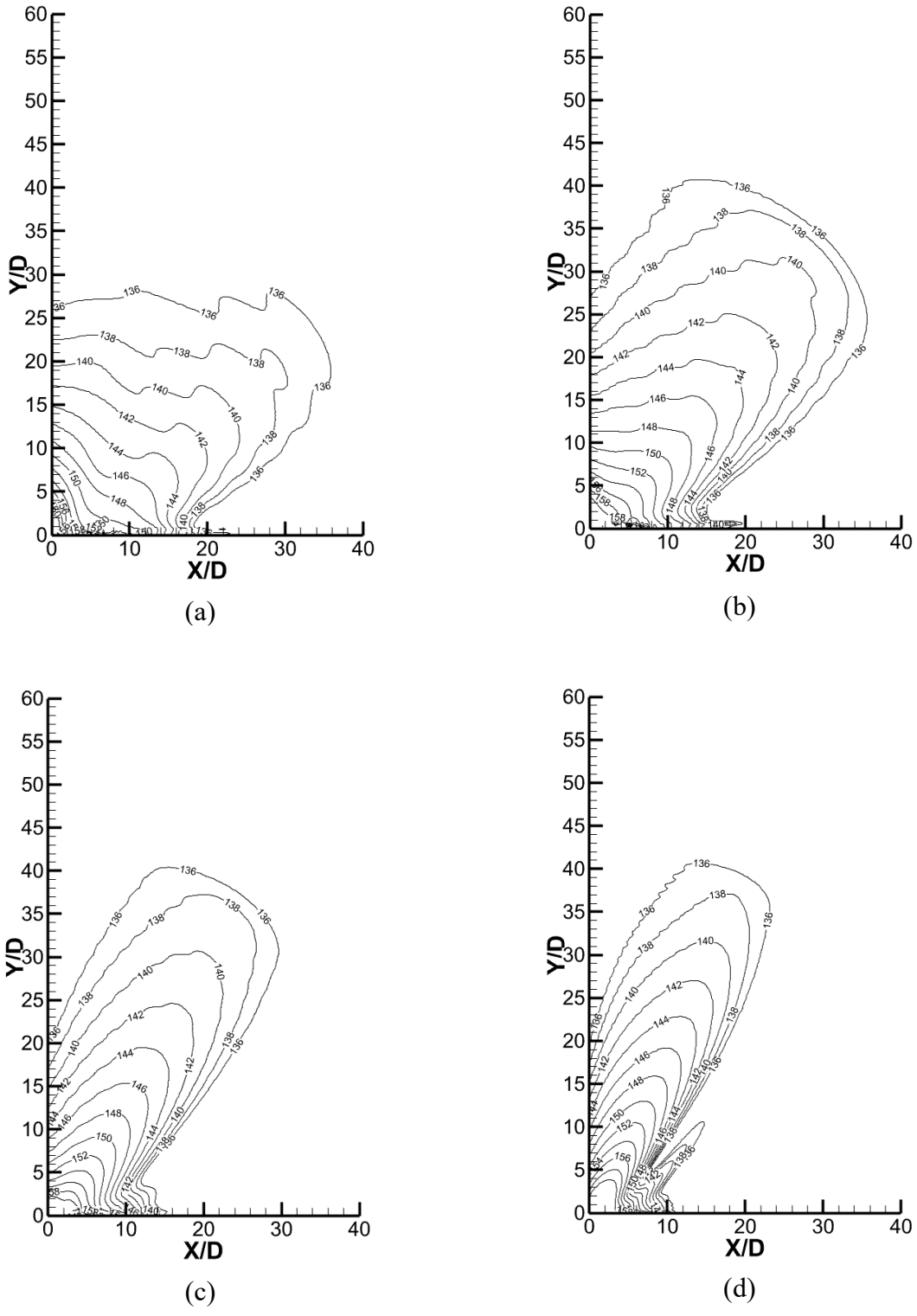


Figure 4.13 SPL contours minor plane (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

4.3 Directivity

This section presents Directivity and Iso-Surface plots for the rectangular jet case. As the case only simulates a single frequency excitation comparison of spectral data is not useful unlike LES. However, as can be seen the jet can be broken down and directivity can be seen with individual frequencies. See figure 4.14. The Iso-Surface at a particular SPL of interest can be generated as shown in figure 4.15.

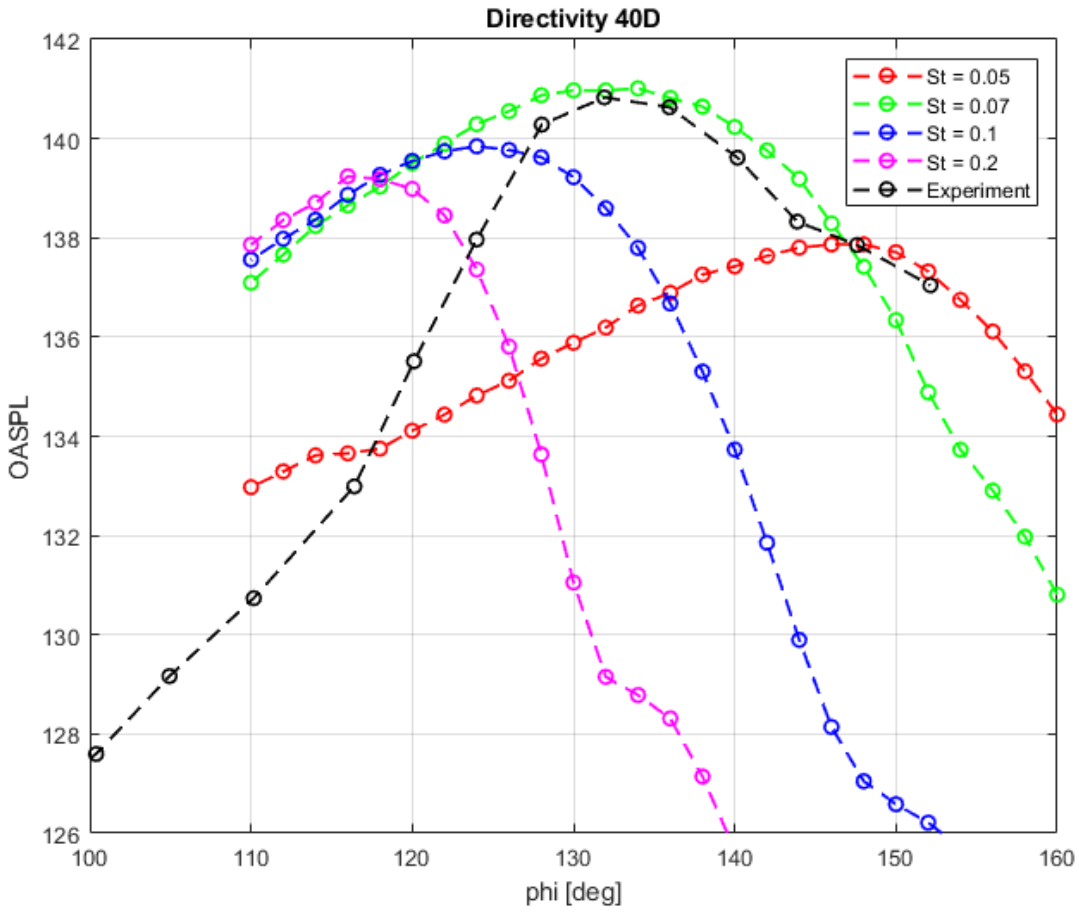
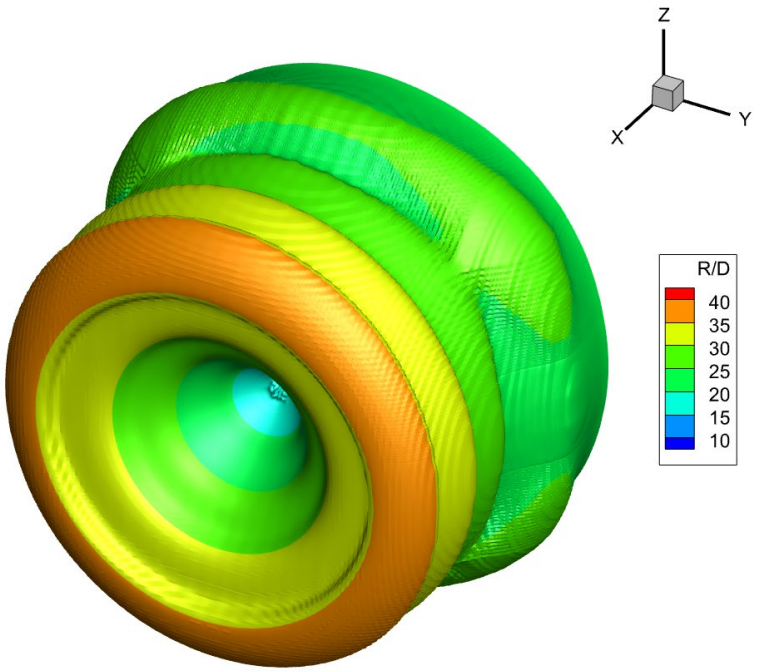
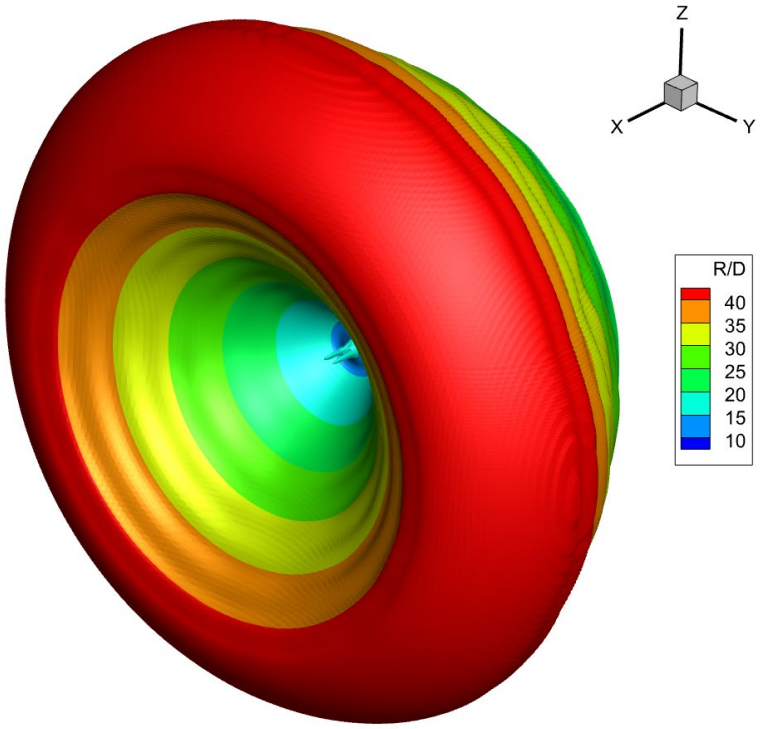


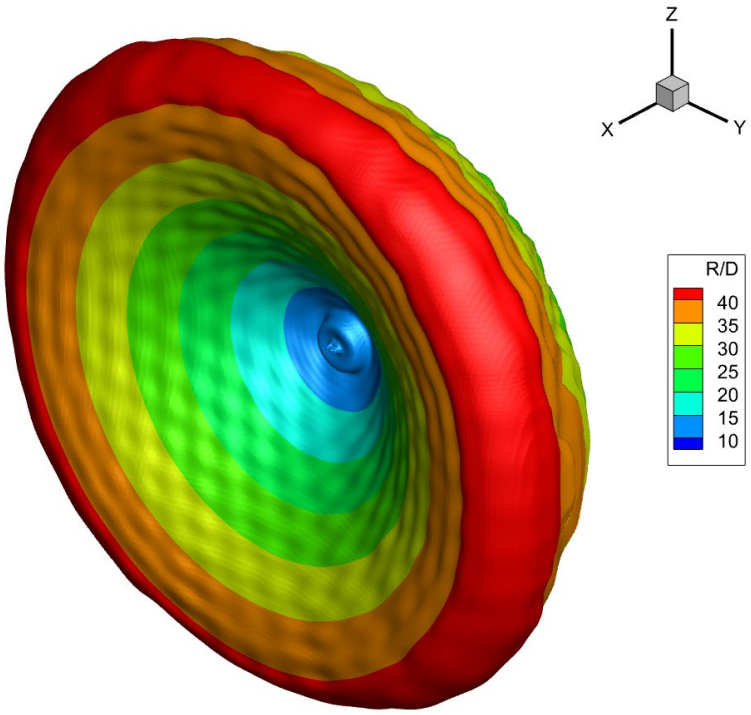
Figure 4.14 Directivity R = 40D



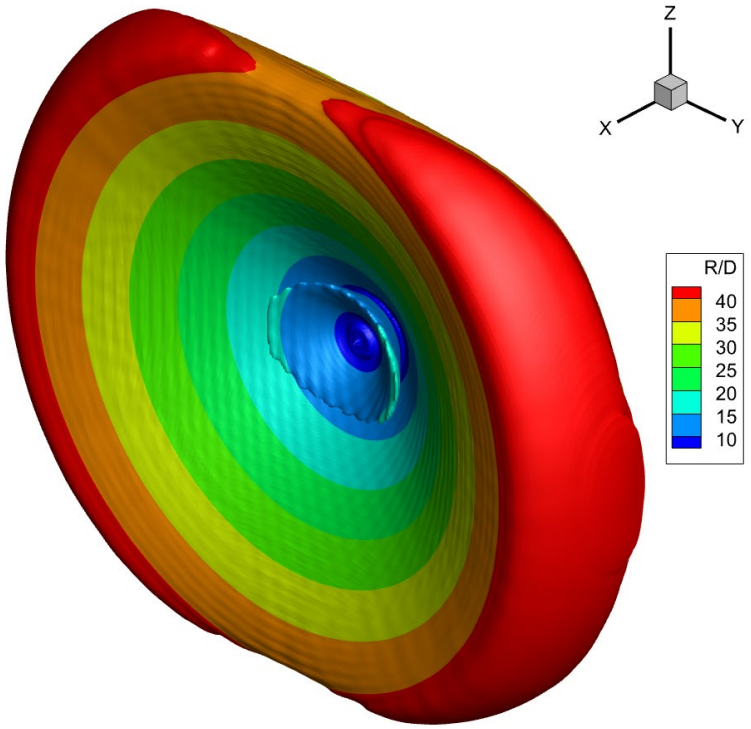
(a)



(b)



(c)



(d)

Figure 4.15 Directivity Iso Surfaces 138 dB (a) St = 0.05, (b) St = 0.07, (c) St = 0.1, (d) St = 0.2

4.4 Convergence

Like any good CFD practice convergence is extremely important in determining when the solution can be stopped. Convergence in simple terms is when the solution no longer changes with significant value. To test this on the rectangular jet points in the grid were sampled and the OASPL was calculated for each time step. As the solution progresses in time the OASPL at each sample point changes. When that change reached negligible value, the solution was deemed converged. See figure 4.16.

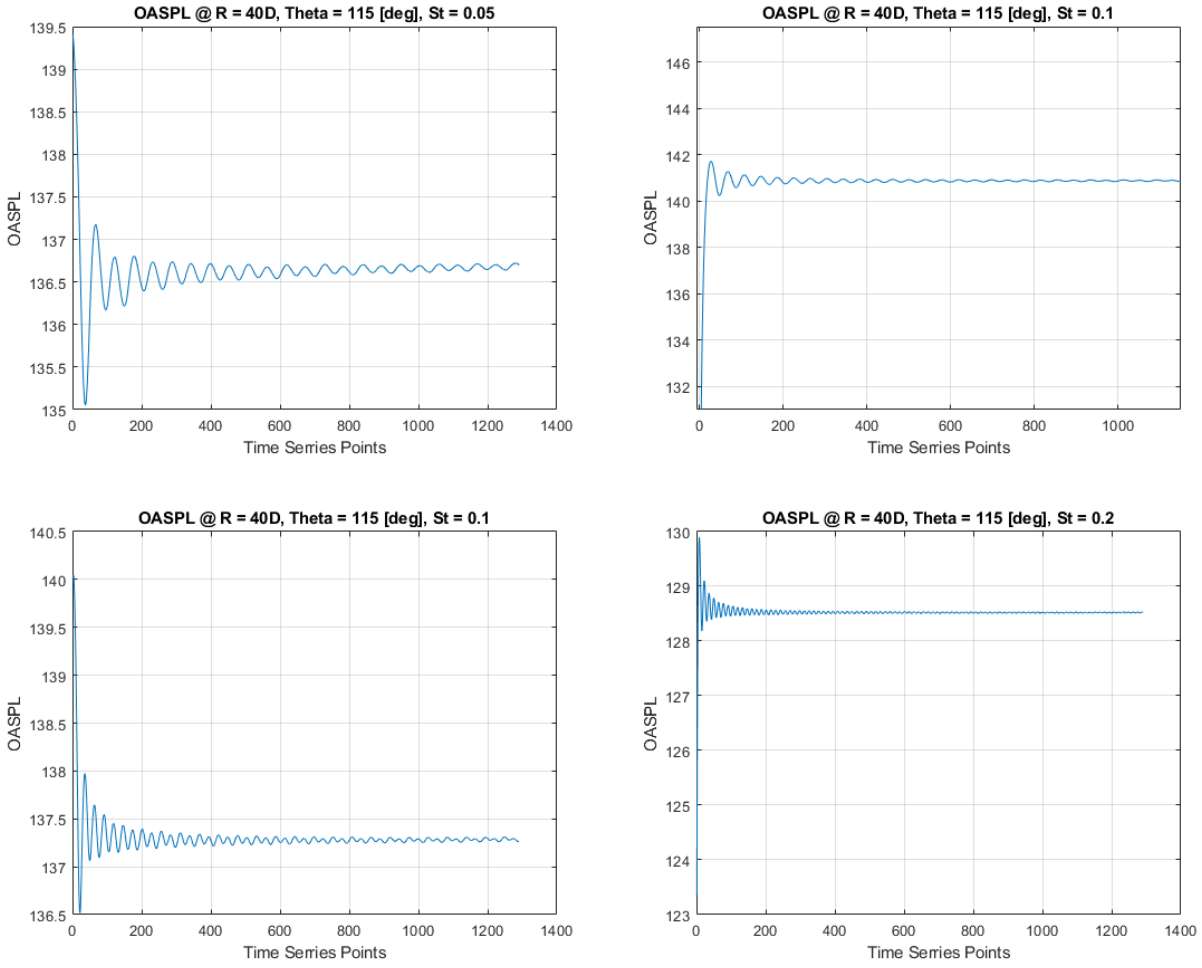


Figure 4.16 Convergence

4.5 FFT to Show Linear Propagation

As the leeFoam solver is a linear solver it is important to check that any inputs propagate linearly. This can be checked by performing an FFT inside the domain. By preforming the FFT as shown in figure 4.17 it can be seen the input frequency, being St = 0.1 in this example, remains the dominate frequency. It is also interesting to note the reduction of the high frequency noise. This shows the effectiveness of the ART term in dampening unwanted higher frequencies.

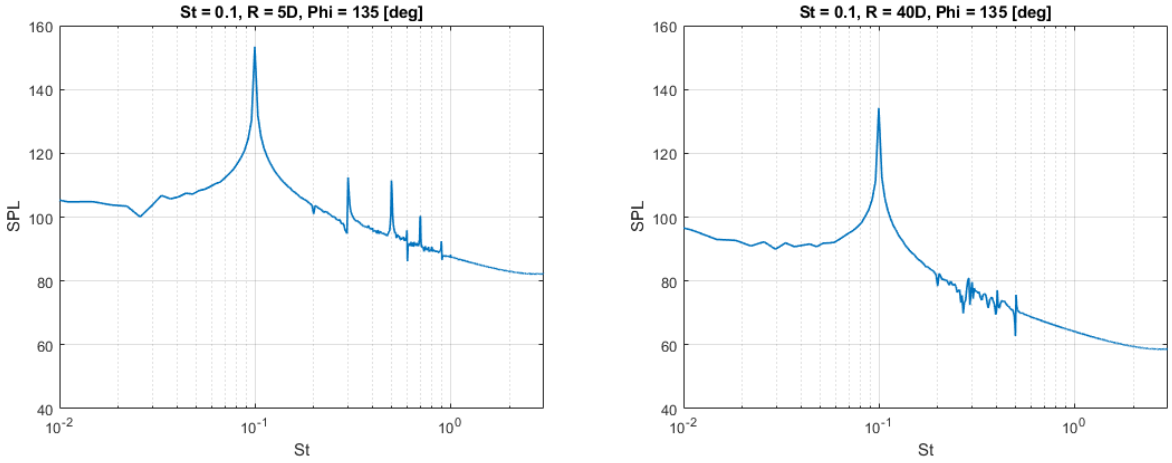


Figure 4.17 FFT to Show Linear Propagation

5 Conclusions and Recommendations

This chapter presents concluding arguments and recommendations from this work. This Thesis had two main objectives. Develop an LEE solver for OpenFOAM and utilize this solver on a rectangular supersonic jet. So, to conclude and to present recommendations for future work several questions need to be answered such as the following:

- Does the solver check out against analytical solutions and is it accurate?
- Can the solver maintain stability and how does it perform?

For the jet case, the questions challenge the solver based on other LEE codes run on supersonic jets and the experimental data presented. This chapter will present arguments to these questions in the following sections.

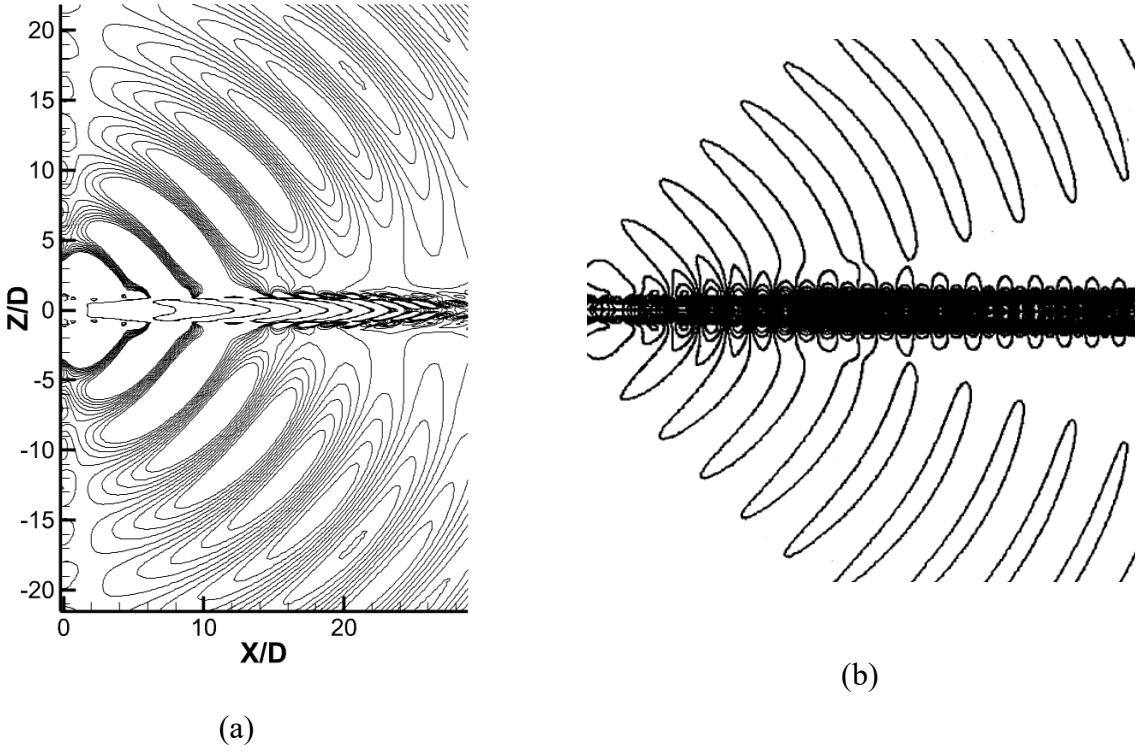
5.1 leeFoam Solver on Analytical Solutions

As shown in chapter 2 the leeFoam solver does compare to analytical solutions well. With the proper grid and numerical setup, the relative error is very low i.e., < 1%. This shows the solver can predict the acoustic solution accurately.

5.2 leeFoam Solver on the Rectangular Jet

For the supersonic rectangular jet, the case and the methods used in this work to solve the jet present numerous conclusions. To start, the main purpose of an LEE code is to produce an accurate acoustic solution quickly. The leeFoam solver achieves this goal as it does produce a solution quickly. The jet cases as mentioned were able to be completed in just a few hours on small amounts of resources of less than 200 CPUs. This is a vast improvement compared to LES. However, do the LEE results here hold any weight compared to LES? This is where we enter potentially problematic territory.

As mentioned, the major difference between LEE and LES is the physics being solved. LES solves for the flow physics and the acoustic solution is derived from this. LEE on the other hand only solves for the acoustic solution. LEE leaves out viscous and non-linear physics and these are present in supersonic jets. So, the question is, how much does the viscous and non-linear terms matter? Looking at the results, this is inconclusive at this time. The code does not match up with expected linear-stability results in the jet core. Looking at Figure 5.1 the leeFoam results are compared with results from the Mankbadi 1998 [8] work. It is clear there are some discrepancies in the jet core.



(a) (b)

Figure 5.1 Axial Velocity Disturbance Field leeFoam (a) Mankbadi 1998 (b)

One possible problem is the jet case tested here has shock formations as shown in the mean flow figures from chapter 3. The jet from the Mankbadi 1998 work was a round jet with no shock structure so it is a hard comparison to accurately make. The jet tested in this case is a perfectly

expanded jet and ideally there would be no shocks present. However, it is common in real world jets to still have a shock structure in the core due to slight imperfections. The OpenFOAM RANS results also agree with the experimental results, so this is a valid mean flow.

It is possible the shock structure is interfering with the results as the LEE's do not modal non-linear effects that are present in the given mean flow. This could also explain the stability issues faced in earlier simulations. So, further testing should be performed to rule this problem out.

What is conclusive however, is the solver produces an outstanding acoustic radiation field from the jet. This matches up with the results in chapter 2 and the acoustic field from the Mankbadi 1998 work [8].

The acoustic solution, however, is notably a lot simpler than LES results. The method presented in this work is a single mode and single frequency excitation. The real-world jet is not this simple as there are many frequencies and modes present. This is why spectra analysis comparison to experiment is not useful. However, the method used does offer a starting point for future work and it is possible to see how individual frequencies propagate through the domain. Other results that are beyond the scope of this thesis can be extracted from this approach such as the work from Malczewski, 2023 [21]. It is strongly encouraged to review that paper which is in the second publication from this thesis.

The final conclusion for the rectangular jet is on the numerical grid. The grid is arguably the most important aspect of any CFD project, especially for LEE. As mentioned before, the LEE's can become unstable as there are no viscous dissipation terms stabilizing any instabilities. Grid to grid oscillations is a real problem for the leeFoam solver even with artificial dissipation added. It is

therefore recommended; grids should be structured and with the least amount of non-orthogonality possible.

5.3 Overall Conclusions

Overall, even though there are some further tests that should be accomplished it appears the leeFoam solver can produce an acoustic solution like other LEE codes. The results show the solution is accurate, free from instabilities, and free of reflections from the boundary treatment. The fast nature of the LEE's and the OpenFOAM environment allow this code to be easily used in a CAA researchers workflow. The main benefit really in developing this code in OpenFOAM was found clear as the code can easily be run in parallel and with custom 3D grids. Other LEE codes are usually closed to the public but the leeFoam solver is open and is simple to use once a fundamental understanding of OpenFOAM is achieved.

5.4 Recommendations for Future Work

Recommendations for future work should revolve around continuing research on LEE and the supersonic jet. Further testing should be accomplished on how the code performs in the mean flow region of the jet. As mentioned before, it is hypothesized the issues revolve around the shock core structure but, this has yet to be proven. In reality if this is the problem then it is recommended the leeFoam solver needs to be coupled with an LES solver and be used strictly for the far-field propagation. This would be slower than pure LEE, however it is hypothesized greater overall accuracy and usefulness would be achieved. The benefit of this method would be a whole domain could be simulated and there would be no need to perform FW-H.

Another future consideration that would be less complicated and might shed some light on the jet core issues would be to run a random excitation on the inlet. The work from Mankbadi 1998 [8] showed the random input performed better than the single mode in predicting the directivity and this would be a good next step.

6 References

[1] Henderson, B. S., & Huff, D. L. (2021). A history of jet noise research at the National Aeronautics and Space Administration. *The Journal of the Acoustical Society of America, 150*(2), 1346-1369. doi:10.1121/10.0005891

[2] Dutchen, S. (n.d.). The effects of noise on health. Retrieved March 14, 2023, from https://hms.harvard.edu/magazine/viral-world/effects-noise-health#:~:text=They've%20shown%20that%20noise,attention%20deficits%3B%20childhood%20learning%20delays%3B

[3] Marques Goncalves, M., Salehian, S., Golubev, V. V., & Mankbadi, R. R. (2022). Active control of coherent structures on an axisymmetric jet. *AIAA SCITECH 2022 Forum*. https://doi.org/10.2514/6.2022-0123

[4] TAM, C. K., VISWANATHAN, K., AHUJA, K. K., & PANDA, J. (2008). The sources of jet noise: Experimental evidence. *Journal of Fluid Mechanics, 615*, 253-292. doi:10.1017/s0022112008003704

[5] Mora, P., Baier, F., Kailasanath, K., & Gutmark, E. J. (2016). Acoustics from a rectangular supersonic nozzle exhausting over a flat surface. *The Journal of the Acoustical Society of America, 140*(6), 4130-4141. doi:10.1121/1.4967158

[6] Cohen, H., Crichton, R. G. F., & H., S. H. I. (2008). *Gas turbine theory*. Pearson education.

[7] Callender, B., Gutmark, E., & Martens, S. (2005). Far-field acoustic investigation into Chevron nozzle mechanisms and trends. *AIAA Journal*, *43*(1), 87–95. https://doi.org/10.2514/1.6150

[8] Mankbadi, R., Hixon, R., Shih, S., & Povinelli, L. A. (1998). Use of linearized Euler equations for supersonic jet noise prediction. AIAA Journal, 36, 140-147. doi:10.2514/3.13792

[9] Mankbadi, R. R. (1994). Transition, turbulence, and noise theory and applications for scientists and Engineers. Boston u.a.: Kluwer Academic Publ.

[10] Lighthill, M. J. 1952 On sound generated aerodynamically: I. General theory, Proc. R. Soc. Lond. A 211, 564–587.

[11] Lighthill, M. J. 1954 On sound generated aerodynamically: II. Turbulence as a source of sound. Proc. R. Soc. Lond. A 222, 1–32.

[12] GOLDSTEIN, M. E. (2003). A generalized acoustic analogy. *Journal of Fluid Mechanics*, *488*, 315–333. https://doi.org/10.1017/s0022112003004890

[13] Greenshields, C. J., & Weller, H. G. (2022). *Notes on computational fluid dynamics general principles*. Reading, UK: CFD Direct Limited.

[14] Crank, J. and Nicolson, P., "A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type," *Advances in Computational Mathematics*, Vol. 6, Issue 1, 1996, pp. 207-226.

[15] Versteeg, H. K., and Malalasekera, W., An introduction to computational fluid dynamics: the finite volume method, 2nd ed. Pearson education, 2007.

[16] Fumiya, A. (2019, May 26). Fvoptions acousticdampingsource in openfoam: CFD with a mission. Retrieved March 14, 2023, from https://caefn.com/openfoam/fvoptions-acousticdampingsource

[17] Bernicke, P., Akkermans, R. A., Ewert, R., & Dierke, J. (2020). Acoustic relaxation term for damping and forcing of waves. *AIAA Journal, 58*(5), 2029-2041. doi:10.2514/1.j058236

[18] Tam, C. K., & Webb, J. C. (1993). Dispersion-relation-preserving finite difference schemes for computational acoustics. *Journal of Computational Physics, 107*(2), 262-281. doi:10.1006/jcph.1993.1142

[19] Salehian, S., & Mankbadi, R. (2020). Jet noise in airframe integration and shielding. *Applied Sciences, 10*(2), 511. doi:10.3390/app10020511

[20] Salehian, S., & Mankbadi, R. (2019). Numerical simulation of acoustic shielding effect on supersonic jets. *AIAA Propulsion and Energy 2019 Forum*. doi:10.2514/6.2019-3822

[21] Malczewski, B. J., Mankbadi, R. R., & Good, P. P. (2023). 3D nonlinear integral technique based on linearized Euler equations for the prediction of supersonic rectangular jet noise sources. *AIAA SCITECH 2023 Forum*. doi:10.2514/6.2023-0023

[22] Salehian, S., Good, P. P., Golubev, V. V., & Mankbadi, R. R. (2023). An OpenFoam-based Lee solver for prediction of noise generated by a supersonic jet issued from a rectangular nozzle. *AIAA SCITECH 2023 Forum*. doi:10.2514/6.2023-0613

[23] MANKBADI, R., HAYDER, M., & POVINELLI, L. (1993). The structure of supersonic jet flow and its radiated sound. *31st Aerospace Sciences Meeting*. doi:10.2514/6.1993-549

[24] Viswanath, Kamal, et al. "Flow Statistics and Noise of Ideally Expanded Supersonic Rectangular and Circular Jets." *AIAA Journal*, vol. 55, no. 10, 2017, pp. 3425–3439., https://doi.org/10.2514/1.j055717.

7 Publications

Salehian, S., Good, P. P., Golubev, V. V., & Mankbadi, R. R. (2023). An OpenFoam-based Lee solver for prediction of noise generated by a supersonic jet issued from a rectangular nozzle. *AIAA SCITECH 2023 Forum*. doi:10.2514/6.2023-0613

Malczewski, B. J., Mankbadi, R. R., & Good, P. P. (2023). 3D nonlinear integral technique based on linearized Euler equations for the prediction of supersonic rectangular jet noise sources. *AIAA SCITECH 2023 Forum*. doi:10.2514/6.2023-0023

Appendix A: Derivation of the Linearized Euler Equations

This section presents a derivation of the Linearized Euler Equations (LEE). There are many methods published of how to obtain the LEE's and this method is a combination of them. It is set to give the equations in a form that is easy to deploy in OpenFOAM. It is suggested to check out the derivations from the following works shown below as this method took influence from.

- Mankbadi, 1998
- Bogey et al., 2002
- Blom, 2003
- Tam, 2012
- Akhnoukh, 2015

Out of all of the differences between the researchers mentioned above they all derive the LEE's based on separating the mean and fluctuating components denoted by X_0 and X' respectively. See equation A.1, this will be used to derive the LEE's.

$$\rho = \rho_0 + \rho' \qquad \boldsymbol{U} = \boldsymbol{U_0} + \boldsymbol{U'} \qquad p = p_0 + p' \tag{A.1}$$

A.1 Continuity Equation

The first equation from the Euler equations which comes from neglecting viscous effects in the Navier-Stokes equations is the continuity equation as shown in equation A.2.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{U}) = S_c \tag{A.2}$$

A.2 then can be broken down using equation A.1 into the mean and fluctuating components. See equation A.3.

$$\frac{\partial \rho_0}{\partial t} + \frac{\partial \rho'}{\partial t} + \nabla \cdot (\rho' \boldsymbol{U}_0) + \nabla \cdot (\rho_0 \boldsymbol{U}') + \nabla \cdot (\rho_0 \boldsymbol{U}_0) + \nabla \cdot (\rho' \boldsymbol{U}') = S_c \qquad (A.3)$$

The mean flow is stagnant in time so pure mean flow terms can be removed. Also, the perturbations are deemed to be small so any pure perturbation terms can be removed as well. The final product is shown in equation A.4.

$$\frac{\partial \rho'}{\partial t} + \nabla \cdot (\rho' \boldsymbol{U}_0) + \nabla \cdot (\rho_0 \boldsymbol{U}') = S_c \qquad (A.4)$$

The next equation is the momentum equation as shown in equation A.5. The before procedure follows except the extra terms mean some cluttering so they are broken up into the time and space components for ease of the reader.

$$\rho \frac{d\boldsymbol{U}}{dt} + \rho(\boldsymbol{U} \cdot \nabla)\boldsymbol{U} + \nabla p = S_m \qquad (A.5)$$

$$\rho' \frac{d\boldsymbol{U_0}}{dt} + \rho_0 \frac{d\boldsymbol{U}'}{dt} + \rho' \frac{d\boldsymbol{U}'}{dt} + \rho_0 \frac{d\boldsymbol{U_0}}{dt} = S_{m,t} \qquad (A.6)$$

$$\rho_0 \frac{d\boldsymbol{U}'}{dt} = S_{m,t} \qquad (A.7)$$

$$\rho'^{(\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0} + \rho'^{(\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0} + \rho'^{(\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0} + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}' + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U} + \rho_0(\boldsymbol{U}\cdot\nabla)\boldsymbol{U}' + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0 + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0 + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0 = S_{m,s} \quad (A.8)$$

$$\rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}' + \rho'^{(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0} + \rho_0(\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0 + \nabla \mathrm{p}' = S_{m,s} \quad (A.9)$$

$$\rho_0\frac{d\boldsymbol{U}'}{dt} + \rho_0(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}' + \rho'^{(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0} + \rho_0(\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0 + \nabla \mathrm{p}' = S_m \quad (A.10)$$

After some manipulation,

$$\frac{d\boldsymbol{U}'}{dt} + (\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}' + \frac{\rho'}{\rho_0}(\boldsymbol{U}_0\cdot\nabla)\boldsymbol{U}_0 + (\boldsymbol{U}'\cdot\nabla)\boldsymbol{U}_0 + \frac{1}{\rho_0}\nabla \mathrm{p}' = S_m \quad (A.11)$$

Same procedure for the energy equation.

$$\frac{\partial(\rho E)}{\partial t} + \nabla\cdot(\rho E\boldsymbol{U}) = S_e \quad (A.12)$$

$$\rho = \frac{p}{RT} \quad e = C_v T \quad (A.13)$$

$$\frac{\partial}{\partial t}\left(\frac{p_0 + p'}{\gamma - 1}\right) + \frac{1}{2}(\rho_0 \boldsymbol{U}_0 \boldsymbol{U}_0 + \rho' \boldsymbol{U}_0 \boldsymbol{U}_0 + \rho_0 \boldsymbol{U}_0 \boldsymbol{U}' + \rho_0 \boldsymbol{U}' \boldsymbol{U}') = S_{e,t} \qquad (A.14)$$

$$\nabla\left(\frac{p_0 + p'}{\gamma - 1}\right) + \nabla\left(\frac{1}{2}(\rho_0 \boldsymbol{U}_0 \boldsymbol{U}_0 + \rho' \boldsymbol{U}_0 \boldsymbol{U}_0 + \rho_0 \boldsymbol{U}_0 \boldsymbol{U}' + \rho_0 \boldsymbol{U}' \boldsymbol{U}')\right) = S_{e,s} \qquad (A.15)$$

After some manipulation,

$$\frac{\partial p'}{\partial t} + \boldsymbol{u}' \cdot \nabla P + p' \gamma (\nabla \cdot \boldsymbol{U}) + \boldsymbol{U} \cdot \nabla p' + \gamma P (\nabla \cdot \boldsymbol{u}') = S_e \qquad (A.16)$$