

# **Aeroacoustic simulation of a linear cascade by a prefactored compact scheme**



Pietro Ghillani

Department of Engineering

University of Leicester

A thesis submitted for the degree of

*Doctor of Philosophy*

2012

# **Aeroacoustic simulation of a linear cascade by a prefactored compact scheme**

Pietro Ghillani

This work documents the development of a three-dimensional high-order prefactored compact finite-difference solver for computational aeroacoustics (CAA) based on the inviscid Euler equations. This time explicit scheme is applied to representative problems of sound generation by flow interacting with solid boundaries.

Four aeroacoustic problems are explored and the results validated against available reference analytical solution. Selected mesh convergence studies are conducted to determine the effective order of accuracy of the complete scheme. The first test case simulates the noise emitted by a still cylinder in an oscillating field. It provides a simple validation for the CAA-compatible solid wall condition used in the remainder of the work. The following test cases are increasingly complex versions of the turbomachinery rotor-stator interaction problem taken from NASA CAA workshops. In all the cases the results are compared against the available literature.

The numerical method features some appreciable contributions to computational aeroacoustics. A reduced data exchange technique for parallel computations is implemented, which requires the exchange of just two values for each boundary node, independently of the size of the zone overlap. A modified version of the non-reflecting buffer layer by Chen is used to allow aerodynamic perturbations at the through flow boundaries. The Giles subsonic boundary conditions are extended to three-dimensional curvilinear coordinates.

These advances have enabled to resolve the aerodynamic noise generation and near-field propagation on a representative cascade geometry with a time-marching scheme, with accuracy similar to spectral methods.

*To Lino (1920-2003), Paola (1922-2006), and Ugo (1922-2010)*

## **Acknowledgements**

I would like to express my sincere gratitude to Dr. Aldo Rona for his invaluable guidance and support throughout the course of this research work at the University of Leicester. I would also like to thank my colleague Mr. Ivan Spisso for providing me with the original version of the code and for having shared his knowledge along these years of study. I am also obliged to Dr. Emmanuil Georgoulis and Dr. Andrea Cangiani from the Department of Mathematics for their help every time I had a doubt on the most exquisitely mathematical aspects of my research. I cannot forget Dr. Marco Grottadaurea, Dr. Mohammed Fekry Farah El-Dosoky, Dr. David Adebayo, Mr. Davide Di Pasquale, Mr. Emanuele Monti and Mr. Massimiliano Canali for the many useful discussions on a range of scientific matters exceeding that of this study and for their sincere friendship. Many other fellow researchers at the University of Leicester from all over the world should be named here for teaching me their languages, their costumes and for sharing the happiness of their day to day life. Unfortunately, they are way too many to fit this page. Many thanks go to all the staff of the IT services, especially in the persons of Liam Gretton, Jon Wakelin, Chris Rudge and Gary Gilchrist for allowing me to perform all the computations of this work on the ALICE High Performance Computing Facility at the University of Leicester, and for being so responsive each time I had a

technical problem. Thanks also go to Tom Robotham for the maintenance of my PC and to Richard Heath for the provided software.

It is also important to acknowledge the support received in many forms from the following organizations: the AeroTraNet project for the scientific knowledge provided, the CASPUR (Rome) for the course on code optimisation, the CINECA (Bologna) for giving me the possibility to attend the summer school of parallel computing, the Free Software Foundation and all the people who created and made available free and open source quality software such as Linux, the gcc compiler, OpenMPI, L<sup>A</sup>T<sub>E</sub>X and many others which were used on a daily basis for this research: their work has made my work easier.

Finally and foremost, I have to thank my father Enrico and my mother Maria for having been so incredibly kind, respectful, and quiet during the long days of this study.

# Contents

<b>Contents</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Thesis layout . . . . .	1
1.2 Background and problem definition . . . . .	2
1.2.1 Background . . . . .	2
1.2.2 Problem definition . . . . .	3
1.3 Aims of this study . . . . .	4
1.4 Literature review . . . . .	4
1.4.1 Analytical solutions . . . . .	5
1.4.2 Numerical solutions . . . . .	6
<b>2 Numerical method</b>	<b>11</b>
2.1 Equations . . . . .	12
2.1.1 The Euler equations . . . . .	12
2.1.2 The characteristic form of the Euler equations . . . . .	17
2.1.3 Transformations between the different forms of the flow variables	21
2.1.4 Non-dimensionalisation . . . . .	23

2.2	Spatial discretisation . . . . .	24
2.2.1	Hixon's prefactored compact scheme . . . . .	26
2.2.2	Treatment of the boundaries . . . . .	29
2.2.3	High-order filters . . . . .	36
2.3	Time-marching scheme . . . . .	42
2.3.1	Low-storage Runge-Kutta schemes . . . . .	46
2.3.2	Optimised Runge-Kutta schemes . . . . .	49
2.4	Curvilinear coordinates . . . . .	52
2.5	Boundary conditions . . . . .	59
2.5.1	Kim and Lee characteristic generalised boundary conditions . . . . .	59
2.5.1.1	Inflow/outflow . . . . .	62
2.5.1.2	Inviscid wall . . . . .	66
2.5.1.3	Inter-block boundary . . . . .	67
2.5.2	Giles subsonic non-reflecting boundary conditions . . . . .	69
2.5.3	Symmetry . . . . .	79
2.5.4	Buffer zone . . . . .	82
2.6	Parallelisation . . . . .	83
2.6.1	Communication in reduced form . . . . .	89
2.7	Input/output . . . . .	92
2.8	Error norms . . . . .	94
<b>3</b>	<b>Test case applications</b>	<b>95</b>
3.1	First test case: dipole sound generated by an oscillating flow field around a still cylinder . . . . .	97
3.1.1	Problem definition . . . . .	97

---

3.1.2	Numerical methods . . . . .	102
3.1.3	Results . . . . .	106
3.2	Second test case: sound generation by interaction between an incident gust and a cascade of flat plates (2D) . . . . .	117
3.2.1	Problem definition . . . . .	117
3.2.2	Numerical methods . . . . .	120
3.2.3	Results . . . . .	129
3.2.3.1	Accuracy of the vortical gust . . . . .	130
3.2.3.2	Results from the complete model . . . . .	134
3.3	Third test case: sound generation by interaction between an incident gust and a cascade of flat plates (3D) . . . . .	161
3.3.1	Problem definition . . . . .	161
3.3.2	Numerical methods . . . . .	163
3.3.3	Results . . . . .	173
3.3.3.1	Accuracy of the vortical gust . . . . .	175
3.3.3.2	Results from the complete model . . . . .	183
3.4	Fourth test case: sound generation by interaction between an incident gust and a cascade of aerofoils . . . . .	203
3.4.1	Problem definition . . . . .	203
3.4.2	Numerical methods . . . . .	208
3.4.3	Results . . . . .	214
3.4.3.1	Steady flow through an aerofoil cascade without in-flow gust . . . . .	215
3.4.3.2	Unsteady flow through an aerofoil cascade with in-flow gust . . . . .	224

**CONTENTS**

**CONTENTS**

---

<b>4</b>	<b>Conclusions</b>	<b>243</b>
<b>5</b>	<b>Future work</b>	<b>247</b>
<b>A</b>	<b>Numerical results of the fourth test case</b>	<b>250</b>
	<b>Bibliography</b>	<b>253</b>

# Nomenclature

## Roman Symbols

$c$	Speed of sound
$c_p$	Specific heat at constant pressure
$c_v$	Specific heat at constant volume
$e$	Specific internal energy
$e_t$	Specific total internal energy
$\mathbf{f}$	Vector of the body forces
$f_x$	Component of the body forces along the $x$ axis
$f_y$	Component of the body forces along the $y$ axis
$f_z$	Component of the body forces along the $z$ axis
$i$	Imaginary unit
$I$	Identity matrix
$\mathbf{l}$	Left eigenvector
$p$	Static pressure
$p_{tot}$	Stagnation pressure
$\mathbf{Q}$	Vector of the primitive variables
$\mathbf{r}$	Right eigenvector
$\mathbf{R}$	Vector of the characteristic variables
$R$	Specific gas constant

## NOMENCLATURE

---

$t$	Time
$T$	Absolute temperature
$T_{tot}$	Stagnation temperature
$\mathbf{u}$	Velocity vector
$\mathbf{U}$	Vector of the conservative variables
$u$	Velocity component along the $x$ axis
$v$	Velocity component along the $y$ axis
$w$	Velocity component along the $z$ axis

### Greek Symbols

$\lambda$	Eigenvalue
$\omega$	Angular frequency
$\rho$	Density

### Superscripts

$T$	Transpose operator
-----	--------------------

### Acronyms

CAA	Computational Aeroacoustics
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy (condition)
CGNS	CFD General Notation System
HOOS	High-Order One-Sided
HPC	High Performance Computing
LOC	Low-Order Centred
LODI	Local One-Dimensional Inviscid
MPI	Message Passing Interface
ODE	Ordinary Differential Equation

## **NOMENCLATURE**

---

PML Perfectly Matched Layer

SIDS Standard Interface Data Structures

# Chapter 1

## Introduction

### 1.1 Thesis layout

This thesis is divided into five chapters. The first chapter contains an introduction on the object of this research including a brief description of the problem, its background and a literature review. In addition, the aims of this work and its expected outcomes are detailed.

The second chapter presents the numerical methods that have been implemented in the flow solver to address a series of test cases of incremental complexity. Particular attention was given to the implementation of the optimised explicit time-integration scheme, the generalised characteristic-based boundary conditions, the code parallelisation and the data input/output in standardised form.

The third chapter presents the results obtained by the solution of four increasingly complex test cases related to the turbomachinery rotor-stator interaction problem.

Chapter four, the conclusions, contains a summary of the work done that focuses especially on the newly devised numerical methods and on the impact that these ad-

## 1. INTRODUCTION

---

vances have in the field of computational aeroacoustics.

To conclude, in the fifth chapter, the current limitations of the solver are discussed. Approaches for their alleviation are proposed in future work.

## 1.2 Background and problem definition

### 1.2.1 Background

In the last decades, the growing popularity of the commercial civil air transport has led to more stringent legislation regarding noise-pollution in the areas surrounding the airports and to an increased environmental awareness [UK Department of Transport, 2003; Various, 2005]. Therefore, research, both theoretical and applied, in the field of aeroacoustics, primarily related to the various components of the noise emitted by aircraft, has received a strong impulse. Due to the complexity of the involved geometries and the inherent unsteady nature of the aerodynamic sound generation, the theoretical acoustic analysis and the traditional techniques of Computational Fluid Dynamics (CFD) are of limited applicability [Tam, 2004; Tam & Hardin, 1997]. For these reasons, in the last 30 years a branch of the field of computational fluid mechanics, called Computational Aeroacoustics (CAA), has emerged [Tam, 1995]. Starting directly from the fundamental flow governing equations, the CAA aims to reproduce all the aspects of the sound generation and propagation in air by overcoming various problems of physical and numerical nature including the disparity in magnitude between mean and acoustic flow variables and the necessity to model the high-frequency noise components. Among the many numerical methods developed in CAA, of particular relevance are the high-order finite-difference schemes which have remarkable low dispersion and

## 1. INTRODUCTION

---

dissipation properties that are required to increase the accuracy and the efficiency of the computation by reducing the number of grid points per wavelength, while still ensuring tolerable levels of numerical error [Colonius & Lele, 2004; Rona & Spisso, 2007].

### 1.2.2 Problem definition

As mentioned, the aircraft contain many sources of noise. Among the most relevant is the noise emitted by the turbofan engines, often referred to as *turbomachinery noise*. Its various components, originating from different parts of the engine, have relative importance depending of the flight configuration (landing or take-off), but the two main sources are usually the fan and the turbine [Groeneweg *et al.*, 1995]. In this case, the physical process of the noise generation includes a number of interacting factors like the blades unsteady response to internal and external disturbances and the influence of the duct. Due to the practical problems involved in the collection of data from the internal parts of the engine, experimental studies of these complex interactions is often impossible. For this reason, the development of analytical and numerical methods able to model the physical processes at the basis of the noise generation are of high importance. The interaction between incident vortical disturbances having complex spectral content and a cascade of static or rotating blades is one of these fundamental processes, and is the subject of this research. Noise is originated by the conversion of the hydrodynamic kinetic energy of the disturbances into pressure fluctuations pressure over the blade surface that is followed by the emission of acoustic waves characterised by a specific spectrum and directivity. Complex analytical models have been developed over the years to accurately describe the influence of the various parameters defining the problem, like the number of blades, their shape and relative position. These mod-

## 1. INTRODUCTION

---

els have helped to clarify various aspects of the involved phenomena but are always approximations of the actual complex intertwined physical processes. The numerical methods provided by CAA have been developed to overcome these limitations.

This study focuses on the particular case of vortical gusts with a limited spectral content impinging on a stator blade cascade. Due to its complexity, the problem described is transposed into a series of four test cases in which the geometry of the blades cascade is modified in order to include different aspects of the actual turbomachine configuration.

### 1.3 Aims of this study

The principal aim of this work is the solution of the turbomachinery-related problem stated in Section 1.2.2. For this purpose, a three-dimensional high-order prefactored compact finite-difference solver for CAA, based on the inviscid Euler equations with explicit time-integration, was developed. However, the code has general applicability, and the larger aim of this study is to demonstrate its ability to address real-world problems involving the conversion of hydrodynamic kinetic energy into sound by the interaction with rigid surfaces.

### 1.4 Literature review

In the following, a literature review of the most relevant analytical and numerical formulations regarding the problem of this work is presented.

For the various numerical methods implemented in the solver, the essential literature references are given in the specific sections of Chapter 2. An exception is made

## 1. INTRODUCTION

---

for Section 2.3 in which the theory is preceded by an extensive literature review of the optimised explicit Runge-Kutta methods.

For the relevant literature regarding the basic characteristics of the original version of the in-house finite-difference prefactored compact solver, refer to the work of Spisso [unpublished].

### 1.4.1 Analytical solutions

From an analytical point of view, the first solution to the unsteady subsonic problem of a vortical gust impinging on an infinite cascade of flat plates, based on vortex sound theory, was given by Smith [1972]. The geometry was limited to two dimensions and the mean flow was aligned with the plates surfaces, no stagger between successive blades was considered. The author supplied the predicted pressure distribution both in the upstream and the downstream direction along with the unsteady blade lift and moment, and compared the results with experimental data. Based on the linear theory of Smith, Whitehead [1987] developed a computer program called LINSUB for the numerical evaluation of the pressure field in which approximations for the computation of the infinite sums and of the numerical integration were introduced. This program was later modified by Hall [1997a] to obtain one of the numerical reference solutions considered in this work. A different solution to the problem solved by Smith, but limited to the transmission and reflection of the acoustic waves, was proposed by Koch [1971] who started from the treatment given by Mani & Horvay [1970], based on the finite Wiener-Hopf technique. This exact analysis was later extended by Peake [1993] to cover the case of a vortical incoming gust.

The three-dimensional version of the problem was first solved in an approximate

## 1. INTRODUCTION

---

form by Atassi & Hamad [1981] and then by Envia & Kerschen [1986] who obtained the solution for a swept vane cascade from the superimposition of the solution obtained for an isolated stator vane. The neglected effects due to the interaction between the blades and their finite chord were introduced in the treatment of Glegg [1999] that, similarly to Koch [1971], based his solution on the Wiener-Hopf method. The work by Glegg illustrates the response of the blades cascade to an incoming acoustical or vortical three-dimensional disturbance by supplying the analytical solution for the unsteady loading of the blades and the acoustic power of each propagating mode.

In the case of a swept cascade bounded by two parallel walls, no analytical solution is known. The treatment given by Envia [2000], based on a previous work by the same author [Envia, 1988] allows to obtain an approximate solution to the problem by considering only the contribution from the leading edge of the infinitely thin blades while discarding the contribution of the trailing edge and the mutual influence of the blades. In this work the solution provided by Envia is used as a reference to assess the accuracy of one of the solved test cases.

Other relevant contributions to this problem have been made by Majumdar & Peake [1996], Hanson & Horan [1998], Evers & Peake [2002], Cheong *et al.* [2006, 2009] and Wei & Cheong [2010].

### 1.4.2 Numerical solutions

Numerical solutions to the various versions of the gust-cascade interaction problem have become available in literature since the problems were proposed in the respective NASA CAA workshops [Dahl, 2000, 2004; Tam & Hardin, 1997].

The two-dimensional version of the problem of a vortical gust impinging into a rec-

## 1. INTRODUCTION

---

tilinear cascade of thin flat plates was first solved by four research groups whose contributions are included in the proceedings of the second NASA CAA workshop [Tam & Hardin, 1997]. The solution proposed by Hu & Manthey [1997] employed the Perfectly Matched Layer (PML) technique for the enforcement of the boundary closures at the inflow and at the outflow, and of the optimised LDDRK explicit Runge-Kutta scheme [Hu *et al.*, 1996] for the time-integration. The PML technique, originally applied to Maxwell's equations, is, in this case, adapted to the inviscid Euler equations. Another contribution was provided by Tam *et al.* [1997] whose results were based on the application of the explicit spatial discretisation given by the Dispersion-Relation-Preserving (DRP) of Tam & Webb [1993]. The most interesting aspects of the adopted numerical methods are, again, the external boundary closures that in this case were tailor-made for the absorption of the analytically predicted duct modes. Hall [1997b] adopted for the solution of the problem an original approach by employing a finite-element formulation based on a variational method, with results. The last contribution from the second NASA CAA workshop was given by Lockard & Morris [1997] whose solution is characterised by the use of a structured irregular mesh, the optimised RK56 two-step explicit Runge-Kutta time-integration scheme by Hu *et al.* [1996], and the adoption of the DRP scheme by Tam & Webb [1993] with the modified coefficients given by Lockard *et al.* [1995]. At the inflow and outflow the Giles subsonic boundary conditions were applied. Further contributions to this problem were given during the following NASA workshops with the solutions given by Wang *et al.* [2000] and by Bin *et al.* [2004] that used this problem as an introductory test for their codes. The former adopted an original technique, also applied to the three-dimensional form of the problem, called space-time conservation element and solution element (CE/SE), based on a second-order finite volume formulation that allows to specify the non-reflecting

## 1. INTRODUCTION

---

boundary conditions at the inflow and at the outflow in a particularly simple form, without resorting to the usual characteristic-based closures. The latter proposed a solution based on a particular version of the DRP schemes, devised by one of the authors [Cheong & Lee, 2001], called Grid-Optimised Dispersion-Relations-Preserving (GODRP). This spatial scheme was coupled to a third-order Adams-Bashforth time-integration scheme. At the inflow and at the outflow the boundary conditions by Kim & Lee [2000], also used in this work, were enforced.

Other solutions to this problem were published by Hixon *et al.* [2000] and by Ragab & Salem-Said [2007]. The study by Hixon *et al.* compares the performances of four different pairs of inflow/outflow non-reflecting boundary conditions: the characteristic-based 1D by Thompson [1987, 1990], the subsonic non-reflecting by Giles [1990] (modified for the input of the vortical gust from the inflow boundary), the formulation proposed by Hagstrom & Goodrich [2003], and the PML implementation of Abarbanel *et al.* [1999]. The results show that the PML technique is the most accurate but has the disadvantage of requiring additional computational nodes. The study by Ragab and Salem-Said contains the solution to the original problem as defined by the NASA CAA workshop [Tam & Hardin, 1997] and a number of its variations that make their work the most extensive in literature on this subject. The spatial scheme was a sixth-order compact by Lele [1992] with filtering following Visbal & Gaitonde [2002] and time-integration was performed by a five-stage optimised explicit Runge-Kutta scheme by Carpenter & Kennedy [1994]. The inviscid wall was modelled following Poinso & Lele [1992] and, at the outflow, the closure was provided by the Giles [1990] boundary conditions. At the inflow, Giles condition was modified to allow the introduction of the vortical gust. The article by Ragab & Salem-Said [2007] also introduces corrections in the scheme to deal with the singularities of the pressure distribution. The authors

## 1. INTRODUCTION

---

further extended their work on this subject in Salem-Said & Ragab [2008].

The three-dimensional version of the problem with the swept cascade of flat plates, defined by Dahl [2000], was solved by Wang *et al.* [2000] adopting the same numerical techniques employed for the solution of the 2D problem already described in this review. It has to be noted that the supplied solution does not satisfy all the requirements defined by the problem stated in Dahl [2000] and it is limited to sweep angles lower than  $15^\circ$ . No other numerical solution of this problem is known.

In the occasion of the fourth NASA CAA workshop, the two-dimensional version of the problem including a blades cascade with realistic profile [Dahl, 2004], was solved by six research groups. Bin *et al.* [2004] used numerical techniques here already described for the version of the problem with infinitely thin plates. The solution given by Nallasamy *et al.* [2004] was obtained with a parallel [Hixon *et al.*, 2002] non-linear solver based on the spatial scheme by Hixon [2000a], and with the time-integration performed by an explicit optimised Runge-Kutta scheme of Hu *et al.* [1996]. The spurious oscillations were removed applying the tenth-order dissipation model by Kennedy & Carpenter [1994] and, at the inflow and outflow, a modified version of the non-reflecting conditions by Giles [1990] were used. In this case, the boundary closure has not only the ability to feed the vortical gust into the computational field, but also that of maintaining a constant mean condition specified by the problem, both at the inflow and at the outflow [Hixon *et al.*, 2003, 2004]. Wang *et al.* [2004] solved the problem using the CE/SE method already described in this review for the versions of the problem previously discussed [Wang *et al.*, 2000]. A highly original approach to the problem was illustrated by Hixon [2004]. His newly developed Space-Time Mapping Analysis (STMA) technique, particularly suited for the parallel computation on multiple processors, replaces the time-marching method with an itera-

## 1. INTRODUCTION

---

tive procedure based on a “space-time” volume that avoids the need of synchronisation between different processes [Hixon, 2003]. During the fourth NASA CAA workshop, the last two contributions came from Escribano *et al.* [2004] and Coupland [2004] that, for the solution of the problem, adopted frequency-domain type of solvers. The former, using unstructured grids, performed a two-step analysis first by computing the mean steady-state flow by the means of a non-linear code called  $Mu^2s^2T$ , and then by separately solving the unsteady problem, with a linear version of the same code called  $Mu^2s^2T-L$ , for each of the three different harmonics of the incoming vortical gust. At the through boundaries the Giles [1990] conditions were used. The latter, employing a structured multi-block non-linear code called HYDRA, performed the integration by an unspecified five-stage Runge-Kutta scheme with multi-grid acceleration. In this case the treatment of the boundaries was performed in a problem-specific fashion based on the analytical prediction of the propagation modes.

The group of research of the University of Toledo, Ohio, USA is actively working on problems [Nallasamy *et al.*, 2007] that are direct three-dimensional extensions with blades of increasingly realistic shape, of the series of problems here illustrated. In particular, their most recent publications [Hixon *et al.*, 2010, 2011] regard the modelisation of realistic three-dimensional rotor wakes with a technique called Vortical Gust Boundary Condition (VGBC), and the extension of the subsonic boundary condition by Giles [1990] to unsteady three-dimensional flows.

## Chapter 2

### Numerical method

The original in-house code by Spisso [unpublished] from which this work started, was based on the spatial compact prefactored finite-difference scheme by Hixon [2000b] and was capable to integrate solutions in time by the use of the classical four-stage fourth-order Runge-Kutta method and of the more recent two-stage low-dispersion and low-dissipation Runge-Kutta scheme by Hu *et al.* [1996]. The applicability of this solver was restricted to 2D problems in primitive linear form defined on regular Cartesian meshes. The range of available boundary conditions included the inviscid wall by Tam & Dong [1994] and the buffer layer by Chen *et al.* [2004]. The applicable numerical high-order filtering techniques of Gaitonde & Visbal [1998, 1999] were restricted to their explicit formulation and to their implicit prefactored compact formulation by Hixon [1999]. The code did not allow to run parallel computations and could not manage the non-dimensionalisation of the variables and the input and output of data in a standard format.

During the course of this work the 2D Cartesian code was extended to 3D curvilinear coordinates to solve the flow governing equations in either primitive, conservative,

## 2. NUMERICAL METHOD

---

linearised or non-linear form. The equations can also be non-dimensionalised with respect to a prescribed set of reference values. In addition, the data input-output from standard files was implemented to improve the portability of the initial models and of the results. The range of solvable problems was extended with the introduction of a number of characteristic-based boundary treatments and with the possibility to run simulations on large models thanks to the domain decomposition technique and to the parallelisation of the code. Finally, the most recent optimised explicit Runge-Kutta time-marching schemes were implemented to speed-up the temporal integration.

In this chapter, the algorithms used in the solver are described in detail with a particular emphasis on the features newly implemented.

### 2.1 Equations

#### 2.1.1 The Euler equations

The inviscid Euler equations are derived by the application of the principles of conservation of mass, momentum, and energy to an arbitrary volume of inviscid fluid. The application of these principles to an arbitrary closed region of fluid, called *control volume*, bounded by a *control surface* that can either be fixed in the space (Eulerian frame of reference) or moving with the fluid (Lagrangian frame of reference) leads to different formulations of the Euler equations. If the control volume has a finite extension, the *integral* formulation is obtained. Else, if the control volume is infinitely small so to have homogeneous flow properties within it, the *differential* formulation is derived. When the Eulerian frame of reference is used, the equations are said to be in *conservative* form and the unknown variables are defined by vector  $\mathbf{U} = [\rho, \rho\mathbf{u}, \rho e_t]^T$ . Their

## 2. NUMERICAL METHOD

---

differential formulation is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.1a)$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \rho \mathbf{f}, \quad (2.1b)$$

$$\frac{\partial (\rho e_t)}{\partial t} + \nabla \cdot (\rho e_t \mathbf{u}) = -\nabla \cdot (p \mathbf{u}) + \rho \mathbf{f} \cdot \mathbf{u}, \quad (2.1c)$$

where  $\rho$  is the density,  $p$  is the static pressure,  $\mathbf{u} = [u, v, w]$  is the vector of the velocity components in three-dimensional Cartesian form,  $e_t$  is the specific total energy that is given by the sum of the specific internal energy  $e$  and the kinetic energy  $\mathbf{u} \cdot \mathbf{u}/2$ ,  $\mathbf{f} = [f_x, f_y, f_z]$  is the body force vector per unit mass that could be caused by the presence of a gravitational or an electromagnetic field, and  $t$  is the time. Equations (2.1) are applicable to homogeneous, non-reacting, mass-conserving flows.

If the Lagrangian approach is used, the equations are in *primitive* form and the unknown variables are defined by vector  $\mathbf{Q} = [\rho, \mathbf{u}, p]^T$ , in which  $(\rho, p)$  are a selected pair of thermodynamic variables that define the thermodynamic fluid state. Their complete differential formulation is

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0, \quad (2.2a)$$

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho} \nabla p + \mathbf{f}, \quad (2.2b)$$

$$\frac{Dp}{Dt} = -\gamma p \nabla \cdot \mathbf{u} + (\gamma - 1) \rho \mathbf{f} \cdot \mathbf{u}, \quad (2.2c)$$

where  $\gamma$  is the ratio of the specific heat at constant pressure  $c_p$  to that at constant volume  $c_v$ , with  $\frac{D}{Dt}$  denoting the material derivative.

The Euler equations in their 3D form have 6 unknown variables; therefore, to solve

## 2. NUMERICAL METHOD

---

the system, an additional relation is required. Usually, when dealing with gases, the closure is provided by the *equation of state* for a perfect gas

$$p = \rho RT, \quad (2.3)$$

that relates the static pressure  $p$  with the density  $\rho$  and the absolute temperature  $T$  through the specific gas constant  $R$ . This equation introduces one further variable  $T$  that is directly related to the specific internal energy  $e$  by the expression

$$e = c_v T. \quad (2.4)$$

In order to numerically compute the Euler equations, they can be more conveniently recast in *strong* or *vector* conservative form

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{G}}{\partial z} = \mathbf{S}, \quad (2.5)$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e_t \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ \rho f_x \\ \rho f_y \\ \rho f_z \\ \rho \mathbf{f} \cdot \mathbf{u} \end{bmatrix}, \quad (2.6)$$

## 2. NUMERICAL METHOD

---

$$\mathbf{E} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho w u \\ \rho e_t u + pu \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho w v \\ \rho e_t v + pv \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \rho w \\ \rho u w \\ \rho v w \\ \rho w^2 + p \\ \rho e_t w + pw \end{bmatrix}. \quad (2.7)$$

The same system of equations can be written in the so-called *quasi-linear* form, where the spatial derivatives of the flow variables (either in conservative or primitive form) are premultiplied by Jacobian matrices

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{U}}{\partial x} + \mathbf{B} \frac{\partial \mathbf{U}}{\partial y} + \mathbf{C} \frac{\partial \mathbf{U}}{\partial z} = \mathbf{S}, \quad (2.8a)$$

$$\frac{\partial \mathbf{Q}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{Q}}{\partial x} + \mathbf{B} \frac{\partial \mathbf{Q}}{\partial y} + \mathbf{C} \frac{\partial \mathbf{Q}}{\partial z} = \mathbf{S}, \quad (2.8b)$$

where vectors  $\mathbf{Q}$  and  $\mathbf{S}$ , and matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , when the primitive variables are used, have the form

$$\mathbf{Q} = \begin{bmatrix} \rho \\ u \\ v \\ w \\ p \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ f_x \\ f_y \\ f_z \\ (\gamma - 1)\rho \mathbf{f} \cdot \mathbf{u} \end{bmatrix}, \quad (2.9)$$

## 2. NUMERICAL METHOD

---

$$\mathbf{A} = \begin{bmatrix} u & \rho & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 1/\rho \\ 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & u & 0 \\ 0 & \gamma p & 0 & 0 & u \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} v & 0 & \rho & 0 & 0 \\ 0 & v & 0 & 0 & 0 \\ 0 & 0 & v & 0 & 1/\rho \\ 0 & 0 & 0 & v & 0 \\ 0 & 0 & \gamma p & 0 & v \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} w & 0 & 0 & \rho & 0 \\ 0 & w & 0 & 0 & 0 \\ 0 & 0 & w & 0 & 0 \\ 0 & 0 & 0 & w & 1/\rho \\ 0 & 0 & 0 & \gamma p & w \end{bmatrix}. \quad (2.10)$$

Equations (2.8) lead to results equivalent to the ones from Equation (2.5) when the flow field does not contain strong variations in the flow state and has the computational advantage of requiring the differentiation with respect to the various space directions on the same set of values ( $\mathbf{Q}$  or  $\mathbf{U}$ ) instead of having to compute the derivatives sequentially on  $\mathbf{E}$ ,  $\mathbf{F}$ , and  $\mathbf{G}$ .

The expressions for  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$  in case of conservative variables are slightly more complex and their formulation can be found in Hirsch [1990]. They can be obtained by the application of the chain rule to Equation (2.5) that gives

$$\mathbf{A} = \frac{\partial \mathbf{E}}{\partial \mathbf{U}}, \quad \mathbf{B} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}}, \quad \mathbf{C} = \frac{\partial \mathbf{G}}{\partial \mathbf{U}}. \quad (2.11)$$

The quasi-linear forms of the Euler Equations (2.8a) and (2.8b) allow for the definition of a *linearised* formulation that is oftenly used in CAA. In this case, the Jacobian matrices  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$  do not depend on the actual state of the flow but on a reference state that is considered constant throughout the computation at any point of the field. It is clear that this form only suits certain classes of problems in which the perturbations are of small amplitude so that the non-linear effects can be considered negligible and in which the mean value of any flow variable does not vary in time. This particular form of the equations has the computational advantage of avoiding to update the Jacobian

## 2. NUMERICAL METHOD

---

matrices at each time step. Therefore,  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$  can be computed once and for all at the beginning of the simulation.

### 2.1.2 The characteristic form of the Euler equations

Another form of the Euler equations can be determined by exploiting their hyperbolicity in time, the *characteristic* form. The Euler equations can be re-cast by a principal component analysis as a set of advection wave equations. Their solutions are advection waves of varied nature that propagate through the flow field. This formulation is particularly important for the numerical treatment of the boundary conditions as it allows to selectively reflect, absorb, or feed into the field the waves depending on the type of boundary that is imposed and on the sign of the wave group velocity.

The characteristics are families of curves (or surfaces when in 3D) along which some of the differential terms of the Euler equations become zero as they represent an invariant state for that particular variable. This allows to reduce the Euler equations to a set of ordinary differential equations (ODE).

Solutions of the Euler Equations (2.8b) can be written in wave-like form as

$$\mathbf{Q} = \tilde{\mathbf{Q}} e^{i(\boldsymbol{\xi} \cdot \mathbf{x} - \omega t)}, \quad (2.12)$$

where  $\tilde{\mathbf{Q}}$  is the wave amplitude at time  $t = 0$ ,  $\boldsymbol{\xi} = [\xi_x, \xi_y, \xi_z]$  is the wavenumber vector,  $\mathbf{x}$  is the position vector in space and the whole expression included between brackets represents the phase of the wave that propagates in the  $\boldsymbol{\xi}$  direction.

The study of matrix  $\mathbf{K}$ , defined as

$$\mathbf{K} = \xi_x \mathbf{A} + \xi_y \mathbf{B} + \xi_z \mathbf{C}, \quad (2.13)$$

## 2. NUMERICAL METHOD

---

leads to the computation of matrices  $\mathbf{L}$  and its inverse  $\mathbf{L}^{-1}$  that allow to transform an elemental variation of the flow state  $\delta\mathbf{Q}$  defined in terms of primitive variables  $\mathbf{Q}$  into an elemental variation of the characteristic variables. Determining  $\mathbf{L}$  is an eigenvector/eigenvalue problem where the eigenvalues  $\lambda_i$  are the solution to equation

$$\det(\mathbf{K} - \lambda_i\mathbf{I}) = 0, \quad (2.14)$$

in which  $\mathbf{I}$  is the identity matrix of the same dimensions of  $\mathbf{K}$  and  $\lambda_i$  represents the speed of propagation of the  $i^{\text{th}}$  wave. Left and right eigenvectors can be computed as

$$\begin{aligned} \mathbf{l}_i\mathbf{K} &= \lambda_i\mathbf{l}_i, \\ \mathbf{K}\mathbf{r}_i &= \lambda_i\mathbf{r}_i, \end{aligned} \quad (2.15)$$

where  $i$  is integer,  $1 \leq i \leq N$ , with  $N$  equal to the number of ordinary differential equations of the system. The left eigenvectors constitute the rows of matrix  $\mathbf{L}^{-1}$  and the right eigenvectors (that are proportional to the wave amplitudes) are the columns of matrix  $\mathbf{L}$ .

By premultiplying both sides of Equation (2.8b) by  $\mathbf{L}^{-1}$ , the following compatibility equation is obtained

$$\mathbf{L}^{-1}\frac{\partial\mathbf{Q}}{\partial t} + \mathbf{L}^{-1}\mathbf{A}\frac{\partial\mathbf{Q}}{\partial x} + \mathbf{L}^{-1}\mathbf{B}\frac{\partial\mathbf{Q}}{\partial y} + \mathbf{L}^{-1}\mathbf{C}\frac{\partial\mathbf{Q}}{\partial z} = \mathbf{L}^{-1}\mathbf{S}. \quad (2.16)$$

Equation (2.16) can be re-written as:

$$\mathbf{L}^{-1}\frac{\partial\mathbf{Q}}{\partial t} + (\mathbf{L}^{-1}\mathbf{A}\mathbf{L})\mathbf{L}^{-1}\frac{\partial\mathbf{Q}}{\partial x} + (\mathbf{L}^{-1}\mathbf{B}\mathbf{L})\mathbf{L}^{-1}\frac{\partial\mathbf{Q}}{\partial y} + (\mathbf{L}^{-1}\mathbf{C}\mathbf{L})\mathbf{L}^{-1}\frac{\partial\mathbf{Q}}{\partial z} = \mathbf{L}^{-1}\mathbf{S}, \quad (2.17)$$

## 2. NUMERICAL METHOD

---

which allows it to be stated in terms of the vector of the characteristic variables  $\mathbf{R}$  as

$$\frac{\partial \mathbf{R}}{\partial t} + (\mathbf{L}^{-1} \mathbf{A} \mathbf{L}) \frac{\partial \mathbf{R}}{\partial x} + (\mathbf{L}^{-1} \mathbf{B} \mathbf{L}) \frac{\partial \mathbf{R}}{\partial y} + (\mathbf{L}^{-1} \mathbf{C} \mathbf{L}) \frac{\partial \mathbf{R}}{\partial z} = \mathbf{L}^{-1} \mathbf{S}, \quad (2.18)$$

where

$$\frac{\partial \mathbf{R}}{\partial t} = \mathbf{L}^{-1} \frac{\partial \mathbf{Q}}{\partial t}, \quad \frac{\partial \mathbf{R}}{\partial x} = \mathbf{L}^{-1} \frac{\partial \mathbf{Q}}{\partial x}, \quad \frac{\partial \mathbf{R}}{\partial y} = \mathbf{L}^{-1} \frac{\partial \mathbf{Q}}{\partial y}, \quad \frac{\partial \mathbf{R}}{\partial z} = \mathbf{L}^{-1} \frac{\partial \mathbf{Q}}{\partial z}. \quad (2.19)$$

The choice of vector  $\boldsymbol{\xi}$  can only diagonalise one of the three Jacobian matrices  $\mathbf{A}$ ,  $\mathbf{B}$ , and  $\mathbf{C}$ . For example,  $\boldsymbol{\xi} = [1, 0, 0]$  diagonalises  $\mathbf{A}$ , while  $\mathbf{B}$  and  $\mathbf{C}$  remain non-diagonal.

It is important to notice that the characteristic variables are functions of the spatial direction defined by vector  $\boldsymbol{\xi}$  and that, in this context, they are a linear combination of the primitive variables through coefficients that are represented by the components of the left eigenvectors. These, in turn, are function of the primitive variables and therefore, not constant. This limits the possibility to generally define the vector  $\mathbf{R}$  that exists only if  $\mathbf{L}$  is constant. On the other hand, vector  $\delta \mathbf{R}$  of the characteristic variations can always be defined with respect to  $\delta \mathbf{Q}$  [Hirsch, 1990], where  $\delta$  represents an arbitrary variation (spatial or temporal partial derivative).

Solving Equation (2.14) for  $\lambda_i$  gives (in 3D)

$$\lambda_1 = \lambda_2 = \lambda_3 = \mathbf{u} \cdot \boldsymbol{\xi}, \quad (2.20a)$$

$$\lambda_4 = \mathbf{u} \cdot \boldsymbol{\xi} + c, \quad (2.20b)$$

$$\lambda_5 = \mathbf{u} \cdot \boldsymbol{\xi} - c, \quad (2.20c)$$

## 2. NUMERICAL METHOD

---

where  $c$  is the local isentropic speed of sound. This means that the first three waves have speed equal to the component of the flow velocity along the direction defined by  $\xi$  while the fourth and the fifth also include the contribution of the speed of sound in the opposite directions.

Having determined  $\lambda_i$  for the Euler equations, the eigenvector matrix  $\mathbf{L}$  is obtained from Equation (2.15).

It is important to notice that the eigenvectors are not univocally determined as any of their linear combinations is a valid solution to the eigenvalue problem. Therefore, matrix  $\mathbf{L}$  and its inverse appear in the literature under different forms related to specific normalisations, depending on the choices made by the authors. Throughout this document, the treatment given by Hirsch [1990] is used, as it is particularly suitable for the application of the characteristic-based boundary conditions developed in Kim & Lee [2000, 2003, 2004] and adopted in this work. This gives

$$\mathbf{L} = \begin{bmatrix} \xi_x & \xi_y & \xi_z & \frac{\rho}{2c} & \frac{\rho}{2c} \\ 0 & -\xi_z & \xi_y & \frac{\xi_x}{2} & -\frac{\xi_x}{2} \\ \xi_z & 0 & -\xi_x & \frac{\xi_y}{2} & -\frac{\xi_y}{2} \\ -\xi_y & \xi_x & 0 & \frac{\xi_z}{2} & -\frac{\xi_z}{2} \\ 0 & 0 & 0 & \frac{\rho c}{2} & \frac{\rho c}{2} \end{bmatrix}, \quad (2.21a)$$

$$\mathbf{L}^{-1} = \begin{bmatrix} \xi_x & 0 & \xi_z & -\xi_y & -\frac{\xi_x}{c^2} \\ \xi_y & -\xi_z & 0 & \xi_x & -\frac{\xi_y}{c^2} \\ \xi_z & \xi_y & -\xi_x & 0 & -\frac{\xi_z}{c^2} \\ 0 & \xi_x & \xi_y & \xi_z & \frac{1}{\rho c} \\ 0 & -\xi_x & -\xi_y & -\xi_z & \frac{1}{\rho c} \end{bmatrix}. \quad (2.21b)$$

## 2. NUMERICAL METHOD

---

Finally, it is possible to write the value of the variations of the characteristic variables as function of the primitive variables' variations along with their physical interpretation. These are

$$\delta R_1 = \delta\rho - \frac{1}{c^2}\delta p, \quad (\text{entropy wave}) \quad (2.22a)$$

$$\delta R_2 = -\xi_z\delta u + \xi_x\delta w, \quad (\text{vorticity wave}) \quad (2.22b)$$

$$\delta R_3 = \xi_y\delta u - \xi_x\delta v, \quad (\text{vorticity wave}) \quad (2.22c)$$

$$\delta R_4 = \xi_x\delta u + \xi_y\delta v + \xi_z\delta w + \frac{1}{\rho c}\delta p, \quad (\text{leftward propagating acoustic wave}) \quad (2.22d)$$

$$\delta R_5 = -\xi_x\delta u - \xi_y\delta v - \xi_z\delta w + \frac{1}{\rho c}\delta p. \quad (\text{rightward propagating acoustic wave}) \quad (2.22e)$$

Note that these definitions are derived from a version of matrix  $\mathbf{L}^{-1}$  obtained with a different normalisation (also given by Hirsch [1990]) but they are often preferred over those obtainable from Equation (2.21b), because their physical interpretation is more intuitive.

### 2.1.3 Transformations between the different forms of the flow variables

As shown by Hirsch [1990], when dealing with the Euler equations, as seen in Section 2.1.2, it is possible to switch from primitive variables  $\mathbf{Q}$  to characteristic variables

## 2. NUMERICAL METHOD

---

$\mathbf{R}$  and back by the use of the eigenvector matrices  $\mathbf{L}$  and  $\mathbf{L}^{-1}$

$$\delta\mathbf{R} = \mathbf{L}^{-1}\delta\mathbf{Q}, \quad \delta\mathbf{Q} = \mathbf{L}\delta\mathbf{R}, \quad (2.23)$$

where  $\delta$  represents an arbitrary variation (spatial or temporal partial derivative).

More generally, as the transformation is not limited to small variations only, it is possible to switch from the primitive  $\mathbf{Q}$  to the conservative  $\mathbf{U}$  formulation through matrix  $\mathbf{M}$  and its inverse

$$\mathbf{U} = \mathbf{M}\mathbf{Q}, \quad \mathbf{Q} = \mathbf{M}^{-1}\mathbf{U}, \quad (2.24)$$

where  $\mathbf{M}$  and  $\mathbf{M}^{-1}$  are

$$\mathbf{M} = \frac{\partial\mathbf{U}}{\partial\mathbf{Q}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 \\ v & 0 & \rho & 0 & 0 \\ w & 0 & 0 & \rho & 0 \\ \frac{(u^2+v^2+w^2)}{2} & \rho u & \rho v & \rho w & \frac{1}{\gamma-1} \end{bmatrix}, \quad (2.25)$$

$$\mathbf{M}^{-1} = \frac{\partial\mathbf{Q}}{\partial\mathbf{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ -\frac{w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ \frac{(\gamma-1)(u^2+v^2+w^2)}{2} & -(\gamma-1)u & -(\gamma-1)v & -(\gamma-1)w & \gamma-1 \end{bmatrix}. \quad (2.26)$$

The combination of the above matrices with Equation (2.21a) and (2.21b) allows to directly transform small variations of conservative variables into variations of the

## 2. NUMERICAL METHOD

---

characteristic variables through matrix  $\mathbf{P}$  and its inverse

$$\delta\mathbf{U} = \mathbf{P}\delta\mathbf{R}, \quad \delta\mathbf{R} = \mathbf{P}^{-1}\delta\mathbf{U}, \quad (2.27)$$

where

$$\mathbf{P} = \mathbf{M}\mathbf{L}, \quad (2.28)$$

and

$$\mathbf{P}^{-1} = \mathbf{L}^{-1}\mathbf{M}^{-1}. \quad (2.29)$$

Matrices  $\mathbf{P}$  and  $\mathbf{P}^{-1}$  are determined analytically from Equations (2.28), (2.25), (2.21a) and (2.29), (2.26), (2.21b) and are reported in Hirsch [1990] and in Kim & Lee [2000, 2004].

### 2.1.4 Non-dimensionalisation

A final note regarding the internal non-dimensionalisation of all the variables in the code: following the recommendations of the Standard Interface Data Structures (SIDS), on which the CFD General Notation System (CGNS) is built, it is made with respect to a set of only four reference scales: length  $l_\infty$ , static speed of sound  $c_\infty$ , static density  $\rho_\infty$ , and static temperature  $T_\infty$  [NASA, 2012]. All the other variables are consistently non-dimensionalised with respect to these. For example

$$p_{nd} = p_d/(\rho_\infty c_\infty^2), \quad (2.30a)$$

$$\mu_{nd} = \mu_d/(\rho_\infty c_\infty l_\infty), \quad (2.30b)$$

$$t_{nd} = t_d c_\infty / l_\infty. \quad (2.30c)$$

## 2. NUMERICAL METHOD

---

### 2.2 Spatial discretisation

In the present work, a compact sixth-order central finite-difference scheme is used for all the computations. The spatial derivative  $f'_i$  at point  $i$  depends on the values of function  $f$  and on the derivative itself in the surrounding points as

$$a_1 (f'_{i+1} + f'_{i-1}) + a_2 f'_i = \frac{1}{\Delta x} [b_1 (f_{i+2} - f_{i-2}) + b_2 (f_{i+1} - f_{i-1})], \quad (2.31)$$

where  $\Delta x$  is the uniform distance between the nodes, and  $a_i$  and  $b_i$  are two sets of coefficients that are [Lele, 1992]

$$a_1 = \frac{1}{5}, \quad a_2 = \frac{3}{5}, \quad b_1 = \frac{1}{60}, \quad b_2 = \frac{7}{15}. \quad (2.32)$$

By substituting the spatial derivatives  $f'$  on the left-hand side of Equation (2.31) with their Taylor series expansion, it can be shown that the coefficients in Equations (2.32) ensure that the leading term of the truncation error is of sixth order.

Central high-order schemes like this are particularly suited to the use in CAA because they have no dissipation error and, therefore, they can propagate acoustic waves over long distances with little attenuation in amplitude.

The ability of this scheme to propagate waves depending on their spectral content can be studied using the 1D advection wave equation as in Hu *et al.* [1996]

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0, \quad (2.33)$$

where  $c$  is the group velocity with which the wave propagates.

## 2. NUMERICAL METHOD

---

Assume a solution for  $u$  of the type

$$u(x, t) = \tilde{u}_0 e^{i(kx - \omega t)}, \quad (2.34)$$

where  $k$  is the *real* wavenumber and  $\omega$  the angular frequency.

Applying the discretisation of Equation (2.31) to Equation (2.33), the semi-discrete equation at point  $i$  is obtained

$$\frac{\partial u_i}{\partial t} - c \frac{a_1}{a_2} \left( \frac{\partial u_{i+1}}{\partial x} - \frac{\partial u_{i-1}}{\partial x} \right) + \frac{c}{\Delta x} \left[ \frac{b_1}{a_2} (u_{i+2} - u_{i-2}) + \frac{b_2}{a_2} (u_{i+1} - u_{i-1}) \right] = 0. \quad (2.35)$$

Through Fourier analysis, the semi-discrete Equation (2.35) becomes

$$\frac{\partial \tilde{u}}{\partial t} + i c k^* \tilde{u} = 0, \quad (2.36)$$

where  $\tilde{u}$  is the spatial Fourier transform of  $u$  and  $k^*$  is the *numerical* (or *effective*) wavenumber that is related to the real wavenumber  $k$  by the relation [Lele, 1992]

$$k^* = \frac{b_2 \sin(k) + (b_1/2) \sin(2k)}{a_2 + 2a_1 \cos(k)}. \quad (2.37)$$

As can be seen by Figure 2.1 where the effective wavenumber  $k^*$  is plotted as a function of the real wavenumber  $k$  (the continuous line represents the ideal case of an exact differentiation), the former is a good approximation of the latter only on a restricted range of frequencies. A *maximum resolvable wavenumber*  $k_c^*$  can be defined depending on a maximum tolerable error between two wavenumbers. For instance, Hu *et al.* [1996] set the maximum error  $|k^* - k|$  equal to  $0.005/\Delta x$ . The value of  $k_c^*$  can be determined numerically with iterative methods. For a maximum tolerable error of

## 2. NUMERICAL METHOD

---

$0.005/\Delta x$ ,  $k_c^*$  is  $1.355/\Delta x$  and the relative minimum number of points-per-wavelength  $N_\lambda$  is immediately obtained as  $N_\lambda = 2\pi / (k_c^* \Delta x) = 4.63$ . Another important value that can be derived from Equation (2.37) is the *maximum effective wavenumber*  $k_{max}^*$  that in Figure 2.1 corresponds to the y coordinate of the maximum of the function shown as a dashed line. It is an absolute value and for the compact sixth-order scheme of Equation (2.31) it is equal to  $1.989/\Delta x$ .

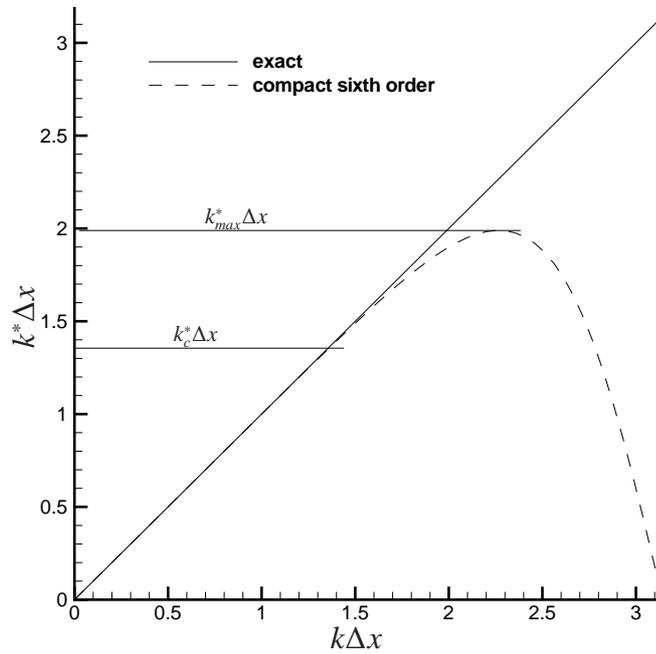


Figure 2.1: Numerical wavenumber as a function of the real wavenumber for the an ideal exact spatial discretisation and for the compact sixth-order scheme of Equation (2.31).

### 2.2.1 Hixon's prefactored compact scheme

The compact scheme represented by Equation (2.31) can be numerically implemented in many ways. In this work the *prefactored small-stencil* version given by Hixon

## 2. NUMERICAL METHOD

---

[2000b] is used. In matrix form a compact scheme of any order can be written as

$$\mathbf{A}\mathbf{d} = \frac{1}{\Delta x}\mathbf{C}\mathbf{f}, \quad (2.38)$$

where, in the case of a sixth-order scheme of Equation (2.31),  $\mathbf{A}$  is a tridiagonal matrix of the coefficients  $a_i$  ( $i = 1, 2$ ),  $\mathbf{d}$  is the vector of the spatial derivatives of function  $f$ ,  $\mathbf{C}$  is a pentadiagonal matrix of the coefficients  $b_i$  ( $i = 1, 2$ ), and  $\mathbf{f}$  is the vector of the values of function  $f$ .

Hixon introduces the forward  $\mathbf{d}^F$  and backwards  $\mathbf{d}^B$  derivative operators in the equations

$$\mathbf{A}^F\mathbf{d}^F = \frac{1}{\Delta x}\mathbf{B}^F\mathbf{f}, \quad (2.39a)$$

$$\mathbf{A}^B\mathbf{d}^B = \frac{1}{\Delta x}\mathbf{B}^B\mathbf{f}, \quad (2.39b)$$

that satisfy

$$\mathbf{d} = \frac{1}{2}(\mathbf{d}^F + \mathbf{d}^B), \quad (2.40)$$

where  $\mathbf{A}^F$ ,  $\mathbf{A}^B$ ,  $\mathbf{B}^F$ , and  $\mathbf{B}^B$  are unknown matrices to which the following restrictions are applied:

$$\mathbf{A}^F = [\mathbf{A}^B]^T, \quad (2.41a)$$

$$\mathbf{B}^F = -[\mathbf{B}^B]^T. \quad (2.41b)$$

Under these conditions, Equations (2.39) can be written in expanded form for a generic

## 2. NUMERICAL METHOD

---

node  $i$  as

$$\begin{aligned} c_1 \left( \frac{\partial f_{i+1}}{\partial x} \right)^F + (1 - c_1 - c_3) \left( \frac{\partial f_i}{\partial x} \right)^F + c_3 \left( \frac{\partial f_{i-1}}{\partial x} \right)^F &= \\ &= \frac{1}{\Delta x} [c_2 f_{i+1} - (2c_2 - 1) f_i - (1 - c_2) f_{i-1}], \end{aligned} \quad (2.42a)$$

$$\begin{aligned} c_3 \left( \frac{\partial f_{i+1}}{\partial x} \right)^B + (1 - c_1 - c_3) \left( \frac{\partial f_i}{\partial x} \right)^B + c_1 \left( \frac{\partial f_{i-1}}{\partial x} \right)^B &= \\ &= \frac{1}{\Delta x} [(1 - c_2) f_{i+1} + (2c_2 - 1) f_i - c_2 f_{i-1}]. \end{aligned} \quad (2.42b)$$

Isolating  $\left( \frac{\partial f_i}{\partial x} \right)^F$  and  $\left( \frac{\partial f_i}{\partial x} \right)^B$ , and substituting them into Equation (2.40), leads to

$$\begin{aligned} (c_1 c_3) (f'_{i+2} + f'_{i-2}) + (c_1 + c_3) (1 - c_1 - c_3) (f'_{i+1} + f'_{i-1}) [c_1^2 + c_3^2 + (1 - c_1 - c_3)^2] f'_i &= \\ = \frac{1}{\Delta x} \left[ \frac{1}{2} (c_1 + c_2 (c_3 - c_1)) (f_{i+2} - f_{i-2}) + \left( \frac{1}{2} - c_1 - c_2 (c_3 - c_1) \right) (f_{i+1} - f_{i-1}) \right], \end{aligned} \quad (2.43)$$

that can be directly compared to the equation defining the finite-difference scheme to obtain the coefficients  $c_1$ ,  $c_2$  and  $c_3$ .

In the case of a sixth-order scheme of Equation (2.31), Hixon gives the solution

$$c_1 = \frac{1}{2} - \frac{1}{2\sqrt{5}}, \quad (2.44a)$$

$$c_2 = 1 - \frac{1}{30c_1}, \quad (2.44b)$$

$$c_3 = 0. \quad (2.44c)$$

With the same procedure shown here Hixon [2000b] also defines a fourth-order and an eighth-order scheme.

## 2. NUMERICAL METHOD

---

As a result, tridiagonal matrix  $A$  is split into the two bidiagonal matrices  $A^F$  and  $A^B$ , and the pentadiagonal matrix  $B$  is substituted by the tridiagonal matrices  $B^F$ , and  $B^B$ . This means that the spatial derivative at a node of index  $i$  within the computational field can be obtained as the sum of two contributions (the half-derivatives) that depend on the value of the function  $f$  at  $i$ ,  $i + 1$  and  $i - 1$  (the stencil is reduced from five to three points) and on the value of the same half-derivative of only one of the neighbouring nodes. This allows to execute the computation of the two half-derivatives in an independent fashion with “sweeps” starting from the opposite boundaries. The complete spatial derivatives are then recovered simply adding up the two components as in Equation (2.40). Hixon [2000b] claims that this technique, although implying one addition per each point more, has to be preferred with respect to the most optimised Thomas algorithm “since additions are much less expensive than multiplications” and “the advantages of the prefactorization offset the slight increase in computational cost”.

### 2.2.2 Treatment of the boundaries

For the sweeps to start from the opposite sides of the field, an initial approximation of the two half-derivatives at the boundaries is needed. Hixon [2000b] suggests to compute them by using high-order explicit approximations that, depending on the type of boundary, can use either one-sided or centred stencils.

For walls and outer boundaries, the sixth-order one-sided approximations with 7-

## 2. NUMERICAL METHOD

---

point stencil at nodes with index  $i = 1$  and  $i = imax$  have the form

$$\frac{\partial f_1^B}{\partial x} = \sum_{i=1}^7 s_i f_i, \quad (2.45a)$$

$$\frac{\partial f_1^F}{\partial x} = - \sum_{i=1}^7 e_{imax+1-i} f_i, \quad (2.45b)$$

$$\frac{\partial f_{imax}^B}{\partial x} = \sum_{i=imax-6}^{imax} e_i f_i, \quad (2.45c)$$

$$\frac{\partial f_{imax}^F}{\partial x} = - \sum_{i=imax-6}^{imax} s_{imax+1-i} f_i, \quad (2.45d)$$

where:

$$s_1 = \frac{-610 - 96\sqrt{5}}{300}, \quad e_{imax} = \frac{860 - 96\sqrt{5}}{300}, \quad (2.46a)$$

$$s_2 = \frac{1300 + 346\sqrt{5}}{300}, \quad e_{imax-1} = \frac{-2300 + 346\sqrt{5}}{300}, \quad (2.46b)$$

$$s_3 = \frac{-1375 - 535\sqrt{5}}{300}, \quad e_{imax-2} = \frac{3125 - 535\sqrt{5}}{300}, \quad (2.46c)$$

$$s_4 = \frac{1125 + 475\sqrt{5}}{300}, \quad e_{imax-3} = \frac{-2875 + 475\sqrt{5}}{300}, \quad (2.46d)$$

$$s_5 = \frac{-600 - 260\sqrt{5}}{300}, \quad e_{imax-4} = \frac{1650 - 260\sqrt{5}}{300}, \quad (2.46e)$$

$$s_6 = \frac{185 + 81\sqrt{5}}{300}, \quad e_{imax-5} = \frac{-535 + 81\sqrt{5}}{300}, \quad (2.46f)$$

$$s_7 = \frac{-25 - 11\sqrt{5}}{300}, \quad e_{imax-6} = \frac{75 - 11\sqrt{5}}{300}. \quad (2.46g)$$

For symmetrical, periodical and inter-block boundaries, the ninth-order centred 11-point stencil is used. Although a tenth-order formulation is possible, Hixon [2000b] uses one of the available degrees of freedom to match more closely the performance of

## 2. NUMERICAL METHOD

---

the interior scheme. At a node with index  $j$  the formulation is

$$\frac{\partial f_j^F}{\partial x} = \sum_{i=-5}^5 b_i f_{i+j}, \quad (2.47a)$$

$$\frac{\partial f_j^B}{\partial x} = - \sum_{i=-5}^5 b_{-i} f_{i+j}, \quad (2.47b)$$

where:

$$b_{-5} = -0.000429279312812, \quad b_5 = 0.00115802227448959, \quad (2.48a)$$

$$b_{-4} = 0.00444354988633981, \quad b_4 = -0.01539771995493, \quad (2.48b)$$

$$b_{-3} = -0.0192836978258309, \quad b_3 = 0.0997639212217881, \quad (2.48c)$$

$$b_{-2} = 0.0507098949263327, \quad b_2 = -0.425480581264144, \quad (2.48d)$$

$$b_{-1} = -0.264423227620118, \quad b_1 = 1.40224343904655, \quad (2.48e)$$

$$b_0 = -0.833304321377665. \quad (2.48f)$$

The procedure used to obtain these coefficients, that in the original article by Hixon [2000b] are given in single precision only, is described in detail by Spisso [unpublished] and it is applied in the following part of this section to obtain upwind and downwind stencils, not previously available, needed for the solution of the second test case as described in Section 3.2. These stencils allow the computation of the half-derivatives at a node of index  $i$  based on the values of function  $f$  from index  $i - 1$  to  $i + 5$  (referred to as “(5-1) 7-point stencil”), or based on the values of the function from index  $i - 2$  to  $i + 4$  (referred to as “(4-2) 7-point stencil”). The same coefficients can also be used, by inverting their signs, for the application of the stencil in the opposite spatial direction.

## 2. NUMERICAL METHOD

---

For example, the coefficients for the backward derivative of the (4-2) 7-point stencil are computed. First, the coefficients  $A$ ,  $B$ ,  $K$ ,  $D$ ,  $E$ ,  $F$ , and  $G$  of the Taylor expansion of the two half-derivatives introduced in Equations (2.39) [Spisso, unpublished]

$$\begin{aligned} \frac{\partial f_i^F}{\partial x} = & \frac{\partial f_i}{\partial x} + A\Delta x \frac{\partial^2 f_i}{\partial x^2} - B\Delta x^3 \frac{\partial^4 f_i}{\partial x^4} + K\Delta x^5 \frac{\partial^6 f_i}{\partial x^6} + D\Delta x^6 \frac{\partial^7 f_i}{\partial x^7} \\ & + E\Delta x^7 \frac{\partial^8 f_i}{\partial x^8} + F\Delta x^8 \frac{\partial^9 f_i}{\partial x^9} + G\Delta x^9 \frac{\partial^{10} f_i}{\partial x^{10}} + O(\Delta x)^{11}, \end{aligned} \quad (2.49a)$$

$$\begin{aligned} \frac{\partial f_i^B}{\partial x} = & \frac{\partial f_i}{\partial x} - A\Delta x \frac{\partial^2 f_i}{\partial x^2} + B\Delta x^3 \frac{\partial^4 f_i}{\partial x^4} - K\Delta x^5 \frac{\partial^6 f_i}{\partial x^6} + D\Delta x^6 \frac{\partial^7 f_i}{\partial x^7} \\ & - E\Delta x^7 \frac{\partial^8 f_i}{\partial x^8} + F\Delta x^8 \frac{\partial^9 f_i}{\partial x^9} - G\Delta x^9 \frac{\partial^{10} f_i}{\partial x^{10}} + O(\Delta x)^{11}, \end{aligned} \quad (2.49b)$$

have to be determined. Note that when the two Equations (2.49) are added, the contribution from the even derivatives is always zero as they have opposite coefficients.  $A$ ,  $B$ ,  $K$ ,  $D$ ,  $E$ ,  $F$ , and  $G$  are characteristic of the adopted spatial scheme and can be computed as a function of coefficients  $c_1$ ,  $c_2$ ,  $c_3$  of Equation (2.44) by substituting Equations (2.49) into (2.42). For Hixon's sixth-order scheme they are equal to [Spisso, unpublished]

$$A = \frac{1}{3 + 3\sqrt{5}}, \quad B = \frac{5 + 7\sqrt{5}}{720}, \quad (2.50a)$$

$$K = \frac{-5 + 17\sqrt{5}}{21600}, \quad D = \frac{1}{2100}, \quad (2.50b)$$

$$E = \frac{-25 - 227\sqrt{5}}{6048000}, \quad F = \frac{-277}{4536000}, \quad (2.50c)$$

$$G = \frac{1475 - 839\sqrt{5}}{544320000}. \quad (2.50d)$$

## 2. NUMERICAL METHOD

---

The (4-2) 7-point backward derivative can be written in expanded form as

$$\frac{\partial f_i^B}{\partial x} = \frac{1}{\Delta x} (s_{-2}f_{i-2} + s_{-1}f_{i-1} + s_0f_i + s_1f_{i+1} + s_2f_{i+2} + s_3f_{i+3} + s_4f_{i+4}), \quad (2.51)$$

where coefficients  $s_j$  (with  $-2 \leq j \leq 4$ ) are the unknowns.

By substituting the values of the function  $f$  with their Taylor expansion up to the sixth order and neglecting the rest, and then regrouping the values of the derivatives of different order, Equation (2.51) becomes

$$\begin{aligned} \frac{\partial f_i^B}{\partial x} = & \left[ \frac{1}{\Delta x} (s_{-2} + s_{-1} + s_0 + s_1 + s_2 + s_3 + s_4) f_i \right. \\ & + (-2s_{-2} - s_{-1} + s_1 + 2s_2 + 3s_3 + 4s_4) \frac{\partial f}{\partial x} \\ & + \left( \frac{2^2}{2!} s_{-2} + \frac{1}{2!} s_{-1} + \frac{1}{2!} s_1 + \frac{2^2}{2!} s_2 + \frac{3^2}{2!} s_3 + \frac{4^2}{2!} s_4 \right) \Delta x \frac{\partial^2 f}{\partial x^2} \\ & + \left( -\frac{2^3}{3!} s_{-2} - \frac{1}{3!} s_{-1} + \frac{1}{3!} s_1 + \frac{2^3}{3!} s_2 + \frac{3^3}{3!} s_3 + \frac{4^3}{3!} s_4 \right) (\Delta x)^2 \frac{\partial^3 f}{\partial x^3} \\ & + \left( \frac{2^4}{4!} s_{-2} + \frac{1}{4!} s_{-1} + \frac{1}{4!} s_1 + \frac{2^4}{4!} s_2 + \frac{3^4}{4!} s_3 + \frac{4^4}{4!} s_4 \right) (\Delta x)^3 \frac{\partial^4 f}{\partial x^4} \\ & + \left( -\frac{2^5}{5!} s_{-2} - \frac{1}{5!} s_{-1} + \frac{1}{5!} s_1 + \frac{2^5}{5!} s_2 + \frac{3^5}{5!} s_3 + \frac{4^5}{5!} s_4 \right) (\Delta x)^4 \frac{\partial^5 f}{\partial x^5} \\ & \left. + \left( \frac{2^6}{6!} s_{-2} + \frac{1}{6!} s_{-1} + \frac{1}{6!} s_1 + \frac{2^6}{6!} s_2 + \frac{3^6}{6!} s_3 + \frac{4^6}{6!} s_4 \right) (\Delta x)^5 \frac{\partial^6 f}{\partial x^6} \right]. \end{aligned} \quad (2.52)$$

## 2. NUMERICAL METHOD

---

Finally, comparing Equation (2.52) and (2.49b)

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 & 3 & 4 \\ \frac{2^2}{2!} & \frac{1}{2!} & 0 & \frac{1}{2!} & \frac{2^2}{2!} & \frac{3^2}{2!} & \frac{4^2}{2!} \\ -\frac{2^3}{3!} & -\frac{1}{2!} & 0 & \frac{1}{2!} & \frac{2^3}{3!} & \frac{3^3}{3!} & \frac{4^3}{3!} \\ \frac{2^4}{4!} & \frac{1}{4!} & 0 & \frac{1}{4!} & \frac{2^4}{4!} & \frac{3^4}{4!} & \frac{4^4}{4!} \\ -\frac{2^5}{5!} & -\frac{1}{5!} & 0 & \frac{1}{5!} & \frac{2^5}{5!} & \frac{3^5}{5!} & \frac{4^5}{5!} \\ \frac{2^6}{6!} & \frac{1}{6!} & 0 & \frac{1}{6!} & \frac{2^6}{6!} & \frac{3^6}{6!} & \frac{4^6}{6!} \end{bmatrix} \begin{bmatrix} s_{-2} \\ s_{-1} \\ s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ -A \\ 0 \\ B \\ 0 \\ -K \end{bmatrix}. \quad (2.53)$$

By numerically inverting the matrix on the left-hand side of Equation (2.53), the stencil coefficients of Equation (2.51) are found

$$s_{-2} = \frac{20 + 8\sqrt{5}}{600}, \quad s_{-1} = \frac{-190 - 78\sqrt{5}}{600}, \quad (2.54a)$$

$$s_0 = \frac{-450 + 130\sqrt{5}}{600}, \quad s_1 = \frac{850 - 50\sqrt{5}}{600}, \quad (2.54b)$$

$$s_2 = \frac{-300 - 20\sqrt{5}}{600}, \quad s_3 = \frac{80 + 12\sqrt{5}}{600}, \quad (2.54c)$$

$$s_4 = \frac{-10 - 2\sqrt{5}}{600}. \quad (2.54d)$$

In a similar way, the coefficients of the forward derivative at node  $i$  defined as

$$\frac{\partial f_i^F}{\partial x} = - \sum_{j=-2}^4 e_j f_{i+j}, \quad (2.55)$$

## 2. NUMERICAL METHOD

---

are recovered

$$e_{-2} = \frac{-20 + 8\sqrt{5}}{600}, \quad e_{-1} = \frac{290 - 78\sqrt{5}}{600}, \quad (2.56a)$$

$$e_0 = \frac{250 + 130\sqrt{5}}{600}, \quad e_1 = \frac{-750 - 50\sqrt{5}}{600}, \quad (2.56b)$$

$$e_2 = \frac{300 - 20\sqrt{5}}{600}, \quad e_3 = \frac{-80 + 12\sqrt{5}}{600}, \quad (2.56c)$$

$$e_4 = \frac{10 - 2\sqrt{5}}{600}. \quad (2.56d)$$

For the sake of completeness, the (5-1) 7-point stencils along with their coefficients are also shown

$$\frac{\partial f_i^B}{\partial x} = \sum_{j=-1}^5 s_j f_{i+j}, \quad \frac{\partial f_i^F}{\partial x} = - \sum_{j=-1}^5 e_j f_{i+j}, \quad (2.57a)$$

$$s_{-1} = \frac{-25 - 11\sqrt{5}}{300}, \quad e_{-1} = \frac{75 - 11\sqrt{5}}{300}, \quad (2.57b)$$

$$s_0 = \frac{-435 - 19\sqrt{5}}{300}, \quad e_0 = \frac{335 - 19\sqrt{5}}{300}, \quad (2.57c)$$

$$s_1 = \frac{775 + 115\sqrt{5}}{300}, \quad e_1 = \frac{-725 + 115\sqrt{5}}{300}, \quad (2.57d)$$

$$s_2 = \frac{-500 - 150\sqrt{5}}{300}, \quad e_2 = \frac{500 - 150\sqrt{5}}{300}, \quad (2.57e)$$

$$s_3 = \frac{250 + 90\sqrt{5}}{300}, \quad e_3 = \frac{-250 + 90\sqrt{5}}{300}, \quad (2.57f)$$

$$s_4 = \frac{-75 - 29\sqrt{5}}{300}, \quad e_4 = \frac{75 - 29\sqrt{5}}{300}, \quad (2.57g)$$

$$s_5 = \frac{10 + 4\sqrt{5}}{300}, \quad e_5 = \frac{-10 + 4\sqrt{5}}{300}. \quad (2.57h)$$

More generally, half-derivatives at node with index  $i = 1$  based on one-sided sten-

## 2. NUMERICAL METHOD

---

cils including  $l$  points have the form

$$\frac{\partial f_1^B}{\partial x} = \sum_{j=1}^l s_j f_j, \quad (2.58a)$$

$$\frac{\partial f_1^F}{\partial x} = - \sum_{j=1}^l e_j f_j. \quad (2.58b)$$

Table 2.1 lists the numerical coefficients to be applied to Equations (2.58) in order to obtain lower-order approximations of the half-derivatives. These coefficients were obtained with the same procedure used to derive the one sided coefficients of Equations (2.54) and (2.57). The same coefficients can be used to compute the half-derivatives in the opposite spatial direction by inverting their sign. Some of the coefficients in Table 2.1 were used for the solution of the second test case in Section 3.2 and of the fourth test case in Section 3.4.

Similarly, Table 2.2 shows the coefficients  $b_j$  to be used in the equations that define backward and forward half-derivatives at node with index  $i$  based on centred stencils of length  $2l + 1$

$$\frac{\partial f_i^B}{\partial x} = \sum_{j=-l}^l b_j f_{i+j}, \quad (2.59a)$$

$$\frac{\partial f_i^F}{\partial x} = - \sum_{j=-l}^l b_{-j} f_{i+j}. \quad (2.59b)$$

### 2.2.3 High-order filters

The high-order finite-difference schemes tend to introduce spurious waves without physical meaning into the numerical predictions. This is due to the fact that the difference equations have additional solutions that are not suitable for the original dif-

## 2. NUMERICAL METHOD

---

Table 2.1: Coefficients  $s_j$  and  $e_j$  for lower-order one-sided stencils of various length and order of accuracy.

j	5 <sup>th</sup> order (l=6)		4 <sup>th</sup> order (l=5)		3 <sup>rd</sup> order (l=4)		2 <sup>nd</sup> order (l=3)	
	$s_j$	$e_j$	$s_j$	$e_j$	$s_j$	$e_j$	$s_j$	$e_j$
1	$\frac{-117-17\sqrt{5}}{60}$	$\frac{157-17\sqrt{5}}{60}$	$\frac{-110-14\sqrt{5}}{60}$	$\frac{140-14\sqrt{5}}{60}$	$\frac{-20-2\sqrt{5}}{12}$	$\frac{24-2\sqrt{5}}{12}$	$\frac{-17-\sqrt{5}}{12}$	$\frac{19-\sqrt{5}}{12}$
2	$\frac{230+56\sqrt{5}}{60}$	$\frac{-370+56\sqrt{5}}{60}$	$\frac{195+41\sqrt{5}}{60}$	$\frac{-285+41\sqrt{5}}{60}$	$\frac{31+5\sqrt{5}}{12}$	$\frac{-41+5\sqrt{5}}{12}$	$\frac{22+2\sqrt{5}}{12}$	$\frac{-26+2\sqrt{5}}{12}$
3	$\frac{-200-74\sqrt{5}}{60}$	$\frac{400-74\sqrt{5}}{60}$	$\frac{-130-44\sqrt{5}}{60}$	$\frac{230-44\sqrt{5}}{60}$	$\frac{-14-4\sqrt{5}}{12}$	$\frac{22-4\sqrt{5}}{12}$	$\frac{-5-\sqrt{5}}{12}$	$\frac{7-\sqrt{5}}{12}$
4	$\frac{125+51\sqrt{5}}{60}$	$\frac{-275+51\sqrt{5}}{60}$	$\frac{55+21\sqrt{5}}{60}$	$\frac{-105+21\sqrt{5}}{60}$	$\frac{3+5\sqrt{5}}{12}$	$\frac{-5+5\sqrt{5}}{12}$		
5	$\frac{-45-19\sqrt{5}}{60}$	$\frac{105-19\sqrt{5}}{60}$	$\frac{-10-4\sqrt{5}}{60}$	$\frac{20-4\sqrt{5}}{60}$				
6	$\frac{7+3\sqrt{5}}{60}$	$\frac{-17+3\sqrt{5}}{60}$						

Table 2.2: Coefficients  $b_j$  for lower-order centred stencils of various length and order of accuracy.

j	8 <sup>th</sup> order (l=4)	6 <sup>th</sup> order (l=3)	4 <sup>th</sup> order (l=2)	2 <sup>nd</sup> order (l=1)
-4	0.00184262134833348			
-3	-0.0172874108616684	-0.00921310674166744		
-2	0.0729375719116774	0.068010840825008	0.0460655337083369	
-1	-0.349306533340048	-0.386119737833376	-0.414589803375032	-0.396994335208351
0	-0.683039164783257	-0.578689325833263	-0.429618127333277	-0.206011329583298
1	1.24402679999329	1.11388026216662	0.918743529958302	0.603005664791649
2	-0.320395761421656	-0.231989159174992	-0.12060113295833	
3	0.0560459224716651	0.024120226591666		
4	-0.00482404531833322			

## 2. NUMERICAL METHOD

---

ferential equations. There are also other sources of disturbances that can degrade the solution. Among them are the discontinuities in the definition of the boundary conditions and the discontinuities in the governing equations. The two main approaches for the removal of these unwanted high-frequency oscillations are the application of a damping coefficient as suggested, for example, by Tam & Webb [1993], or the application, at each time step, of a low-pass filter. The latter solution was adopted for this study.

Filters, like finite-difference schemes, can be explicit or implicit and are characterised by their order of accuracy. It is important to notice that, to preserve the quality of the results, the applied filter has to be at least two orders higher than the level of accuracy of the finite-difference scheme used in conjunction with it.

The filter applied in this work is the implicit filter proposed by Gaitonde & Visbal [1998, 1999]. The general formulation of this filter of  $2N^{th}$ -order is

$$\alpha_f \hat{\mathbf{Q}}_{i-1} + \hat{\mathbf{Q}}_i + \alpha_f \hat{\mathbf{Q}}_{i+1} = \sum_{n=0}^N \frac{a_n}{2} (\mathbf{Q}_{i+n} + \mathbf{Q}_{i-n}), \quad (2.60)$$

where  $\mathbf{Q}_i$  is the vector of the unfiltered flow solution in primitive form (the treatment of the conservative form is equivalent) at node of index  $i$ ,  $\hat{\mathbf{Q}}_i$  is the vector of the filtered flow solution at the same node,  $a_n$  are  $N + 1$  coefficients that characterise the filter, and  $\alpha_f$  is a free parameter with a range  $-0.5 < \alpha_f \leq 0.5$ . The sets of  $N + 1$  coefficients  $a_n$  for filters from the second to the tenth order of accuracy are reported by Gaitonde & Visbal [1999]. They are obtained in terms of the free parameter  $\alpha_f$  with a Taylor series expansion based on templates proposed by Lele [1992] for spatial finite-difference schemes with spectral-like resolution. The free coefficient  $\alpha_f$  controls

## 2. NUMERICAL METHOD

---

the dissipation introduced by the filtering. The dissipation decreases with the increase of  $\alpha_f$  until it reaches the value 0.5 where the application of the filter has no effect on the flow solution. The free coefficient  $\alpha_f$  has to be set depending on the distortion of the employed grid. As reported by Gaitonde & Visbal [1999], it is usually appropriate to adopt values in the range  $0.3 < \alpha_f < 0.5$  with lower values required only for grids containing discontinuities in the metrics.

The spectral function  $SF$  of the filters defined by Equation (2.60) is

$$SF(\omega) = \frac{\sum_{n=0}^N a_n \cos(n\omega)}{1 + 2\alpha_f \cos(\omega)}, \quad (2.61)$$

where  $\omega$  is the angular frequency. Due to the symmetry of Equation (2.60) the spectral function  $SF$  is always real and the filter is non-dispersive.

In multidimensional problems, the filter is applied in sequence to the various directions of the space alternating them at every stage to eliminate the possible bias. The frequency of application of the filter can be varied depending on the requirements of the problem. It is usually applied at the end of each Runge-Kutta stage or only once at the end of the time step.

The central scheme of Equation (2.60) is characterised by a stencil of  $2N + 1$  nodes and the main problem of its implementation regards how to treat the points located near the boundaries. Note that, in this implementation, the boundary points are filtered only in the transverse directions and not in the direction normal to the boundary surface. Two approaches can be used to solve this problem: the application to the near-boundary points of high-order one-sided (HOOS) filters, or the application of lower-order centred (LOC) filters. The former solution is described by Gaitonde & Visbal [1999]. The formulation of the  $10^{\text{th}}$ -order HOOS filters applied to a boundary with node index

## 2. NUMERICAL METHOD

---

$i = 1$  is, for example,

$$\alpha_f \hat{\mathbf{Q}}_{i-1} + \hat{\mathbf{Q}}_i + \alpha_f \hat{\mathbf{Q}}_{i+1} = \sum_{n=1}^{11} a_{n,i} \mathbf{Q}_n, \quad (2.62)$$

where index  $i$  ranges from 2 to 5. As for the centred filter of Equation (2.60), the coefficients  $a_{n,i}$  are expressed as a function of the free parameter  $\alpha_f$  and were obtained using a similar procedure [Gaitonde & Visbal, 1999].

The spectral function  $SF$  of the filters defined by Equation (2.62) is [Gaitonde & Visbal, 1999]

$$SF(\omega) = \frac{\sum_{n=1}^{11} a_{n,i} \left\{ \cos [(n-i)\omega] + \sqrt{-1} \sin [(n-i)\omega] \right\}}{1 + 2\alpha_f \cos(\omega)}, \quad (2.63)$$

where, again,  $i$  ranges from 2 to 5. Equation (2.62) is asymmetrical, and due to this, the spectral function  $SF(\omega)$  has a non-zero imaginary part. In a range of the angular frequency  $\omega$  that depends on the order of the HOOS filter, the real part of function  $SF$  is larger than 1 and, although these filters are only locally applied and not extended to the whole computational field, this can cause the amplification of instabilities and lead to the divergence of the predicted solution. All the coefficients for the application of HOOS filters, including odd-order filters and the filters for the nodes located on the boundary (here not used) can be found in Gaitonde & Visbal [1998].

The second solution for the filtering of the nodes near the boundaries consists in the LOC method for which, approaching the boundary, the stencil size and the order of the filter is reduced while the centred formulation is retained. Their formulation also follows Equation (2.60) with  $N = 1$  on the first point in the flow field from the boundary,  $N = 2$  on the second, and so on. The spectral function  $SF$  of these filters

## 2. NUMERICAL METHOD

---

is, again, given by Equation 2.61 with  $N$  related to the order of accuracy of the filter applied to the specific node considered. This solution ensures that  $SF$  is always real and that the filters are non-dispersive. This makes the LOC approach more robust than the HOOS. The LOC is easy to implement but has the drawback of lowering the order of accuracy of the filter. This aspect can become the dominant factor on the accuracy of the results for some types of problems. Further details about this technique along with the list of the coefficients for different orders of accuracy are given by Visbal & Gaitonde [1998].

It is worth to notice that with both the LOC and the HOOS methods, the free coefficient  $\alpha_f$  applied to the near-boundary nodes is not restrained to the general value adopted for the treatment of the internal field, and near the boundaries it can be varied giving a further degree of freedom to the user. In particular, for the case of the LOC technique, this property can be used to decrease the impact of the adoption of the low-order stencils on the accuracy of the flow solution by increasing the local value of  $\alpha_f$  when approaching the boundary. This technique, proposed by Visbal & Gaitonde [1998] and commonly found in the literature (see, for example, Ragab & Salem-Said [2007]), has also been applied to some of the test cases shown in Chapter 3.

In the high-order prefactored solver used in this work, the filter defined analytically by Equation (2.60) is numerically implemented under three forms: an *explicit* form where Equation (2.60) is simplified by setting  $\alpha_f = 0$ , a *prefactored* form following Hixon [1999] that applies to the high-order filter the same technique developed for the spatial discretisation scheme discussed in Section 2.2.1, and a *standard implicit* treatment requiring the inversion of the matrix containing the  $\alpha_f$  coefficients on the left-hand side of Equation (2.60).

## 2. NUMERICAL METHOD

---

In the course of this work both the explicit and the standard implicit numerical implementations have been used for the solution of the test cases of Chapter 3. Unfortunately, the implicit prefactored implementation was adopted only for preliminary computations whose results are not shown in this report. This is due to the fact that this particular technique, although very efficient, has a strong limitation that currently does not allow the use of variable  $\alpha_f$  strategy near the boundary owing to the complexity of its numerical implementation. The application of the explicit formulation, due to its higher dissipation with respect to the implicit version, has to be restrained to problems where the numerical instabilities can not be overcome with more conservative methods.

### 2.3 Time-marching scheme

The numerical time-integration of a finite-difference scheme can be executed in implicit or explicit form. A popular choice for its characteristics of stability and for the optimised use of the computational memory is the explicit Runge-Kutta class of schemes.

Consider a general semi-discrete equation like

$$\frac{\partial \mathbf{Q}}{\partial t} = F(\mathbf{Q}), \quad (2.64)$$

where, in the case of the Euler equations in primitive form,  $\mathbf{Q}$  is the vector of the flow solution and  $F$  contains the discretisation of the spatial derivatives (the time dependence is omitted). The generic algorithm of an explicit Runge-Kutta scheme with  $s$

## 2. NUMERICAL METHOD

---

stages advancing time from  $t_n$  to  $t_n + \Delta t$  is

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + \sum_{i=1}^s w_i \mathbf{K}_i, \quad (2.65)$$

where

$$\mathbf{K}_i = \Delta t F \left( \mathbf{Q}^n + \sum_{j=1}^{i-1} \beta_{ij} \mathbf{K}_j \right). \quad (2.66)$$

The sets of constant coefficients  $w_i$  and  $\beta_{ij}$  characterise the scheme.

Following Hu *et al.* [1996] the analysis of the stability and of the accuracy of the general scheme shown above can be made on the simple advection wave Equation (2.33) having a solution of the form of Equation (2.34).

Fourier transforming Equation (2.65) and computing the ratio of the results between two successive steps  $n$  and  $n + 1$ , leads to the value of  $r$ , the *numerical* amplification factor

$$r = \frac{\tilde{\mathbf{Q}}_k^{n+1}}{\tilde{\mathbf{Q}}_k^n} = 1 + \sum_{j=1}^s c_j (-ick^* \Delta t)^j, \quad (2.67)$$

where  $\tilde{\mathbf{Q}}_k^n$  is the spatial Fourier transform of  $\mathbf{Q}^n$ ,  $c_j$  are coefficients related to  $w_i$  and  $\beta_{ij}$  of Equations (2.65) and (2.66), and  $c$  and  $k^*$  are defined as in Section 2.2. The numerical amplification factor expresses, for each frequency (related to  $ck^* \Delta t$ ), if the particular time-marching scheme, characterised by coefficients  $c_j$ , tends to amplify or reduce the wave amplitude, and it is strictly related to its accuracy and stability. These can be studied by comparing the *numerical* with the *exact* amplification factor  $r_e$

$$r_e = e^{-ick^* \Delta t}, \quad (2.68)$$

of which Equation (2.67) can be seen as an approximation. Considering the Taylor

## 2. NUMERICAL METHOD

---

series expansion of  $r_e$  it becomes clear that the formal order of accuracy of the Runge-Kutta scheme is related to the number of the coefficients  $c_j$  of Equation (2.67) that match those of the series expansion. This observation also explains why the maximum order of accuracy of a scheme is related to the number of stages  $s$  that controls the number of available coefficients  $c_j$ .

The ratio of the amplification factors

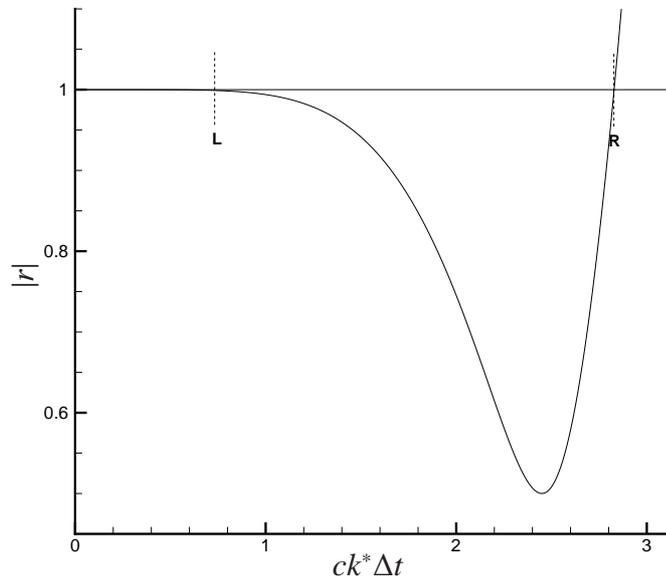
$$\frac{r}{r_e} = |r| e^{-i\delta}, \quad (2.69)$$

expresses the two components of error introduced by the numerical integration:  $|r|$  is the error of amplitude (related to the *dissipation* of the scheme), and  $\delta$  is the error of phase (related to the *dispersion* of the scheme). They are both functions of  $ck^*\Delta t$  and depend on the characteristics of the internal scheme. Note that  $|r|$  has to be compared to the theoretical value of 1. For instance, the classical four-stage fourth-order Runge-Kutta scheme has dissipation and dispersion characteristics as shown in Figure 2.2. The details of the figure also show the stability limit  $R$ , an absolute value related to the error amplitude that indicates the range of frequencies that are not amplified by the scheme ( $|r| \leq 1$ ), and the accuracy limits  $L_{diss}$  and  $L_{disp}$  related, respectively to the dissipation and to the dispersion of the time-integration scheme. As in the case of the spatial scheme, described in Section 2.2, these values depend on a formal maximum tolerable error in the propagation of the waves that can be fixed depending on the accuracy requirements of the computation. For example, Hu *et al.* [1996] set both the maximum difference between  $|r|$  and 1, and the maximum  $\delta$  to 0.001. The overall accuracy limit  $L$  of the scheme is given by the lower between  $L_{diss}$  and  $L_{disp}$ .

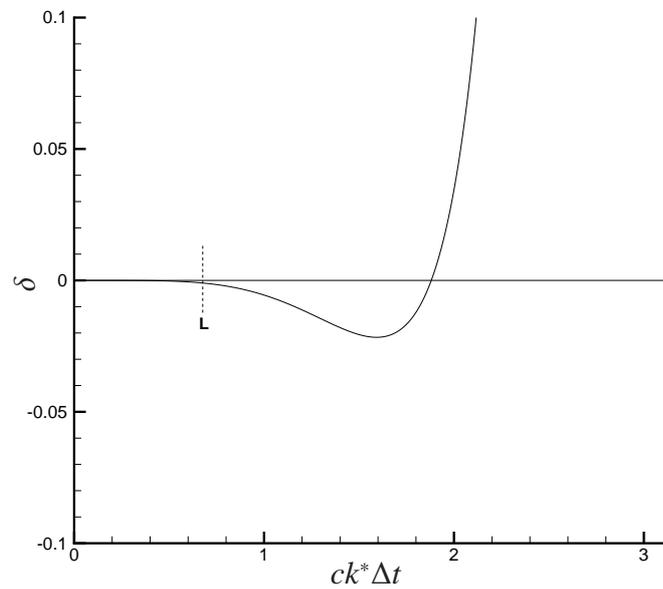
The Courant-Friedrichs-Lewy (CFL) condition for the particular association of the

## 2. NUMERICAL METHOD

---



(a) Amplitude error  $|r|$  (dissipation).



(b) Phase error  $\delta$  (dispersion).

Figure 2.2: Errors of the classical four-stage fourth-order Runge-Kutta scheme as functions of the effective wavenumber. Where appropriate, the stability limit  $R$  and the accuracy limit  $L$  are shown.

## 2. NUMERICAL METHOD

---

spatial and the time-integration scheme adopted can be found by relating  $L$  and  $R$  with the maximum resolvable wavenumber  $k_c^*$  and the maximum effective wavenumber  $k_{max}^*$  defined in Section 2.2 that are typical of the spatial scheme, in the form

$$c \frac{\Delta t}{\Delta x} \leq \min \left( \frac{L}{k_c^* \Delta x}, \frac{R}{k_{max}^* \Delta x} \right). \quad (2.70)$$

The maximum value for the time step  $\Delta t$  can be obtained from Equation (2.70) depending on the minimum value of the inter-nodal distance  $\Delta x$  in the grid and on the speed of sound  $c$ .

### 2.3.1 Low-storage Runge-Kutta schemes

The modern Runge-Kutta schemes used in association with finite-difference high-order spatial schemes, require two main characteristics: the lowest possible memory storage, and the highest stability or accuracy limits (or both at the same time). Essentially, four factors can be varied for this optimisation: the type of algorithm that is related to the memory requirements and to the efficiency, the number of steps and of stages of the Runge-Kutta scheme, and the set of coefficients employed that is related to both stability and accuracy.

The optimisation of the memory requirements of the explicit Runge-Kutta schemes has been an important issue since the birth of the digital computing machines as, at that time, the number of available registers was one of the main limiting factors of the computation. Gill [1951] gives a first version of a fourth-order algorithm of optimised 3N type, where N is the number of equations simultaneously solved, and 3 is the required number of levels of memory storage for each equation (that for the classical four stage fourth-order Runge-Kutta is equal to 4). The basic idea behind the optimisation, still at

## 2. NUMERICAL METHOD

---

the basis of the most recent methods, is to use, at each stage, the information contained in the memory registers without resetting them. Fyfe [1966] demonstrates that all the Runge-Kutta schemes can be written in  $3N$  form, and Williamson [1980] shows that all the second-order, many third-order and some fourth-order schemes can be arranged in  $2N$  form. The study of Williamson, along with that of van der Houwen [1972] who illustrated another type of  $2N$  algorithm, have remained for a long time the main reference for the development of new schemes.

Recently, Ketcheson [2010] has analysed in a rigorous mathematical form the available methods giving a new type of classification and introducing a more general class of low-storage algorithms. In addition, the author considers the memory requirements to provide an error estimate of the computation and the ability to restart of each type of algorithm.

In brief, Ketcheson found that the algorithm used by Williamson [1980] (called  $2N$ ) can be written in pseudo-code as

```
S1 := Q(n)
for i= 2:s+1 do
  S2 := A(i) S2 + Dt F(S1)
  S1 := S1 + B(i) S2
end
Q(n+1) = S1
```

where  $S1$  and  $S2$  are the two memory registers of size  $N$ ,  $s$  is the number of stages,  $Dt$  is the time step,  $F$  and  $Q$  are as defined for Equation (2.64),  $A(i)$  and  $B(i)$  are constant coefficients typical of the scheme, and the operator  $:=$  denotes the action of storing the right-hand side of the equation into the memory register on the left-hand side.

## 2. NUMERICAL METHOD

---

This algorithm can be implemented with only 2 levels of storage under the assumption that the assignment  $S2 := S2 + F(S1)$  does not require additional memory. The algorithms proposed by Carpenter & Kennedy [1994], Stanescu & Habashi [1998], Berland *et al.* [2006], and Allampalli *et al.* [2009] are of this type.

In contrast, the algorithm devised by van der Houwen [1972] (called 2R by Ketcheson to distinguish it from the one proposed by Williamson) in pseudo-code has the form

```
S2 := Q(n)
for i = 1:s do
  S1 := S2 + ( a(i,i-1) - b(i-1) ) Dt S1
  S1 := F(S1)
  S2 := S2 + b(i) Dt S1
end
Q(n+1) = S2
```

where  $a(i, j)$  and  $b(i-1)$  are the sets of constant coefficients typical of the scheme. The above algorithm can be found in the literature in various forms. This, proposed by Calvo *et al.* [2003, 2004], is one of the most simple. Those of Kennedy *et al.* [2000], and Hu *et al.* [1996] are equivalent. In this case the algorithm can be implemented with only 2 levels of storage under the assumption that the assignment  $S1 := F(S1)$  does not require additional memory.

Ketcheson found that both the above algorithms are actually special cases of

```
S2 := 0
S1 := Q(n)
for i = 2:s+1 do
```

## 2. NUMERICAL METHOD

---

```
S2 := S2 + d(i-1) S1
S1 := g1(i) S1 + g2(i) S2 + b(i, i-1) Dt F(S1)
end
Q(n+1) = S1
```

This algorithm is called 2S and  $g1(i)$ ,  $g2(i)$ ,  $d(i)$ , and  $b(i, j)$  are its constant coefficients. Due to the higher number of available coefficients with respect to the 2N and 2R cases, this class of algorithms allows to write fifth- and sixth-order schemes in low-storage form and also allows some four-stage fourth-order schemes that are not possible in 2N or 2R form.

### 2.3.2 Optimised Runge-Kutta schemes

In the last 20 years the development of the explicit Runge-Kutta schemes has been driven by the research in the aeroacoustic field in order to find methods optimised for the wave propagation. The optimisation of a scheme is essentially based on the variation of its number of steps and of stages, and on the set of constant coefficients, in order to extend either the limit of accuracy  $L$  or that of stability  $R$ , or of both at the same time.

An important contribution in this sense is given by Hu *et al.* [1996] with their Low-Dissipation and Low-Dispersion Runge-Kutta (LDDRK) schemes that are the first example of optimisation for both dissipation and dispersion errors. Furthermore, they propose two-step schemes that, alternating the number of stages and the set of coefficients used, can largely improve the characteristics of the method. The authors base their study on the minimisation of  $|r - r_e|^2$  (see Equation (2.67) and (2.68) for the definition of  $r$  and  $r_e$ ) that implies the minimisation of the errors. The single-step

## 2. NUMERICAL METHOD

---

schemes proposed with 4 and 5 stages are second-order accurate (referred to as *RK24* and *RK25* in Table 2.3), and that with 6 stages (referred to as *RK46* in Table 2.3) is fourth-order. Both the two-step methods devised (referred to as *RK4-6* and *RK5-6* in Table 2.4) are fourth-order and alternate a step of 6 stages with one of 4 or 5 stages. In the article by Hu *et al.* [1996] the low-storage implementation is discussed and a 3N algorithm is provided.

Stanescu & Habashi [1998] start from the work by Hu noting that with the given 3N implementation, the fourth-order schemes turn second-order for non-linear problems. They provide the coefficients for all the schemes proposed by Hu to be used in association with a 2N algorithm that preserves the fourth-order also in the non-linear case.

The most comprehensive work on optimised schemes so far is given by Kennedy *et al.* [2000] that analyse the performance of a large number of Runge-Kutta schemes with various levels of memory optimisation applied to the solution of the complete Navier-Stokes equations.

Bogey & Bailly [2004] concentrate on uniform and slowly non-uniform grids optimising at the same time the spatial scheme, the time-integration and the filtering, all in the same range of wavenumbers. The two time-marching schemes proposed are second-order accurate and require 5 and 6 stages.

The work by Calvo *et al.* [2004] also aims to optimise various aspects of the wave propagation adopting a new optimisation function that tries to blend “good stability properties, high order, low dissipation and dispersion and low storage use”. The authors obtain a fourth-order six-stage scheme (referred to as *new LDD46* in Table 2.3) with characteristics, in most cases, superior to those proposed by Hu *et al.* [1996].

Very similar results to those of Calvo are obtained by Berland *et al.* [2006] using

## 2. NUMERICAL METHOD

---

a different approach in the optimisation. In this case the fourth-order accuracy is retained even for non-linear problems adopting a 2N low-storage algorithm against the 2R algorithm of Calvo. The accuracy and stability limits of the fourth-order accurate six-stage scheme are shown in Table 2.3 where it is referred to as *RK46-L/NL*.

Finally, the work of Allampalli *et al.* [2009] extends the range of available schemes with a different approach. In this case the optimisation function of the High-Accuracy Large-step Explicit Runge-Kutta schemes (HALE-RK) tends to maximise the stability limits rather than the accuracy. This type of schemes can be useful with problems in which the length of the propagating waves is large relatively to the smallest cell of the grid. For these problems the accuracy optimisation over a large range of frequencies results in an unnecessary limitation of the time step. The resulting one- and two-step methods having 6 or 7 stages are all fourth-order accurate. The characteristics of accuracy and stability of the two-step scheme (referred to as *RK6-7*) are shown in Table 2.4.

The finite-difference compact solver used for this study implements a selection of five of the above discussed optimised explicit Runge-Kutta time-marching schemes in low-storage form: (i) the classical fourth-order Runge-Kutta scheme, here implemented in 3N form, (ii) Hu's fourth-order 4-6 alternate Low-Dissipation Low-Dispersion (LDDRK46) (referred to as *RK4-6* in the tables and the figures), (iii) Berland's fourth-order six-stage Low-Dissipation Low-Dispersion (LDDRK46-NL) (referred to as *RK46-L/NL* in the tables and the figures), (iv) Calvo's fourth-order six-stage Low-Dissipation Low-Dispersion (new 2N LDD46) (referred to as *new LDD46* in the tables and the figures), and (v) Allampalli's two-step alternated (6-7) fourth-order High-Accuracy Large step Explicit RK (HALE-RK) (referred to as *RK6-7* in the tables and the figures). In most cases the sets of constant coefficients for the various Runge-Kutta schemes are

## 2. NUMERICAL METHOD

---

obtained by the authors with techniques of numerical optimisation that are difficult to reproduce, so their values have been taken directly from the original publications and are not shown here.

It has to be stressed that the solver produced in this work is compatible with both 2N and 2R types of algorithm. Therefore, the implementation of other schemes that can be written in these forms requires little effort.

The  $c_j$  coefficients of Equation (2.67) along with the accuracy and stability limits of a selection of the schemes here discussed, are shown in Table 2.3 for the one-step methods, and in Table 2.4 for the two-step methods. In addition, in Figures 2.3 and 2.4 the values of their error of amplitude  $|r|$  and of phase  $\delta$  as function of  $ck^*\Delta t$  are compared.

### 2.4 Curvilinear coordinates

The finite-difference compact scheme described in Section 2.2 implicitly requires to be applied to uniform grids having nodes equally spaced in every spatial direction. To solve problems involving complex geometrical shapes, non-uniform curvilinear grids must be adopted. In order to overcome this limitation it is necessary to establish a transformation between the actual, non-uniform grid in the physical space defined by coordinates  $x$ ,  $y$ , and  $z$ , and a topologically equivalent uniform grid in the computational space defined by curvilinear coordinates  $\xi$ ,  $\eta$ , and  $\zeta$ . This transformation can be exemplified in 2D as in Figure 2.5 where at each node of the actual grid on the left-hand side corresponds a node on the uniform grid on the right-hand side.

Table 2.3: Coefficients  $c_j$  of Equation (2.67) of the classical fourth-order explicit Runge-Kutta scheme and of various second- and fourth-order single-step explicit Runge-Kutta optimised schemes. The values of the dissipation  $L_{diss}$  and dispersion  $L_{disp}$  accuracy limits for different levels (field “Max error”) of maximum tolerable error are also shown along with the stability  $R$  limit. [ppw] indicates the values expressed in points per wavelength.

Coefficient / limit [nd]	Max error [nd]	Classical	Hu		Bogey	Hu	Berland	Calvo
		RK4	RK24	RK25	RK26	RK46	RK46-L/NL	new LDD46
$c_1$		1	1	1	1	1	1	1
$c_2$		1/2	0.5	0.5	0.5	1/2	1/2	1/2
$c_3$		1/6	0.162997	0.166558	0.165919771368	1/6	1/6	1/6
$c_4$		1/24	0.0407574	0.0395041	0.040919732041	1/24	1/24	1/24
$c_5$		-	-	0.00781071	0.007555704391	0.00781005	0.007856772044	0.007853
$c_6$		-	-	-	0.000891421261	0.00132141	0.000959998595	0.000948
$L_{diss}$	0.002	0.824624	-	-	-	-	-	2.106869
	0.00125	-	-	-	-	-	-	2.031125
	0.001	0.732341	0.849090	1.720940	2.003248	1.750970	2.058761	1.996937
	0.0005	0.732340	0.849096	1.673345	1.909166	1.709902	1.969950	1.896997
$L_{diss}$ [ppw]	0.0005	8.579601	7.399853	3.754865	3.291063	3.674588	3.189515	3.312175
$L_{disp}$	0.002	0.788377	-	-	-	-	-	1.299215
	0.0005 $\pi$	0.747340	0.911251	1.384397	1.530614	2.009860	1.248420	1.242107
	0.00125	-	-	-	-	-	-	1.189958
	0.001	0.677373	0.8562	1.346950	1.468801	1.966030	1.146278	1.140756
$L_{disp}$ [ppw]	0.0005 $\pi$	8.407399	6.895120	4.538572	4.105010	3.126181	5.032910	5.058490
$R$		2.828427	2.8519	3.522508	3.942836	1.750970	3.815986	3.826039
$R$ [ppw]		2.221442	2.203158	1.783725	1.593570	3.588403	1.646543	1.642217

## 2. NUMERICAL METHOD

---

Table 2.4: Coefficients  $c_j$  of Equation (2.67) of three fourth-order two-step explicit Runge-Kutta optimised schemes. The values of the dissipation  $L_{diss}$  and dispersion  $L_{disp}$  accuracy limits for different levels (field “Max error”) of maximum tolerable error are also shown along with the stability  $R$  limit. [ppw] indicates the values expressed in points per wavelength.

Coefficient / limit [nd]	Max error [nd]	Hu		Allampalli
		RK4-6	RK5-6	RK6-7
$c_1(1)$		1	1	1
$c_2(1)$		1/2	1/2	1/2
$c_3(1)$		1/6	1/6	1/6
$c_4(1)$		1/24	1/24	1/24
$c_5(1)$		0	0.0036105	0.005095336935
$c_6(1)$		0	0	0.000923040959
$c_7(1)$		0	0	0
$c_1(2)$		1	1	1
$c_2(2)$		1/2	1/2	1/2
$c_3(2)$		1/6	1/6	1/6
$c_4(2)$		1/24	1/24	1/24
$c_5(2)$		0.0162098	0.0121101	0.008223191190
$c_6(2)$		0.00286365	0.00285919	0.000871526244
$c_7(2)$		0	0	0.000125766378
$L_{diss}$	0.002	1.703043	2.052337	1.120339
$L_{diss}$	0.001	1.637163	2.002488	0.992488
$L_{diss}$	0.0005	1.581532	1.962898	0.880336
$L_{diss}$ [ppw]	0.0005	3.972847	3.200974	7.137254
$L_{disp}$	0.002	1.886675	2.242957	1.122240
$L_{disp}$	0.0005 $\pi$	1.871215	2.230987	1.059158
$L_{disp}$	0.001	1.845561	2.211651	0.954030
$L_{disp}$ [ppw]	0.0005 $\pi$	3.357810	2.816325	5.932247
$R$		2.519454	2.843538	5.529152
$R$ [ppw]		2.493868	2.209636	1.136374

## 2. NUMERICAL METHOD

---

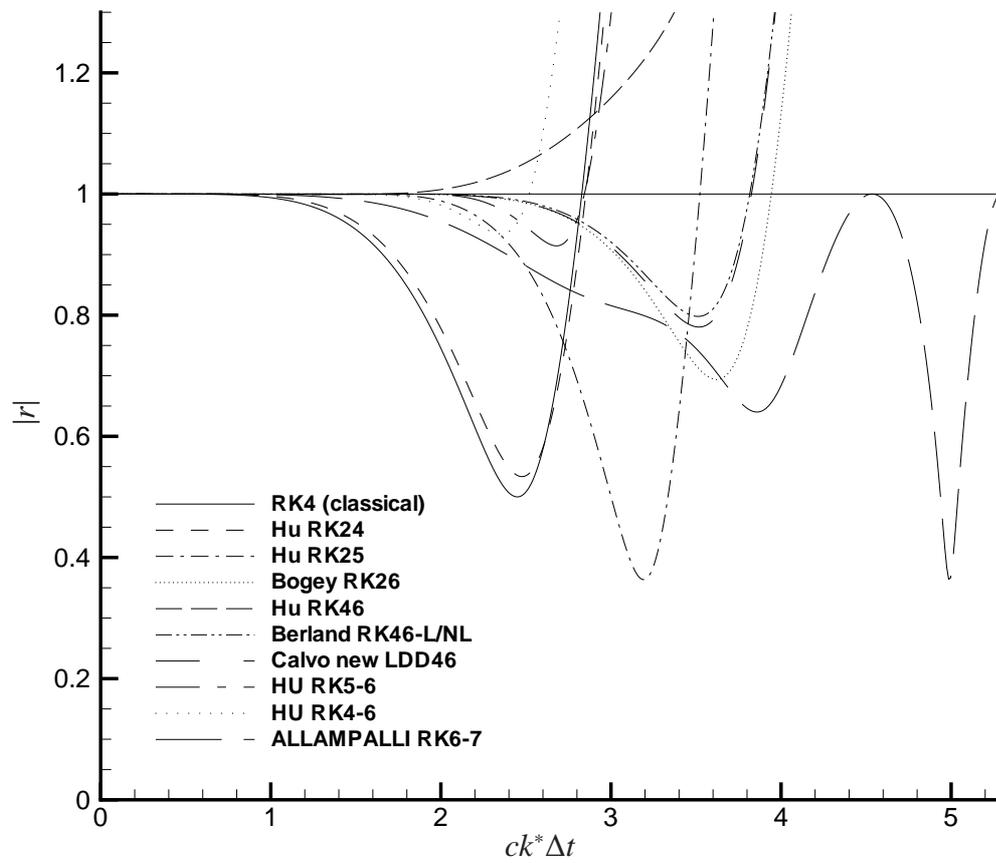


Figure 2.3: Amplitude error  $|r|$  (related to the dissipation) as function of  $ck^*\Delta t$  of a selection of optimised one- and two-step multi-stage explicit Runge-Kutta schemes.

## 2. NUMERICAL METHOD

---

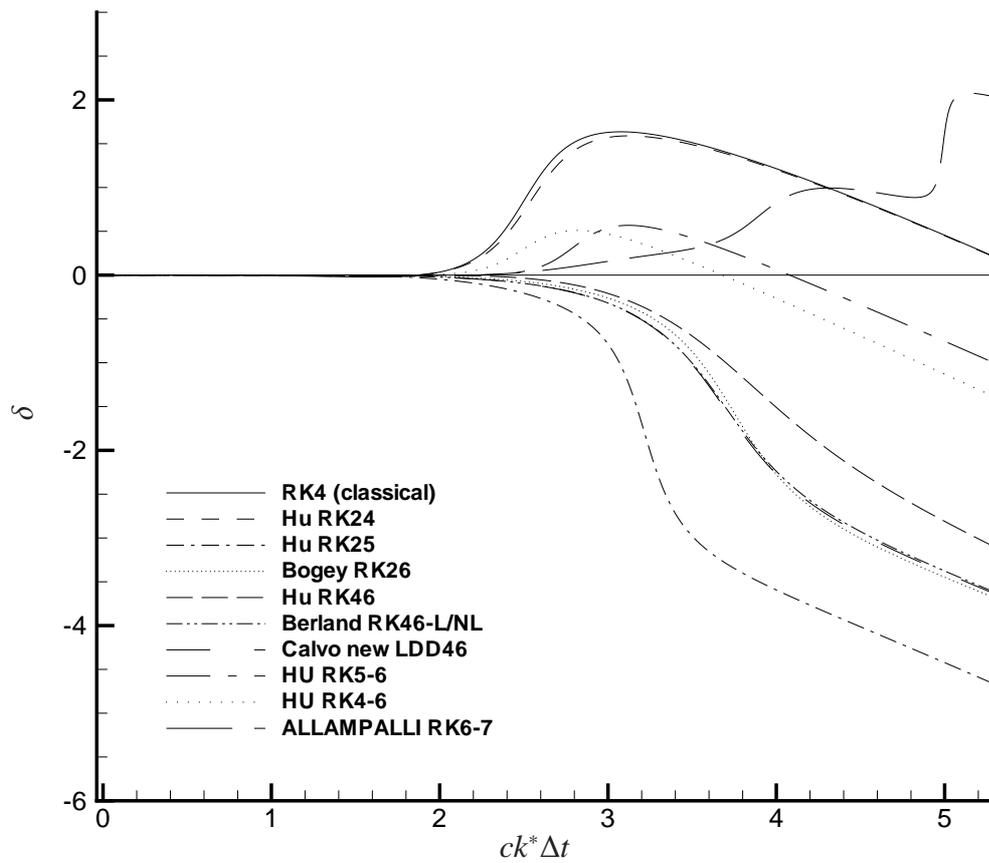


Figure 2.4: Phase error  $\delta$  (related to the dispersion) as function of  $ck^* \Delta t$  of a selection of optimised one- and two-step multi-stage explicit Runge-Kutta schemes.

## 2. NUMERICAL METHOD

---

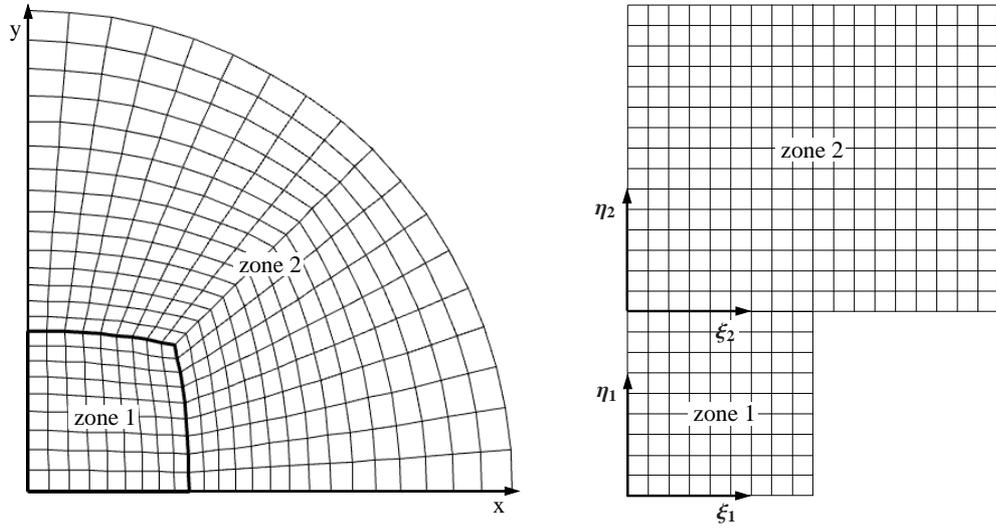


Figure 2.5: An example of physical non-uniform mesh (left-hand side) and its uniform computational counterpart (right-hand side).

The transformation is defined by

$$\xi = \xi(x, y, z), \quad (2.71a)$$

$$\eta = \eta(x, y, z), \quad (2.71b)$$

$$\zeta = \zeta(x, y, z). \quad (2.71c)$$

Note that in this brief treatment no time dependence is considered as the grids are taken as static and the time in the computational space  $\tau$  is always equal to the time  $t$  in the physical space.

The derivatives of the curvilinear coordinates with respect to the Cartesian coordi-

## 2. NUMERICAL METHOD

---

nates, called *metrics*

$$\xi_x = \frac{\partial \xi}{\partial x}, \quad \xi_y = \frac{\partial \xi}{\partial y}, \quad \xi_z = \frac{\partial \xi}{\partial z}, \quad (2.72a)$$

$$\eta_x = \frac{\partial \eta}{\partial x}, \quad \eta_y = \frac{\partial \eta}{\partial y}, \quad \eta_z = \frac{\partial \eta}{\partial z}, \quad (2.72b)$$

$$\zeta_x = \frac{\partial \zeta}{\partial x}, \quad \zeta_y = \frac{\partial \zeta}{\partial y}, \quad \zeta_z = \frac{\partial \zeta}{\partial z}, \quad (2.72c)$$

can be determined either analytically (when Equation (2.71) is in turn known in analytical form) or, more generally, numerically, as in this work, using a method capable to ensure an adequate level of accuracy. As suggested by Visbal & Gaitonde [1998], for their computation it is correct to use the same spatial scheme used for the flow derivatives.

Starting from the Euler equations in conservation form of Equation (2.5) and using the chain rule of differential calculus

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} + \left( \xi_x \frac{\partial \mathbf{E}}{\partial \xi} + \eta_x \frac{\partial \mathbf{E}}{\partial \eta} + \zeta_x \frac{\partial \mathbf{E}}{\partial \zeta} \right) + \left( \xi_y \frac{\partial \mathbf{F}}{\partial \xi} + \eta_y \frac{\partial \mathbf{F}}{\partial \eta} + \zeta_y \frac{\partial \mathbf{F}}{\partial \zeta} \right) \\ + \left( \xi_z \frac{\partial \mathbf{G}}{\partial \xi} + \eta_z \frac{\partial \mathbf{G}}{\partial \eta} + \zeta_z \frac{\partial \mathbf{G}}{\partial \zeta} \right) = \mathbf{S}. \end{aligned} \quad (2.73)$$

Then, by regrouping, dividing by  $J$ , and after some manipulation Hirsch [1990] the form is obtained

$$\begin{aligned} \frac{\partial}{\partial t} \left( \frac{\mathbf{U}}{J} \right) + \frac{\partial}{\partial \xi} \left[ \frac{1}{J} (\xi_x \mathbf{E} + \xi_y \mathbf{F} + \xi_z \mathbf{G}) \right] + \frac{\partial}{\partial \eta} \left[ \frac{1}{J} (\eta_x \mathbf{E} + \eta_y \mathbf{F} + \eta_z \mathbf{G}) \right] \\ + \frac{\partial}{\partial \zeta} \left[ \frac{1}{J} (\zeta_x \mathbf{E} + \zeta_y \mathbf{F} + \zeta_z \mathbf{G}) \right] = \frac{\mathbf{S}}{J}, \end{aligned} \quad (2.74)$$

## 2. NUMERICAL METHOD

---

where  $J$  is the Jacobian determinant

$$J = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{vmatrix}^{-1}. \quad (2.75)$$

By defining

$$\tilde{\mathbf{U}} = \frac{\mathbf{U}}{J}, \quad (2.76a)$$

$$\tilde{\mathbf{E}} = \frac{1}{J} (\xi_x \mathbf{E} + \xi_y \mathbf{F} + \xi_z \mathbf{G}), \quad (2.76b)$$

$$\tilde{\mathbf{F}} = \frac{1}{J} (\eta_x \mathbf{E} + \eta_y \mathbf{F} + \eta_z \mathbf{G}), \quad (2.76c)$$

$$\tilde{\mathbf{G}} = \frac{1}{J} (\zeta_x \mathbf{E} + \zeta_y \mathbf{F} + \zeta_z \mathbf{G}), \quad (2.76d)$$

$$\tilde{\mathbf{S}} = \frac{\mathbf{S}}{J}, \quad (2.76e)$$

the strong conservative form is recovered over a uniform orthogonal grid

$$\frac{\partial \tilde{\mathbf{U}}}{\partial t} + \frac{\partial \tilde{\mathbf{E}}}{\partial \xi} + \frac{\partial \tilde{\mathbf{F}}}{\partial \eta} + \frac{\partial \tilde{\mathbf{G}}}{\partial \zeta} = \tilde{\mathbf{S}}. \quad (2.77)$$

## 2.5 Boundary conditions

### 2.5.1 Kim and Lee characteristic generalised boundary conditions

In their series of articles Kim & Lee [2000, 2003, 2004] propose a framework for the implementation of generalised characteristic boundary conditions in conservative form for the Navier-Stokes equations. They include: soft inflow condition, non-reflecting outflow, pulsating inviscid wall, and a multi-block interface condition.

## 2. NUMERICAL METHOD

---

All of these conditions have been implemented in inviscid form in the high-order finite-difference solver object of this work and the general framework has also been used, with small modifications, to host the subsonic inflow and outflow characteristic-based boundary conditions by Giles [1990] discussed in Section 2.5.2.

The treatment of Kim and Lee strongly relies on the theory given by Hirsch [1990] briefly illustrated in this work in Sections 2.1.2 and 2.1.3.

Starting from the flow equations in the strong conservative form of Equation (2.77), and applying the transformations described in Section 2.1.3 between the conservative and the characteristic variable form, Kim and Lee obtain

$$\frac{\partial \mathbf{R}}{\partial t} + \boldsymbol{\ell} = \mathbf{S}_c, \quad (2.78)$$

where  $\mathbf{R}$  is the vector of the characteristic variables defined in Equation (2.22),  $\boldsymbol{\ell}$  is the vector of the convective terms, and vector  $\mathbf{S}_c$  includes all the remaining terms. For example, in the case of a boundary located in correspondence to the curvilinear coordinate  $\xi = 0$

$$\boldsymbol{\ell} \equiv \boldsymbol{\Lambda} \frac{\partial \mathbf{R}}{\partial \xi} = \mathbf{P}^{-1} \left( \xi_x \frac{\partial \mathbf{E}}{\partial \xi} + \xi_y \frac{\partial \mathbf{F}}{\partial \xi} + \xi_z \frac{\partial \mathbf{G}}{\partial \xi} \right), \quad (2.79a)$$

$$\mathbf{S}_c = J \mathbf{P}^{-1} \left\{ \tilde{\mathbf{S}} - \left[ \mathbf{E} \frac{\partial}{\partial \xi} \left( \frac{\xi_x}{J} \right) + \mathbf{F} \frac{\partial}{\partial \xi} \left( \frac{\xi_y}{J} \right) + \mathbf{G} \frac{\partial}{\partial \xi} \left( \frac{\xi_z}{J} \right) + \frac{\partial \tilde{\mathbf{F}}}{\partial \eta} + \frac{\partial \tilde{\mathbf{G}}}{\partial \zeta} \right] \right\}, \quad (2.79b)$$

where  $\boldsymbol{\Lambda} = [\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5]^T$  is the vector of the eigenvalues shown in Equation (2.20) and matrix  $\mathbf{P}^{-1}$  defines the transformation between the conservative and the characteristic variables of Equation (2.29). The novelty of the treatment by Kim and Lee consists in the conservation of all the terms contained in  $\mathbf{S}_c$  and their use in the enforcement of the boundary conditions without introducing any approximation. For the wall bound-

## 2. NUMERICAL METHOD

---

aries, this idea, originally proposed by Lockard & Morris [1998] but not completely developed, can be seen as a correction of the local one-dimensional inviscid (LODI) relations used by Thompson [1987, 1990] and Poinso & Lele [1992] for their treatments. A resulting “modified” LODI system in primitive form, including the source terms of vector  $\mathbf{S}_c$ , that can be used for the definition of the boundary conditions, is derived from Equation (2.78) as

$$\frac{\partial \rho}{\partial t} + \ell_1 + \frac{\rho}{2c} (\ell_4 + \ell_5) = \mathbf{S}_{c1} + \frac{\rho}{2c} (\mathbf{S}_{c4} + \mathbf{S}_{c5}), \quad (2.80a)$$

$$\frac{\partial \tilde{U}}{\partial t} + \frac{1}{2} (\ell_4 - \ell_5) = \frac{1}{2} (\mathbf{S}_{c4} - \mathbf{S}_{c5}), \quad (2.80b)$$

$$\frac{\partial \tilde{V}}{\partial t} + \ell_3 = \mathbf{S}_{c3}, \quad (2.80c)$$

$$\frac{\partial \tilde{W}}{\partial t} + \ell_2 = \mathbf{S}_{c2}, \quad (2.80d)$$

$$\frac{\partial p}{\partial t} + \frac{\rho c}{2} (\ell_4 + \ell_5) = \frac{\rho c}{2} (\mathbf{S}_{c4} + \mathbf{S}_{c5}), \quad (2.80e)$$

where  $c$  is the speed of sound and  $\tilde{U}$ ,  $\tilde{V}$ , and  $\tilde{W}$  are, respectively, the contravariant velocity normal to the boundary located at  $\xi = 0$  and its components parallel to the plane [Hung, 2002].

The general procedure for the enforcement of all the characteristic-based boundary conditions proposed by Kim and Lee require the following steps (always referring to the boundary at  $\xi = 0$ ):

1. The computation of the flux derivatives along the curvilinear coordinates  $\eta$  and  $\zeta$  using the internal spatial scheme. In the  $\xi$  direction the flux derivatives are computed using the high-order approximation based on one-sided stencils (in this study the explicit 7-point stencil defined in Section 2.2.2). The value of the normal flux derivative is used as a first guess that will be corrected in the next

## 2. NUMERICAL METHOD

---

steps.

2. A first guess for vector  $\boldsymbol{\ell}$  containing the variation of the amplitude of the incoming and outgoing waves is obtained using the following expression that can be easily derived combining Equations (2.76b) and (2.79a)

$$\boldsymbol{\ell} = \mathbf{J}\mathbf{P}^{-1} \left\{ \frac{\partial \tilde{\mathbf{E}}}{\partial \xi} - \left[ \mathbf{E} \frac{\partial}{\partial \xi} \left( \frac{\xi_x}{J} \right) + \mathbf{F} \frac{\partial}{\partial \xi} \left( \frac{\xi_y}{J} \right) + \mathbf{G} \frac{\partial}{\partial \xi} \left( \frac{\xi_z}{J} \right) \right] \right\}. \quad (2.81)$$

3. The values of  $\boldsymbol{\ell}$  are modified by imposing the physical boundary condition (described in the following chapters) obtaining the corrected form  $\boldsymbol{\ell}^*$ .
4. The corrected form of the normal flux derivatives are recovered from  $\boldsymbol{\ell}^*$  by using Equation (2.81)

$$\left( \frac{\partial \tilde{\mathbf{E}}}{\partial \xi} \right)^* = \frac{1}{J} \mathbf{P} \boldsymbol{\ell}^* + \left[ \mathbf{E} \frac{\partial}{\partial \xi} \left( \frac{\xi_x}{J} \right) + \mathbf{F} \frac{\partial}{\partial \xi} \left( \frac{\xi_y}{J} \right) + \mathbf{G} \frac{\partial}{\partial \xi} \left( \frac{\xi_z}{J} \right) \right]. \quad (2.82)$$

Finally, the corrected value of the normal flux derivative can either be used to compute the time derivative at the boundary or, as for the implementation in this work, as the starting value for the sweep along the coordinate normal to the boundary.

### 2.5.1.1 Inflow/outflow

In their article Kim & Lee [2000] define non-reflecting inflow and outflow conditions inspired by the non-reflecting outflow proposed by Poinso & Lele [1992] that, in turn, base their treatment on an idea by Rudy & Strikwerda [1980] applied only to the energy equation. The conditions are enforced on the primitive form of the variables. They are loosely related to a fixed reference state representing the condition of the flow far beyond the boundary. This technique allows to maintain a mean reference state. The

## 2. NUMERICAL METHOD

---

partial reflectivity of the condition depends on a coefficient  $K$  related to the speed of sound  $c$ , to a characteristic length of the domain  $l$ , to the maximum Mach number on the whole computational field  $M_{max}$ , and to a coefficient  $\sigma$  as

$$K = \sigma \left(1 - M_{max}^2\right) \left(\frac{c}{l}\right). \quad (2.83)$$

Note that the correct value for  $\sigma$  depends on the particular problem and should be determined through tests. However, Kim and Lee recommend the value 0.25 that in their experiments allows at the same time to keep the mean flow at the desired value and to avoid most of the reflections.

The number of conditions that need to be imposed is related to the direction of propagation of the characteristic waves that, when outgoing, are computed from the state of the flow while, when incoming, are determined imposing a relation on the corresponding value of  $\ell$ . The corrected form is labelled as  $\ell^*$ .

For a 3D subsonic outflow located at  $\xi = \xi_{max}$  as on the right-hand side of Figure 2.6 (a), only one wave, the acoustic pressure wave defined by Equation (2.22e), is entering the computational domain and its value is specified by the relation

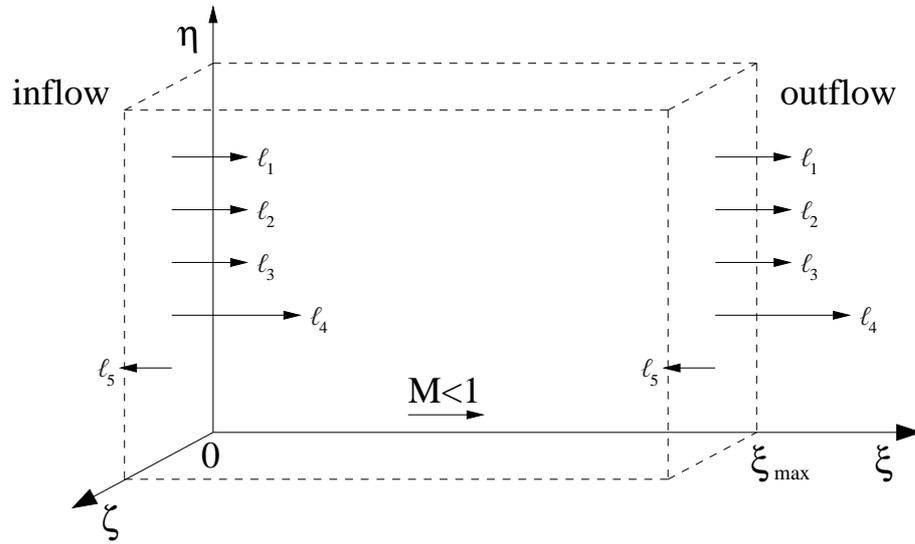
$$\ell_5^* = K \left(\frac{p - p_\infty}{\rho c}\right), \quad (2.84)$$

where  $K$  is as by Equation (2.83),  $p$  and  $\rho$  are the static pressure and density in that point,  $c$  is the speed of sound, and  $p_\infty$  is the value of the pressure at a point downstream of the boundary where the flow state is considered stationary.

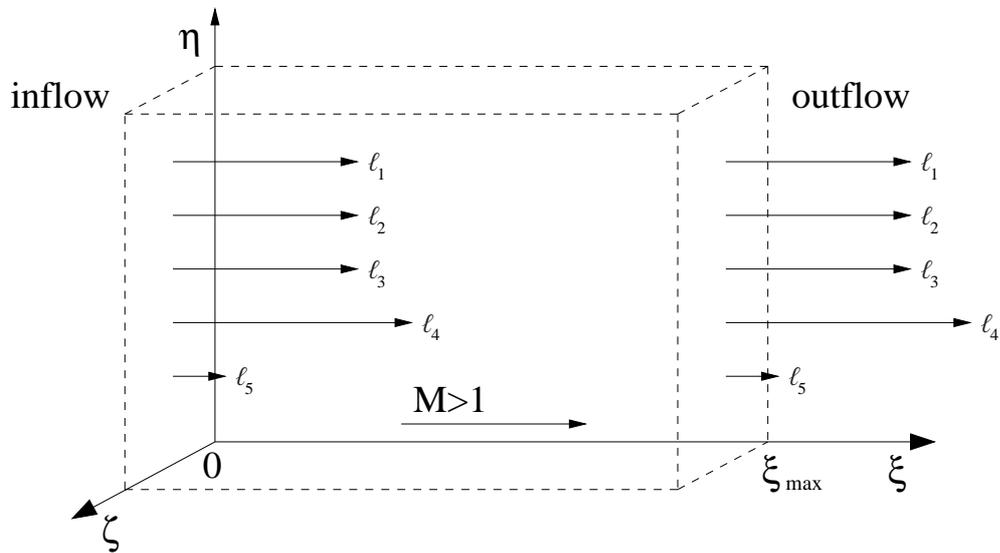
When the outflow is supersonic, as on the right-hand side of Figure 2.6 (b), all the

## 2. NUMERICAL METHOD

---



(a) Case of a subsonic mean flow.



(b) Case of a supersonic mean flow.

Figure 2.6: Waves leaving and entering the computational domain through an inlet plane at  $\xi = 0$  and an outlet plane at  $\xi = \xi_{max}$  in case of subsonic or supersonic mean flow.

## 2. NUMERICAL METHOD

---

waves are outgoing and there is no need for any condition as the state on the boundary is completely determined by the information coming from the flow field.

For a 3D subsonic inflow boundary located at  $\xi = 0$  as on the left-hand side of Figure 2.6 (a), four waves are entering the computational field and the following conditions are imposed on the characteristic waves defined by Equations from (2.22a) to (2.22d)

$$\ell_1^* = 0, \quad (2.85a)$$

$$\ell_2^* = K \left( \frac{\tilde{W} - \tilde{W}_\infty}{2} \right), \quad (2.85b)$$

$$\ell_3^* = K \left( \frac{\tilde{V} - \tilde{V}_\infty}{2} \right), \quad (2.85c)$$

$$\ell_4^* = K \left( \tilde{U} - \tilde{U}_\infty + \frac{P - P_\infty}{\rho c} \right), \quad (2.85d)$$

where the variables with subscript  $\infty$  refer to a stationary state upstream of the boundary. The first condition on  $\ell_1^*$  is quasi-isentropic, the second and the third correct the incoming vorticity relating it to the transverse components of the velocity, and the fourth regards the outgoing acoustic pressure wave.

When the inflow is supersonic as on the left-hand side of Figure 2.6 (b), in addition to the conditions expressed by Equations (2.85), the value of  $\ell_5^*$  has to be corrected similarly to  $\ell_4^*$

$$\ell_5^* = K \left[ - \left( \tilde{U} - \tilde{U}_\infty \right) + \frac{P - P_\infty}{\rho c} \right]. \quad (2.86)$$

The solver also includes two generic inflow and outflow boundary conditions that implement either the subsonic or the supersonic versions of the closure by checking node by node the sign of the eigenvalue corresponding to the component of the con-

## 2. NUMERICAL METHOD

---

travariant velocity normal to the boundary. This version is useful when the flow velocities are in the transonic range.

### 2.5.1.2 Inviscid wall

The condition regarding the inviscid wall, described in Kim & Lee [2004], is based on the assumption that the component of the velocity normal to the wall has to be zero. This condition can be enforced starting from Equation (2.80b) that relates the amplitude of the incoming and the outgoing acoustic waves. In the case of an inviscid wall located at  $\xi = 0$  the amplitude of the reflected wave propagating rightwards  $\ell_4^*$  depends on the amplitude of that propagating leftwards  $\ell_5$  as

$$\ell_4^* = \ell_5 + (S_{c4} - S_{c5}) + 2 \left. \frac{\partial \tilde{U}}{\partial t} \right|_{wall} . \quad (2.87)$$

The condition of Equation (2.87) includes the case of a vibrating (or porous) wall and the purely stationary inviscid wall condition is recovered just by setting  $\left. \frac{\partial \tilde{U}}{\partial t} \right|_{wall} = 0$ . Note that if  $S_{c4} = S_{c5} = 0$  and the computation of  $\ell_5$  with Equation (2.81) does not include the contribution of the transverse components, the condition expressed by Equation (2.87) reduces to the wall boundary closure of Poinso & Lele [1992]. As this condition is not linked to any reference value, to avoid the possible drift in the mean value due to the numerical round off error Kim & Lee [2004] suggested, in addition to the previously stated condition, to overwrite the Cartesian components of the velocity for all the nodes on the boundary with the exact values. For a boundary normal to the

## 2. NUMERICAL METHOD

---

curvilinear coordinate  $\xi$ , the correction is

$$\begin{bmatrix} u^* \\ v^* \\ w^* \end{bmatrix} = \begin{bmatrix} \tilde{\xi}_x & \tilde{\xi}_y & \tilde{\xi}_z \\ \tilde{\xi}_y & -\tilde{\xi}_x & 0 \\ -\tilde{\xi}_z & 0 & \tilde{\xi}_x \end{bmatrix}^{-1} \begin{bmatrix} \tilde{U}^* \\ \tilde{V} \\ \tilde{W} \end{bmatrix}, \quad (2.88)$$

where

$$\tilde{\xi}_x = \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}}, \quad \tilde{\xi}_y = \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}}, \quad \tilde{\xi}_z = \frac{\xi_z}{\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}}. \quad (2.89)$$

This additional correction is also implemented in the solver when the Kim and Lee inviscid wall condition is used.

### 2.5.1.3 Inter-block boundary

To complete the set of boundary conditions that can be specified within their framework, Kim & Lee [2003] propose an original specification of the inter-block boundary condition based on the characteristic waves.

Even for the most simple problem geometries, the non-uniform grids very often contain singularities. At these points the slope of a grid line abruptly changes causing a discontinuity in the grid metrics. This condition is often impossible to avoid and, especially for high-order solvers like the one used in this work, it causes spurious oscillations that contaminate the field. Furthermore, it can be the reason for a discontinuity in the solution.

Kim and Lee tackle the problem by using a multi-block computation that relegates the singularities to the borders so to avoid special treatments within the field assigned

## 2. NUMERICAL METHOD

---

to each process. On the borders where the singularities are present, the computation of the grid metrics and that of the flow derivatives is executed using one-sided stencils that do not cross the borders themselves, so that each block is “isolated” from the discontinuity. Once these conditions are satisfied, the actual boundary closure can be enforced in the following way: if a singularity is present between two abutting blocks having a vertical boundary in common, the flow state of a node on the border of the left block (denoted by the letter “L”) has to be equal to that on the corresponding node of the right block (denoted by the letter “R”). To ensure this, their time derivatives at each Runge-Kutta stage have to be the same. This condition can be seen in terms of characteristic waves within the theory given at the beginning of Section 2.5 as

$$\ell^L - \mathbf{S}_c^L = \ell^R - \mathbf{S}_c^R. \quad (2.90)$$

At this point the question is: which is the process that has to pass the values and which the one that has to receive and overwrite them? For each wave, the direction in which the information has to be passed is determined by the sign of the eigenvalues of Equations (2.20) that is related to the direction of propagation.

This technique, potentially useful for the second, third, and fourth test cases illustrated in Chapter 3, is not used for their actual solution as, with the current configuration of the solver, it cannot be employed in association with explicit and implicit filtering methods whose stencils cross the borders. The only filtering technique currently available in the code compatible with this method is the prefactored implicit filtering defined by Hixon [1999], detailed in Section 2.2.3, that currently does not allow the use of variable  $\alpha_f$  (free coefficient) strategy and, therefore, is not used in this work.

## 2. NUMERICAL METHOD

---

### 2.5.2 Giles subsonic non-reflecting boundary conditions

The Giles non-reflecting boundary conditions for the linearised Euler equations described in Giles [1990] are based on the treatment of the wave equation by Engquist & Majda [1977] and on the analysis of well-posedness by Kreiss [1970]. First, the author formulates an ideal non-local boundary condition based on the Fourier analysis of the outgoing waves: this type of boundary requires the knowledge of the flow state on the whole computational field and its complete history. The computation implies a Fourier and a Laplace transform in time and is, in most cases, too computationally expensive to be practically viable. Along with this exact condition Giles proposes three approximate treatments: 1D for unsteady flows, exact 2D for steady-state solution, and approximate 2D for unsteady flows. This last boundary closure is surely the most interesting for the adoption in practical problems and has proven a very popular choice since its proposal. It is second-order accurate and it basically consists in the computation of the incoming wave amplitudes based on the transverse components of the characteristic waves. The advantage of this approach with respect to the treatment given by Thompson [1987, 1990] is that outgoing waves not normal to the boundary are accounted for.

The formulation of the subsonic inflow boundary condition as given by Giles in his article, valid for the application to uniform 2D Cartesian grids only, is, in dimensional form

$$\begin{bmatrix} -c_\infty^2 & 0 & 0 & 1 \\ 0 & 0 & \rho_\infty c_\infty & 0 \\ 0 & \rho_\infty c_\infty & 0 & 1 \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial t} + \begin{bmatrix} -c_\infty^2 v_\infty & 0 & 0 & v_\infty \\ 0 & \rho_\infty c_\infty u_\infty & \rho_\infty c_\infty v_\infty & c_\infty \\ 0 & \rho_\infty c_\infty v_\infty & \rho_\infty c_\infty \frac{c_\infty - u_\infty}{2} & v_\infty \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial y} = 0. \quad (2.91)$$

## 2. NUMERICAL METHOD

---

At the outflow,

$$\begin{bmatrix} 0 & -\rho_\infty c_\infty & 0 & 1 \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial t} + \begin{bmatrix} 0 & -\rho_\infty c_\infty v_\infty & \rho_\infty c_\infty u_\infty & v_\infty \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial y} = 0, \quad (2.92)$$

where, in this 2D case, the vector of the variables  $\mathbf{Q} = [\rho, u, v, p]^T$  does not include the component along the  $z$  axis, and the speed of sound and the flow variables with subscript  $\infty$  are relative to the reference state around which the equations are linearised:

$$\mathbf{Q}_\infty = [\rho_\infty, u_\infty, v_\infty, p_\infty]^T.$$

For their implementation in a program, it is preferable to use the version in characteristic form (also dimensional)

$$\frac{\partial}{\partial t} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_3 \\ \mathbf{R}_4 \end{bmatrix} + \begin{bmatrix} v_\infty & 0 & 0 & 0 \\ 0 & v_\infty & -\frac{(c_\infty + u_\infty)}{2} & -\frac{(c_\infty - u_\infty)}{2} \\ 0 & -\frac{(c_\infty - u_\infty)}{2} & v_\infty & 0 \end{bmatrix} \frac{\partial}{\partial y} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_3 \\ \mathbf{R}_4 \\ \mathbf{R}_5 \end{bmatrix} = 0, \quad (2.93a)$$

$$\frac{\partial \mathbf{R}_5}{\partial t} + \begin{bmatrix} 0 & -u_\infty & 0 & v_\infty \end{bmatrix} \frac{\partial}{\partial y} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_3 \\ \mathbf{R}_4 \\ \mathbf{R}_5 \end{bmatrix} = 0, \quad (2.93b)$$

where the definition of the vector of the characteristic variables  $\mathbf{R}$  is the same used for the Kim and Lee treatment in Equations (2.22).

Note that the time derivatives of the characteristic variables not supplied by the Giles conditions are computed directly from the state of the flow field. The flux derivatives are used to obtain the values of  $\frac{\partial \mathbf{R}}{\partial x}$  that, in turn, through the multiplication by the respective eigenvalues, supply the value of  $\frac{\partial \mathbf{R}}{\partial t}$ .

## 2. NUMERICAL METHOD

---

To generalise the definition of these boundary conditions for the application to non-uniform grids, the approach of Medida [2007] is followed. The generalised equations in primitive dimensional form are

$$\begin{aligned}
 & \begin{bmatrix} -c_\infty^2 \xi_x & 0 & 0 & \xi_x \\ 0 & -\rho_\infty c_\infty \xi_y & \rho_\infty c_\infty \xi_x & 0 \\ 0 & \rho_\infty c_\infty \xi_x & \rho_\infty c_\infty \xi_y & \Psi \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial t} \\
 & + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \rho_\infty c_\infty (-\xi_y v_\infty + \eta_y u_\infty) & \rho_\infty c_\infty (\xi_x v_\infty - \eta_x u_\infty) & c_\infty v \\ 0 & \rho_\infty c_\infty (\xi_x v_\infty + \frac{\eta_x (c_\infty \Psi - u_\infty)}{2}) & \rho_\infty c_\infty (\xi_y v_\infty + \frac{\eta_y (c_\infty \Psi - u_\infty)}{2}) & \Psi v_\infty - c_\infty \mu \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial \eta} = 0,
 \end{aligned} \tag{2.94a}$$

$$\begin{aligned}
 & \begin{bmatrix} 0 & -\rho_\infty c_\infty \xi_x & -\rho_\infty c_\infty \xi_y & \Psi \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial t} \\
 & + \begin{bmatrix} 0 & \rho_\infty c_\infty (-\xi_x v_\infty + \eta_x u_\infty) & \rho_\infty c_\infty (-\xi_y v_\infty + \eta_y u_\infty) & \Psi v_\infty + c_\infty \mu \end{bmatrix} \frac{\partial \mathbf{Q}}{\partial \eta} = 0,
 \end{aligned} \tag{2.94b}$$

where

$$\Psi = \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}, \quad \mu = \frac{\xi_x \eta_x + \xi_y \eta_y}{\Psi \Phi}, \tag{2.95a}$$

$$\Phi = \sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}, \quad v = \frac{\xi_x \eta_y - \xi_y \eta_x}{\Psi \Phi}. \tag{2.95b}$$

## 2. NUMERICAL METHOD

---

The same equations in characteristic form are

$$\frac{\partial}{\partial t} \begin{bmatrix} R_1 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & v_\infty - \frac{\mu u_\infty}{\Psi^2} & -\frac{\nu(c_\infty \Psi + u_\infty)}{2\Psi^2} & -\frac{\nu(c_\infty \Psi - u_\infty)}{2\Psi^2} \\ 0 & -\frac{\nu(c_\infty \Psi - u_\infty)}{2\Psi^2} & v_\infty - \frac{\mu(c_\infty \Psi + u_\infty)}{4\Psi^2} & -\frac{\mu(3c_\infty \Psi - u_\infty)}{4\Psi^2} \end{bmatrix} \frac{\partial}{\partial \eta} \begin{bmatrix} R_1 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} = 0, \quad (2.96a)$$

$$\frac{\partial R_5}{\partial t} + \begin{bmatrix} 0 & -\frac{\nu u_\infty}{\Psi^2} & \frac{\mu(c_\infty \Psi + u_\infty)}{2\Psi^2} & v_\infty + \frac{\mu(c_\infty \Psi - u_\infty)}{2\Psi^2} \end{bmatrix} \frac{\partial}{\partial \eta} \begin{bmatrix} R_1 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} = 0. \quad (2.96b)$$

Note that in the finite-difference solver used for this study the flow derivatives in the transverse direction, from which the derivatives of the characteristic variables are obtained, are always computed using the standard spatial compact scheme. The flow derivatives normal to the boundary are evaluated with the one-sided 7-point stencils defined in Section 2.2.2. Unlike for the Kim and Lee boundary conditions, for which the corrected version of the flux derivatives on the boundary are used to start the sweeps computing the flow on the nodes in the interior of the computational domain, in this case the correction is directly applied to the time derivative and hence it is restricted to the nodes on the boundary.

For the sake of completeness, here the generalisation to non-uniform grids of the extension to 3D Cartesian coordinates of the Giles boundary condition given by Medida [2007] is shown. These conditions, tested in the third test case illustrated in Section 3.3 have shown a long-time instability that makes them not suitable for the

## 2. NUMERICAL METHOD

---

solution of the problem. The equations are shown in characteristic form only

$$\begin{aligned}
 & \frac{\partial}{\partial t} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & v_\infty & 0 & 0 & 0 \\ 0 & 0 & v_\infty - \frac{\mu_{\xi\eta} u_\infty}{\Psi^2} & -\frac{v_{\xi\eta}(c_\infty \Psi + u_\infty)}{2\Psi^2} & -\frac{v_{\xi\eta}(c_\infty \Psi - u_\infty)}{2\Psi^2} \\ 0 & 0 & -\frac{v_{\xi\eta}(c_\infty \Psi - u_\infty)}{2\Psi^2} & v_\infty - \frac{\mu_{\xi\eta}(c_\infty \Psi + u_\infty)}{4\Psi^2} & -\frac{\mu_{\xi\eta}(3c_\infty \Psi - u_\infty)}{4\Psi^2} \end{bmatrix} \frac{\partial}{\partial \eta} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} \\
 & + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & w_\infty - \frac{\mu_{\xi\zeta} u_\infty}{\Psi^2} & 0 & -\frac{v_{\xi\zeta}(c_\infty \Psi + u_\infty)}{2\Psi^2} & -\frac{v_{\xi\zeta}(c_\infty \Psi - u_\infty)}{2\Psi^2} \\ 0 & 0 & w_\infty & 0 & 0 \\ 0 & -\frac{v_{\xi\zeta}(c_\infty \Psi - u_\infty)}{2\Psi^2} & 0 & w_\infty - \frac{\mu_{\xi\zeta}(c_\infty \Psi + u_\infty)}{4\Psi^2} & -\frac{\mu_{\xi\zeta}(3c_\infty \Psi - u_\infty)}{4\Psi^2} \end{bmatrix} \frac{\partial}{\partial \zeta} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} = 0, \quad (2.97a)
 \end{aligned}$$

$$\begin{aligned}
 & \frac{\partial R_5}{\partial t} + \begin{bmatrix} 0 & 0 & -\frac{v_{\xi\eta} u_\infty}{\Psi^2} & \frac{\mu_{\xi\eta}(c_\infty \Psi + u_\infty)}{2\Psi^2} & v_\infty + \frac{\mu_{\xi\eta}(c_\infty \Psi - u_\infty)}{2\Psi^2} \end{bmatrix} \frac{\partial}{\partial \eta} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} \\
 & + \begin{bmatrix} 0 & -\frac{v_{\xi\zeta} u_\infty}{\Psi^2} & 0 & \frac{\mu_{\xi\zeta}(c_\infty \Psi + u_\infty)}{2\Psi^2} & w_\infty + \frac{\mu_{\xi\zeta}(c_\infty \Psi - u_\infty)}{2\Psi^2} \end{bmatrix} \frac{\partial}{\partial \zeta} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} = 0, \quad (2.97b)
 \end{aligned}$$

## 2. NUMERICAL METHOD

---

where

$$\mu_{\xi\eta} = \frac{(\xi_x\eta_x + \xi_y\eta_y)}{\Psi\Phi}, \quad v_{\xi\eta} = \frac{(\xi_x\eta_y - \xi_y\eta_x)}{\Psi\Phi}, \quad (2.98a)$$

$$\mu_{\xi\zeta} = \frac{(\xi_x\eta_x + \xi_z\eta_z)}{\Psi\Phi}, \quad v_{\xi\zeta} = \frac{(\xi_x\eta_z - \xi_z\eta_x)}{\Psi\Phi}. \quad (2.98b)$$

For the application of this type of boundary condition to the test cases of Chapter 3, two modifications are needed. The first regards the possibility to input vortical waves (function of the node coordinates and of time) from the inlet while retaining the non-reflectivity of the boundary closure (as required for test cases two, three and four in Chapter 3). The second modification is relative to the use of the boundary condition for problems where the reference flow field is not known *a priori* and only a minimum set of mean flow reference values is specified on the boundaries (as in the fourth test case in Chapter 3). For these problems, the boundary condition needs to be able to deal with both a steady-state and an unsteady type of solution.

Consider a boundary where the subsonic non-reflecting Giles boundary condition is enforced. In general, the time derivative of the flow field, in primitive form, is composed of the following three parts

$$\frac{\partial \mathbf{Q}}{\partial t} = \left( \frac{\partial \mathbf{Q}}{\partial t} \right)_{\text{perturbation}} + \left( \frac{\partial \bar{\mathbf{Q}}}{\partial t} \right)_{\text{mean}} + \left( \frac{\partial \mathbf{Q}}{\partial t} \right)_{\text{Giles}}, \quad (2.99)$$

where the first component (with subscript “perturbation”) is responsible to feed into the field the desired perturbation (a vortical gust in the case of the problems of Chapter 3), the second component (“mean”) maintains the prescribed mean conditions, and the last (“Giles”) refers to the reaction of the boundary to the outgoing waves, necessary to maintain the non-reflectivity.

## 2. NUMERICAL METHOD

---

The first component of the right-hand side of Equation (2.99) depends on the analytical definition of the incoming disturbance usually known over the whole field or, at least, over the boundary as a function of time and node location. This component can either be initially subtracted directly from the value of the flow field on which the Giles closure is computed, or, alternatively, it can be subtracted, as the analytical derivative is always known on the boundary, from the time and the flux derivatives used to enforce the non-reflecting condition. This part is then re-added to the computed time derivatives after the application of the Giles closure and makes the “perturbation” component “invisible” to the non-reflecting boundary condition. This very effective method was first proposed by Hixon *et al.* [2000] and also adopted by Ragab & Salem-Said [2007].

The “mean” component of Equation (2.99) is not computed from the actual value  $\mathbf{Q}$  of the flow field but from its mean value, denoted by the bar, over a sufficiently long period of time. For the problems solved in Chapter 3, this time period is equal to the period determined by the fundamental frequency of the incoming gust to which the outgoing acoustic waves are also related. This choice allows to minimise data history and, therefore, the quantity of memory required for the computation.

In general, this component is only needed when the reference flow is unknown and it is univocally determined by a minimum set of conditions over the boundaries. In the fourth test case, described in Section 3.4.1, this set of conditions consists of: mean inflow angle  $\bar{\alpha}$ , mean stagnation pressure  $\bar{p}_{tot}$  and temperature  $\bar{T}_{tot}$  at the inflow, and mean static pressure  $\bar{p}$  at the outflow. The following treatment refers to this case in 2D but, with small adaptations, it can be applied to any set of well-posed boundary conditions in 3D. The problem here is to relate the time derivative  $\left(\frac{\partial \bar{\mathbf{Q}}}{\partial t}\right)_{mean}$  to the value of  $\delta\bar{\alpha}$ ,  $\delta\bar{p}_{tot}$ ,  $\delta\bar{T}_{tot}$ , and  $\delta\bar{p}$  that represent the difference between the current value of these variables and their target value specified by the problem.

## 2. NUMERICAL METHOD

---

At the inflow, the stagnation values and the flow angle are related to the primitive variables as

$$p_{tot} = p + \frac{1}{2}\rho(u^2 + v^2), \quad (2.100a)$$

$$T_{tot} = T + \frac{(u^2 + v^2)}{2c_p} = \frac{p}{\rho R} + \frac{(u^2 + v^2)}{2c_p}, \quad (2.100b)$$

$$\alpha = \arctan\left(\frac{v}{u}\right), \quad (2.100c)$$

where the thermodynamic constants  $R$  and  $c_p$  are defined as in Section 2.1.1 and the static temperature  $T$  is

$$T = \frac{p}{\rho R}. \quad (2.101)$$

From Equations (2.100) it is possible to analytically derive the set of relations

$$\delta p_{tot} = \frac{\partial p_{tot}}{\partial \rho} \delta \rho + \frac{\partial p_{tot}}{\partial u} \delta u + \frac{\partial p_{tot}}{\partial v} \delta v + \frac{\partial p_{tot}}{\partial p} \delta p = \frac{(u^2 + v^2)}{2} \delta \rho + \rho u \delta u + \rho v \delta v + \delta p, \quad (2.102a)$$

$$\delta T_{tot} = \frac{\partial T_{tot}}{\partial \rho} \delta \rho + \frac{\partial T_{tot}}{\partial u} \delta u + \frac{\partial T_{tot}}{\partial v} \delta v + \frac{\partial T_{tot}}{\partial p} \delta p = -\frac{p}{\rho^2 R} \delta \rho + \frac{u}{c_p} \delta u + \frac{v}{c_p} \delta v + \frac{\delta p}{\rho R}, \quad (2.102b)$$

$$\delta \alpha = -\frac{v}{u^2 + v^2} \delta u + \frac{u}{u^2 + v^2} \delta v, \quad (2.102c)$$

from which it is evident that, to obtain the variations of the primitive variables from the variations of the available stagnation properties and of the flow angle, an additional relation is needed. Following Ahmadi & Ghaly [1996] and, similarly, Hixon *et al.* [2003, 2004], this relation is supplied by the outgoing characteristic waves. At the inflow, only  $R_5$  defined by Equation (2.22e) is outgoing. When computed from the

## 2. NUMERICAL METHOD

---

averaged values of the flow in a steady-state condition it has to be equal to zero (note that here, unlike in Section 2.1.2 where  $\xi$  is already of unit length, the metrics have to be divided by  $\Psi$  having value defined by Equation (2.95a))

$$\delta R_5 = -\frac{\xi_x}{\Psi}\delta u - \frac{\xi_y}{\Psi}\delta v + \frac{1}{\rho c}\delta p = 0. \quad (2.103)$$

Collecting Equations (2.102) and (2.103) in matrix form and introducing the time averaged values, it follows that

$$\begin{bmatrix} \delta \bar{p}_{tot} \\ \delta \bar{T}_{tot} \\ \delta \bar{\alpha} \\ \delta \bar{R}_5^* \end{bmatrix} = \begin{bmatrix} \frac{(\bar{u}^2 + \bar{v}^2)}{2} & \bar{\rho}\bar{u} & \bar{\rho}\bar{v} & 1 \\ -\frac{\bar{p}}{\bar{\rho}^2 R} & \frac{\bar{u}}{c_p} & \frac{\bar{v}}{c_p} & \frac{1}{\bar{\rho}R} \\ 0 & -\frac{\bar{v}}{\bar{u}^2 + \bar{v}^2} & \frac{\bar{u}}{\bar{u}^2 + \bar{v}^2} & 0 \\ 0 & -\frac{\xi_x}{\Psi} & -\frac{\xi_y}{\Psi} & \frac{1}{\bar{\rho}c} \end{bmatrix} \begin{bmatrix} \delta \bar{p} \\ \delta \bar{u} \\ \delta \bar{v} \\ \delta \bar{p} \end{bmatrix} = \mathbf{H} \begin{bmatrix} \delta \bar{p} \\ \delta \bar{u} \\ \delta \bar{v} \\ \delta \bar{p} \end{bmatrix}, \quad (2.104)$$

where matrix  $\mathbf{H}$  can be inverted, either analytically or numerically (as in the implementation of this work), to give the sought relationship

$$\begin{bmatrix} \delta \bar{p} \\ \delta \bar{u} \\ \delta \bar{v} \\ \delta \bar{p} \end{bmatrix} = \mathbf{H}^{-1} \begin{bmatrix} \delta \bar{p}_{tot} \\ \delta \bar{T}_{tot} \\ \delta \bar{\alpha} \\ \delta \bar{R}_5^* \end{bmatrix}. \quad (2.105)$$

The correction to be applied to the characteristic variables can be directly computed

## 2. NUMERICAL METHOD

---

by pre-multiplying by matrix  $\mathbf{L}^{-1}$  in Equation (2.21b) both sides of Equation (2.105)

$$\begin{bmatrix} \delta\bar{\mathbf{R}}_1 \\ \delta\bar{\mathbf{R}}_3 \\ \delta\bar{\mathbf{R}}_4 \\ \delta\bar{\mathbf{R}}_5 \end{bmatrix} = \mathbf{L}^{-1} \begin{bmatrix} \delta\bar{p} \\ \delta\bar{u} \\ \delta\bar{v} \\ \delta\bar{p} \end{bmatrix} = \mathbf{L}^{-1}\mathbf{H}^{-1} \begin{bmatrix} \delta\bar{p}_{tot} \\ \delta\bar{T}_{tot} \\ \delta\bar{\alpha} \\ \delta\bar{\mathbf{R}}_5^* \end{bmatrix}. \quad (2.106)$$

It has to be noted that, in the practical computation, the value of  $\delta\bar{\mathbf{R}}_5^*$ , as those of the components of matrix  $\mathbf{H}^{-1}$ , is based on the variation of the mean flow variables, while  $\delta\bar{\mathbf{R}}_5$  is the actual correction of the characteristic variables, that, along with the other components, contribute to establish the corrected values of the actual time derivatives on the boundary. When the flow field is in conservative form, the only difference with the above procedure is that matrix  $\mathbf{H}$  and its inverse have to be multiplied respectively by matrix  $\mathbf{M}^{-1}$  and  $\mathbf{M}$  of Equations (2.26) and (2.25). The correction of Equation (2.106) is applied only at the end of each time step and not at each Runge-Kutta stage as the other terms in Equation (2.99).

At the outflow, the treatment is similar with the only difference that, in this case, one condition is provided by the problem (the value of the static pressure  $p$ ) and the remaining three conditions are given by imposing the amplitude of the outgoing char-

## 2. NUMERICAL METHOD

---

acteristic waves equal to zero. The condition can be summarised as

$$\begin{bmatrix} \delta\bar{\mathbf{R}}_1^* \\ \delta\bar{\mathbf{R}}_3^* \\ \delta\bar{\mathbf{R}}_4^* \\ \delta\bar{p} \end{bmatrix} = \begin{bmatrix} \frac{\xi_x}{\Psi} & 0 & 0 & \frac{-\xi_x}{\bar{c}^2\Psi} \\ 0 & \frac{\xi_y}{\Psi} & -\frac{\xi_x}{\Psi} & 0 \\ 0 & \frac{\xi_x}{\Psi} & \frac{\xi_y}{\Psi} & \frac{1}{\bar{\rho}\bar{c}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta\bar{\rho} \\ \delta\bar{u} \\ \delta\bar{v} \\ \delta\bar{p} \end{bmatrix} = \mathbf{H} \begin{bmatrix} \delta\bar{\rho} \\ \delta\bar{u} \\ \delta\bar{v} \\ \delta\bar{p} \end{bmatrix}, \quad (2.107a)$$

$$\begin{bmatrix} \delta\bar{\mathbf{R}}_1 \\ \delta\bar{\mathbf{R}}_3 \\ \delta\bar{\mathbf{R}}_4 \\ \delta\bar{\mathbf{R}}_5 \end{bmatrix} = \mathbf{L}^{-1} \begin{bmatrix} \delta\bar{\rho} \\ \delta\bar{u} \\ \delta\bar{v} \\ \delta\bar{p} \end{bmatrix} = \mathbf{L}^{-1} \mathbf{H}^{-1} \begin{bmatrix} \delta\bar{\mathbf{R}}_1^* \\ \delta\bar{\mathbf{R}}_3^* \\ \delta\bar{\mathbf{R}}_4^* \\ \delta\bar{p} \end{bmatrix}. \quad (2.107b)$$

### 2.5.3 Symmetry

As seen in Section 2.2.2, the treatment of a symmetry boundary condition can be done in a general standard way by employing the ninth-order centred 11-point stencils defined by Equations (2.47) with the coefficients shown in Equation (2.48). The flow values of the first five rows of nodes near the boundary have to be mirrored on the other side of the boundary and stored in five rinds of additional *ghost* nodes. Then the centred stencil of Equation (2.47) can be applied to compute the forward and backward half-derivatives necessary to start the sweep computing the values of the derivative within the field. The only disadvantage of this technique is related to the amount of error introduced by the centred explicit stencil that, although very similar for its spectral characteristics to the compact scheme, can not ensure the same level of accuracy for every wavenumber.

When the axis is parallel to one of the Cartesian axes, a simple alternative to the use of the explicit stencil is available. It is based on the characteristics of the compact

## 2. NUMERICAL METHOD

---

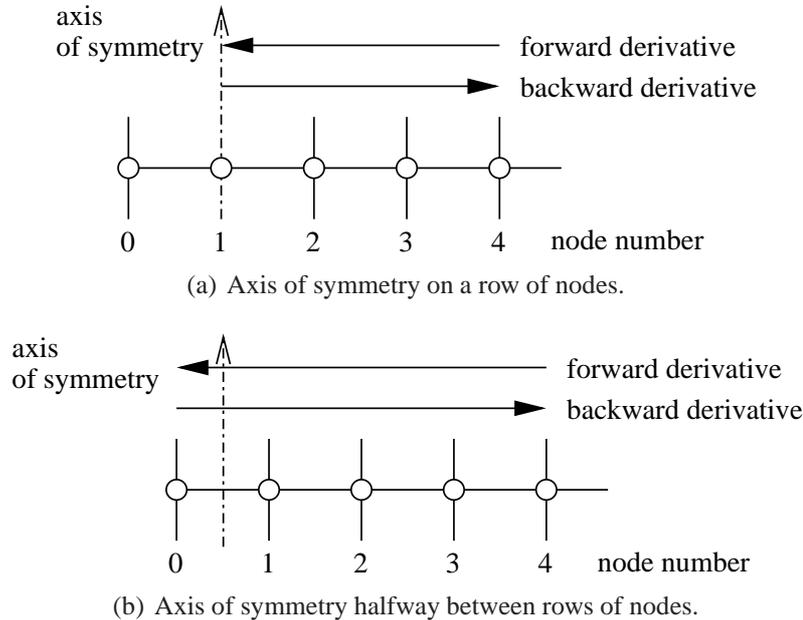


Figure 2.7: Two schematics illustrating the alternative types of treatment of the symmetry boundary condition.

scheme and does not introduce any additional error with respect to it as the boundary nodes are treated exactly as nodes belonging to the interior of the domain. The following description regards a generic 2D problem with the Euler equations in primitive form.

In the standard treatment, the calculation of the forward and backward spatial derivatives for a row of nodes is executed with two sweeps starting from the opposite sides. They are independent and their order of execution bears no importance. With the alternative technique the computation starts from the edge opposite to that where the symmetry is enforced. There, the value of the half-derivatives is known and the sweep evaluating, for example, the forward derivative can be executed. The alternative symmetry condition is applicable to the two following cases: (i) the axis of symmetry is coincident with a row of nodes as shown in Figure 2.7 (a), or (ii) the axis of symmetry lies exactly halfway between two rows of nodes as shown in Figure 2.7

## 2. NUMERICAL METHOD

---

(b). In the former case, when the first node from the axis of symmetry of the physical domain is reached (node 2 in detail (a)), the spatial derivative, being based on the value of the flow at nodes 1, 2 and 3 and on the half-derivative at node 3, can still be calculated by the three-point stencil of the prefactored compact scheme described in Section 2.2. For the calculation at node 1 an exception is introduced in the code: due to the symmetry of the flow variables, it is possible to imagine a further point (number 0 in detail (a) of Figure 2.7) having the same flow characteristics of node 2 with the exception of the velocity component perpendicular to the axis, that has opposite sign. So, the forward derivative at node 1 can be computed as if it was part of the interior domain. Its backward derivative can be evaluated by exploiting again the symmetry of the problem. In fact, for the pressure, the density and the velocity component parallel to the axis of symmetry, it is always equal to the forward derivative with opposite sign. This is obvious from the fact that the sum of the two half-derivatives, representing the derivative at the node, on the boundary has to be zero. The only exception is the velocity component normal to the axis of symmetry, for which the two spatial derivatives are equal and with the same sign. Once the value of the backward derivative at node 1 is known, it is possible to execute the second sweep for the evaluation of the backward half-derivative in the domain interior.

For the second case of Figure 2.7, in which the axis does not lie on a row of nodes but is located halfway between them as in detail (b), the computation is even simpler as the backward derivative at node 0 is equal and opposite to the forward derivative at 1 with the usual exception of the velocity component normal to the axis of symmetry, for which the two contributions have the same sign. In this case, at the end of each Runge-Kutta stage, the values of row 1 need to be copied to row 0 with the exception of the normal velocity component for which a slightly more complex treatment is needed.

## 2. NUMERICAL METHOD

---

When both the opposite boundaries of a block are of symmetric type, the illustrated technique can be used anyway by starting the sweeps with values of the half-derivatives computed by an explicit stencil and executing them twice. This represents an additional computational cost that has to be considered.

### 2.5.4 Buffer zone

The *buffer* (or *sponge*) zone method, originally proposed by Wasistho *et al.* [1997] but here used in the formulation given by Chen *et al.* [2004], reduces the reflections from a boundary by numerically damping the outgoing waves. The numerical damping is applied within a zone abutting the boundary by modifying the solution vector  $\mathbf{Q}$  at every node via a coefficient  $\sigma$  that depends on the node position. In a 1D case, with the left boundary at  $x = 0$  the correction takes the form

$$\mathbf{Q} = \mathbf{Q} - \sigma(\mathbf{Q} - \mathbf{Q}_{\text{target}}), \quad (2.108a)$$

$$\sigma(x) = \alpha \left( \frac{L_b - x}{L_b} \right)^\beta, \quad (2.108b)$$

where  $L_b$  is the buffer layer thickness equal to the distance between the external boundary of the computational domain and the coordinate of the first node not corrected, and  $\alpha$  and  $\beta$  are two coefficients that in this work are set respectively equal to 1 and 3.5.  $\alpha$  controls the value of  $\sigma$  at the boundary and  $\beta$  controls the blending of the interior flow solution with the  $\mathbf{Q}_{\text{target}}$ .  $\mathbf{Q}_{\text{target}}$  is a set of reference values, usually constant. The treatment in case of conservative variables is identical with  $\mathbf{U}$  and  $\mathbf{U}_{\text{target}}$  that take the place of  $\mathbf{Q}$  and  $\mathbf{Q}_{\text{target}}$ . This boundary condition is particularly effective when the buffer thickness  $L_b$  is large and, therefore, the discontinuity at the interface between

## 2. NUMERICAL METHOD

---

the treated zone and the rest of the field is minimised. The main shortcoming of this technique is related to its computational cost due to the additional nodes in the buffer layer.

A modified version of the buffer zone is newly devised in this work in order to associate the non-reflectivity of its standard treatment with the necessity to introduce a vortical gust from the boundary inflow as required by most of the test cases described in Chapter 3. Numerically, the treatment follows Equations (2.108) with the only difference that the target value  $Q_{\text{target}}$  is not constant but equal to the sum of the flow reference value plus the vortical gust superimposed on it. This buffer layer treatment with a time-dependent  $Q_{\text{target}}$  can be implemented provided the gust is known over the whole thickness of the buffer as it is in test case number two and three, and can also be applied to the outflow when the field near the outlet is sufficiently similar to the introduced gust.

### 2.6 Parallelisation

In order to solve problems on models having a number of nodes exceeding the computational power of a single processor in terms of addressable memory, it is necessary to distribute the computational effort executing the code in parallel on multiple processors. The available strategies to realize this are various and essentially depend on the nature of the solver that needs to be “parallelised” and on the maximum number of processors that have to be employed in parallel for the solution of the problem. In order to ensure the maximum flexibility and portability, and to allow the solution of problems on an indefinite number of processors, the message passing interface (MPI) standard system is chosen. In particular, the model adopted for this work is MPI-1

## 2. NUMERICAL METHOD

---

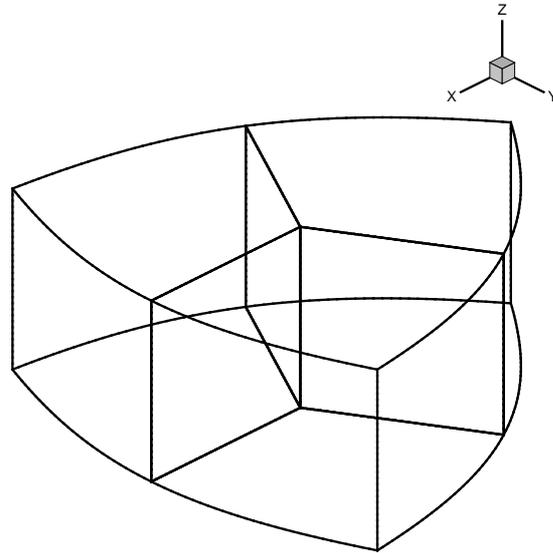
that does not allow shared memory. Details regarding this technique can be found, for example, in Pacheco [1997].

MPI can be used to modify the algorithm of the original “scalar” code in different ways. For instance, it is possible to assign different independent tasks to a number of processes that, at the end of the computation, exchange their partial results to obtain the general solution of the computation. Given the nature of the finite-difference solver used in this work, the strategy called *domain decomposition* or, more accurately, *data decomposition* is used [Smith *et al.*, 1996]. With this strategy the various processes, as the independent entities connected by the message passing interface are called, all execute the same algorithm on sets of data corresponding to different parts of the flow field. Hence, the computational steps executed by the various processes can only differ due to the conditions enforced at the boundaries. The data decomposition type used for this work is called *non-overlapping* as the portions of the physical field assigned to each process have in common only the nodes on the boundary. Note that this definition does not regard the overlap due to the presence of the *ghost* nodes necessary to the data exchange and described further on in this chapter. More about the parallelisation and data decomposition strategies can be found in Smith *et al.* [1996] and Quarteroni & Valli [1999].

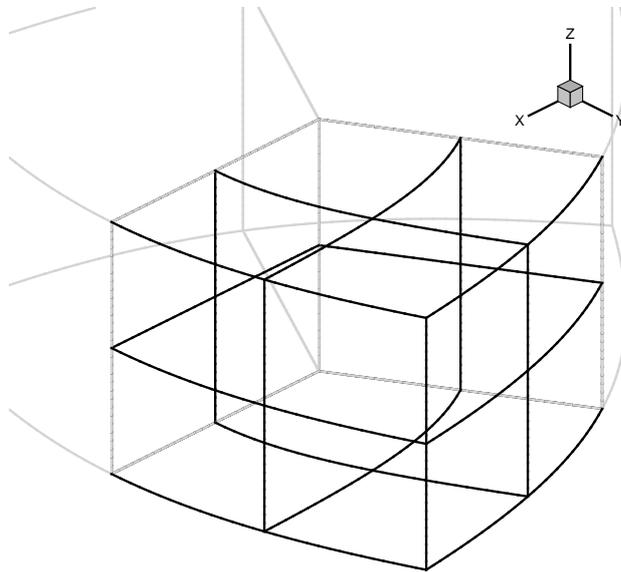
Figure 2.8 shows an example of the two levels of data decomposition related to a simple 3D multi-zone model. The initial model, shown in detail (a), is composed of a number of structured blocks connected through their faces. This is the first level of data decomposition that depends on the structure of the original model. The solver, as shown in detail (b), allows to decompose each of the initial blocks in the desired number of parts along every curvilinear direction. The resulting sub-zones are still structured grids and can be assigned to different processes.

## 2. NUMERICAL METHOD

---



(a) First level of data decomposition related to the structure of the original model.



(b) Second level of data decomposition.

Figure 2.8: General decomposition of a multi-zone model.

## 2. NUMERICAL METHOD

---

The two levels of data decomposition seen in Figure 2.8 imply two levels of data communication. Consider the simple multi-block 2D model of Figure 2.5: it is composed of two structured blocks. Suppose to divide zone 1 in 4 parts and zone 2 in 12 parts as shown on the left-hand side of Figure 2.9. The communication of the processes within each of the initial zones in MPI is called *intra-communication* and is represented in the figure by black solid arrows. The set up of these communications largely relies on functions, provided by MPI, that allow to automatically subdivide a structured mesh in smaller subsets of data assigned to different processors, creating what is called a Cartesian communicator. While the domain is decomposed, a set of information is also provided to each process (essentially, the position of the process within the structured block and its local rank) to allow the exchange of messages between neighbouring processes. The right-hand side of Figure 2.9 illustrates the topology of the model when decomposed by MPI in two separate Cartesian communicators: the local rank of each process and its Cartesian coordinates within the communicator are shown.

The second level of data communication aims to allow the exchange of data between processes that are part of two separate abutting zones. It is usually called *inter-communication* and it is represented by dashed arrows in the schematic on the left-hand side of Figure 2.9. In this case two connected zones can have different orientations of the curvilinear coordinates. For example, in Figure 2.5 the connection along the upper edge of zone 1 has coordinates in agreement between the two blocks as  $\xi_1$  coincides with  $\xi_2$ , and  $\eta_1$  and  $\eta_2$  have the same orientation. In contrast, the connection along the right edge of zone 1 has curvilinear coordinates that are not compatible between the two zones and the message passing requires a manipulation of the data. For this reason, the second level of communication is much more problematic than the previous and

## 2. NUMERICAL METHOD

---

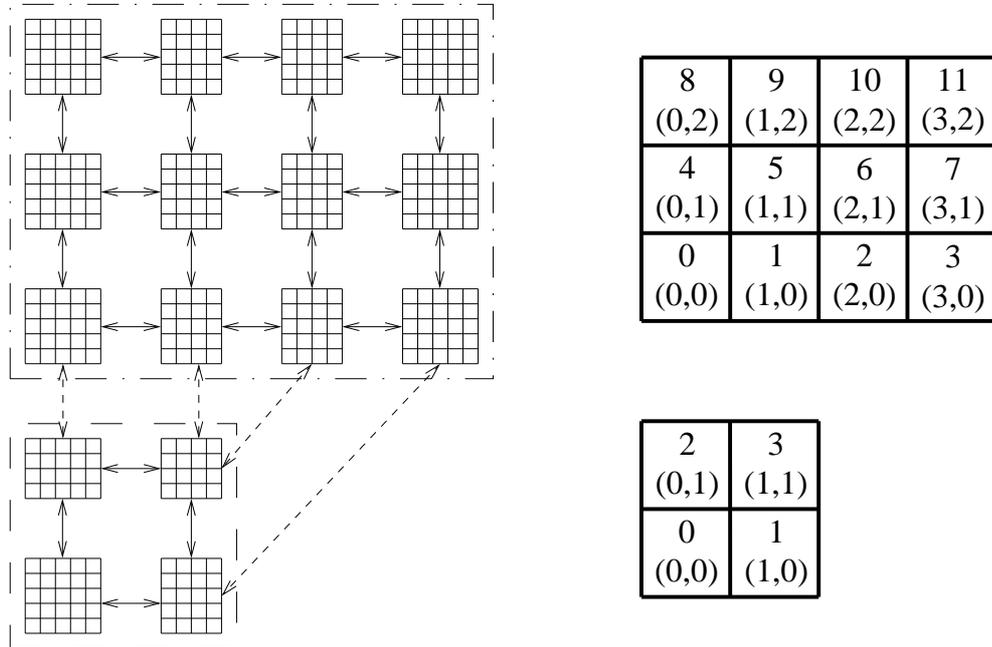


Figure 2.9: An example of the decomposition of a two-zone model via MPI Cartesian communicators.

its implementation can be made in a number of ways following different approaches. That adopted in this work avoids the use of the *inter-communicators* as specified by MPI because they would affect the efficiency of the code as they represent an exception in the code algorithm. More simply, the communication is based on the use of the general ranks. These are values that unambiguously identify each process within the most general communicator (usually called `MPI_COMM_WORLD`, including all the processes). For this purpose the code goes through the following steps:

1. each process, that already has the possibility to communicate with its neighbours belonging to the same block (since it knows their local rank within the Cartesian communicator), exchanges its general rank with the surrounding processes. This allows to pass messages using the general communicator within each block;
2. the *master* process (that, as explained in Section 2.7, manages the general input-

## 2. NUMERICAL METHOD

---

output of data) executes a part of the code that, starting from the geometrical characteristics of each connection between the different zones of the initial model, computes the general rank of each couple of abutting processes and sends a message with this information directly to every process involved;

3. at this point, each process knows the general rank of all its neighbours, including those that are part of a different block, and can use it to send and receive data.

Once the communication between each process and all its neighbours is established, and the initial condition of the flow field is distributed along with the boundary conditions, the proper computation can be started.

Consider a simple 2D mesh with 17 nodes per edge like the one shown in detail (a) of Figure 2.10. The boundary conditions enforced on each side are periodic and use the same algorithm of the inter-block as detailed in Section 2.2.2. The mesh can be divided, for instance, in 4 squared parts, outlined in the figure by thick black lines, having the same extension (9 nodes per edge) and assigned to different processes. Detail (b) of Figure 2.10 shows the grid that each of the four processes has to allocate in its memory. It is not limited to the field portion assigned (greyed out in the figure) but has to include additional rinds of nodes. In fact, as seen in Section 2.2.2, for the sweeps computing the derivatives to start, the values of the half-derivatives on the boundaries are needed. These values are provided by one-sided stencils for a wall or an outer boundary, or by centred stencils in case of inter-block boundary. This implies that along the inter-block boundaries each process also needs to know a part of the field owned by its neighbours. Its size depends on that of the employed stencil. This requires a data exchange between neighbouring processes at the end of each Runge-Kutta stage. The data received from the neighbours is stored in an additional part of the computational grid usually called *ghost* nodes. The adoption of the explicit 11-

## 2. NUMERICAL METHOD

---

point stencil of Equation (2.47) for the computation of the half-derivative (black and white dots in detail (b) in Figure 2.10) implies 5 rinds of ghost nodes (black dots in the figure) along each inter-block boundary. It has to be noted that in the code object of this work the approach is completely parametric and it leaves the freedom to specify any number of rinds and so, if needed, it is possible to use stencils of any length and order of accuracy.

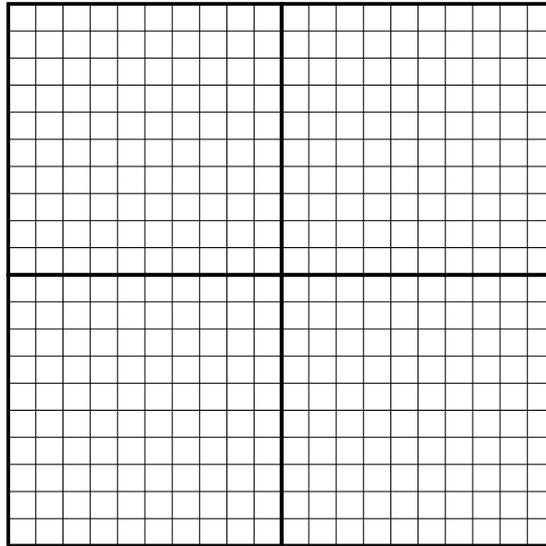
### 2.6.1 Communication in reduced form

The main drawback of the type of computation shown above is that when high-order schemes are used, the number of rind nodes necessary to store the data received from the neighbours can be relevant, especially when 3D grids are concerned. In terms of memory usage, this aspect could represent a strong limitation. To overcome this problem, the computation can be approached in a different way: instead of computing the whole backward and forward derivatives in one go, it is possible to use a two-step process.

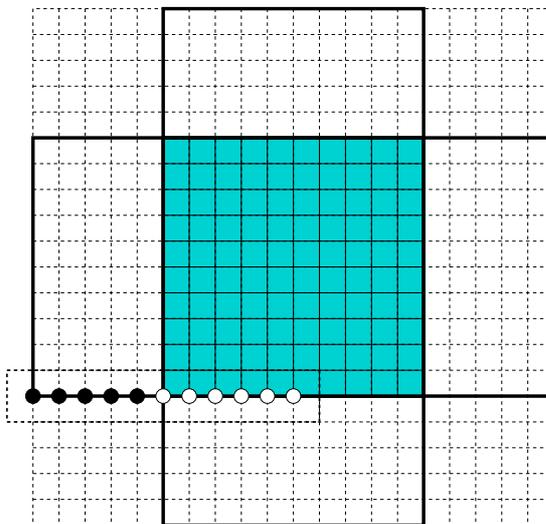
First, instead of using the whole 11-point stencil shown in Equation (2.47), every process computes just one half of the spatial derivatives across the boundary, based on the nodes that belong to its own share of the flow field (e.g. the nodes defined by the white dots in detail (b) of Figure 2.10). This is done with the use of stencils that cover 6 nodes. For the boundary on the left-hand side of the grid in detail (b), denoted by

## 2. NUMERICAL METHOD

---



(a) Two dimensional model with periodic boundaries divided into four parts.



(b) The actual local grid of one of the processes shown in detail (a). In addition to the physical data proper to that process (the area greyed out), five rings of nodes are included to store part of the flow field of the neighbouring processes.

Figure 2.10: An example of the decomposition in four parts of a 2D block with periodic boundaries.

## 2. NUMERICAL METHOD

---

node index 0, the stencils are defined as

$$\left(\frac{\partial f_0^F}{\partial x}\right)_1 = \frac{1}{2}b_0f_0 + \sum_{i=1}^5 b_i f_i, \quad (2.109a)$$

$$\left(\frac{\partial f_0^B}{\partial x}\right)_1 = -\frac{1}{2}b_0f_0 - \sum_{i=1}^5 b_{-i} f_i, \quad (2.109b)$$

where the coefficients  $b_i$  are still those shown in Equation (2.48).

Similarly, for the boundary on the right-hand side of the grid in detail (b) of Figure 2.10, denoted by index  $imax$

$$\left(\frac{\partial f_{imax}^F}{\partial x}\right)_2 = \frac{1}{2}b_0f_{imax} + \sum_{i=-5}^{-1} b_i f_{imax+i}, \quad (2.110a)$$

$$\left(\frac{\partial f_{imax}^B}{\partial x}\right)_2 = -\frac{1}{2}b_0f_{imax} - \sum_{i=-5}^{-1} b_{-i} f_{imax+i}. \quad (2.110b)$$

Then, the two halves of the derivatives (just two values for each boundary node) are exchanged with the neighbouring process and the whole value can be reconstructed. This method reduces the number of required grid nodes to two, and, most importantly, keeps them constant when the order of accuracy is increased. So, even with stencils having more than eleven points, the quantity of additional memory and the bandwidth required by the message passing does not change. Notice that, apart from a different round off, the results are numerically identical to the ones obtained by the standard method.

Unfortunately, this technique cannot be used when the problem to be solved requires the application of high-order filtering methods like the explicit and the standard implicit described in Section 2.2.3 that employ stencils crossing the borders. In these cases the message passing of the whole set of flow data is necessary.

## 2. NUMERICAL METHOD

---

### 2.7 Input/output

In the current version of the solver, the initial data of the problem (essentially: flow field and node coordinates) can either be introduced by modifying a specific module of the code or by using a standard CFD General Notation System (CGNS) external file. The latter method neatly separates the phase of model definition with the actual computation and enables the user to examine and modify the contained data and, if required, to exchange them with other researchers. The CGNS files can be managed by codes written in C or Fortran via an open source mid-level library of functions called *libcgns* available on different platforms. In addition, recently, a similar library called *mexCGNS* has been developed to be used in association with high-level programming languages like MATLAB and GNU Octave.

Independently of the way in which the initial data are introduced, they need to be distributed to all the processes in order to provide them with the minimum set of information required for starting the computation. This distribution can be executed essentially in two ways:

1. One of the processors (called *master*) is in charge of the reading and writing from and to the external files, so that all the stream of data passes through it. It distributes and collects data with one to one communications while the computation is initialised and when the results are output. A variant of this method can be implemented using intermediate passages, with the master process sending blocks of data to *local* masters that then distribute the data to the single processes.
2. Each process, having been provided by the master of enough information, reads the coordinates and the initial flow field directly from the external file. For the output, similarly, each process directly accesses the file and writes its part of the

## 2. NUMERICAL METHOD

---

results.

The first solution is the only one available on those high-performance computing (HPC) clusters in which just one process is allowed to access the disk for reading and writing purposes. For this reason it is the most reliable and it is portable on every system. Unfortunately, it also brings some disadvantages. The main ones regard the superior complexity of the code and the fact that the process of input/output is much slower than that resulting from the second solution. It is also important to notice that the second option can actually be realized in different ways depending on the characteristics of the host system (type of connection between the nodes and file system used) and on the type of output (CGNS or other type of data file). The second option also requires the adoption of MPI-2, a level of the MPI standard that contains an extended set of functions with respect to MPI-1, including shared memory operations and parallel input/output.

The first solution was chosen for the program object of this work because of its higher portability, but the code was written in a way that allows to easily implement the second solution in those cases (e.g. models particularly large that have to be distributed on hundreds of processes) where the requirements regarding the reading/writing speed should become of the highest importance.

All the additional parameters necessary to execute the simulations are read from an external text file. This file is used independently of the type of input of the initial conditions and of the output of the results. It consists of three sections: *parameters always read*, *parameters read if the input is from CGNS*, and *special sections*.

## 2. NUMERICAL METHOD

---

### 2.8 Error norms

Let  $\mathbf{U}_n$  be the numerical solution of the flow state and  $\mathbf{U}_a$  the analytical or the reference solution defined on the same  $i^{th}$  point, the multidimensional error vector  $\mathbf{e}$  can be defined as the difference between the two values:  $\mathbf{e}_i = |\mathbf{U}_n - \mathbf{U}_a|$ . To evaluate the accuracy of a set of results and its order of convergence towards the reference solution when the mesh is refined, the following two error norms are used throughout Chapter 3:

1. the *normalised  $l_2$ -norm* (also referred to in the literature as “ $l^2$ -norm”)

$$l_2 - norm = \sqrt{\frac{\sum_{i=1}^N e_i^2}{N}}, \quad (2.111)$$

2. the  *$l_\infty$ -norm* (also referred to in the literature as  *$l_{max}$ -norm*)

$$l_\infty - norm = \max(e_i), \quad i = 1, \dots, N. \quad (2.112)$$

# Chapter 3

## Test case applications

This chapter contains the results from four test cases that have been solved to demonstrate the ability of the high-order finite-difference solver to simulate the conversion of kinetic energy into noise by the interaction of vorticity waves with rigid surfaces. This goal is achieved by solving four test cases of increasing complexity.

The *first test case* models the sound field generated by an oscillating field around a still cylinder. This test case is used to test the effectiveness of the Kim and Lee generalised boundary conditions detailed in Section 2.5.1. In addition, this case allows to compare the results from the linearised and the non-linear formulation of the Euler equations and study the influence of non-linear effects on the results.

The *second test case* aims to simulate the two-dimensional interaction of an incident gust with a cascade of flat plates. It is a simplified version of the rotor-stator interaction problem for which an analytical solution is available.

The *third test case* is the extension into three dimensions of the second test case and can assess the ability of the solver to address large problems on distorted grids in

### 3. TEST CASE APPLICATIONS

---

three dimensions.

In the *fourth test case*, the rotor-stator interaction in two dimensions is simulated introducing further details including a realistic compressor blade profile and an incoming multi-frequency spectrum vortical gust. This tests the ability of the code to tackle a problem of practical engineering interest using a stretched multi-block computational mesh.

All but the first test case are taken from the NASA workshops on computational aeroacoustics that are periodically organised to verify the technological advancement in the numerical computation of the generation and of the propagation of sound.

In all four cases, the working fluid is air, modelled using perfect gas assumptions, with specific gas constant  $R = 287.06$  J/(kg K) and heat capacity ratio  $\gamma = 1.4$ .

For each problem, the metrics for the transformation from the physical to the computational space and their inverse are evaluated numerically using the same spatial scheme as used for the flow derivatives, even when, as for the first three test cases, the exact analytical solution is available. According to Visbal & Gaitonde [1998] this technique ensures a higher level of overall accuracy.

All the simulations have been run on ALICE, the high performance computing cluster of the University of Leicester. This cluster is composed of 256 nodes, and each node includes two quad-core 2.67GHz Intel Xeon X5550 CPUs with 12 GB of RAM. The operating system is the 64-bit version of Scientific Linux 5.4.

Unless otherwise stated, the solver has been compiled with version 12.1.3 of the 64-bit Intel compiler. The compilation flags included several options to maximise the speed of execution. Other relevant libraries used in the code for specific tasks are: Intel

### 3. TEST CASE APPLICATIONS

---

MPI version 4.0.3, FFTW version 3.2.2, Intel Math Kernel Library (MKL) version 10.3.9, and CGNS library version 2.54.

## 3.1 First test case: dipole sound generated by an oscillating flow field around a still cylinder

The first test case is a modified version of the problem presented by Kim & Lee [2004], where a cylinder oscillates vertically in a steady inviscid flow generating the acoustic field typical of a dipole (see Figure 3.1). Viscosity is neglected.

### 3.1.1 Problem definition

The position of the centre of the cylinder of diameter  $D$  depends on time  $t$  and is defined by the coordinates

$$x(t) = 0, \tag{3.1a}$$

$$y(t) = -\epsilon \frac{c_\infty}{\omega} \cos(\omega t), \tag{3.1b}$$

where  $\epsilon$  is a non-dimensional value equal to  $10^{-4}$ ,  $c_\infty$  is the ambient speed of sound, and  $\omega = 2\pi c_\infty/D$  is the angular frequency of the oscillation. Any point  $P$  in the 2D field is defined by the distance  $r$  from the origin of the axes and by the angle  $\phi$  measured from the axis of oscillation  $y$ . In order to correctly define the boundary condition on the wall surface it is important to notice that the velocity normal to the wall  $\tilde{U}$  and its

### 3. TEST CASE APPLICATIONS

---

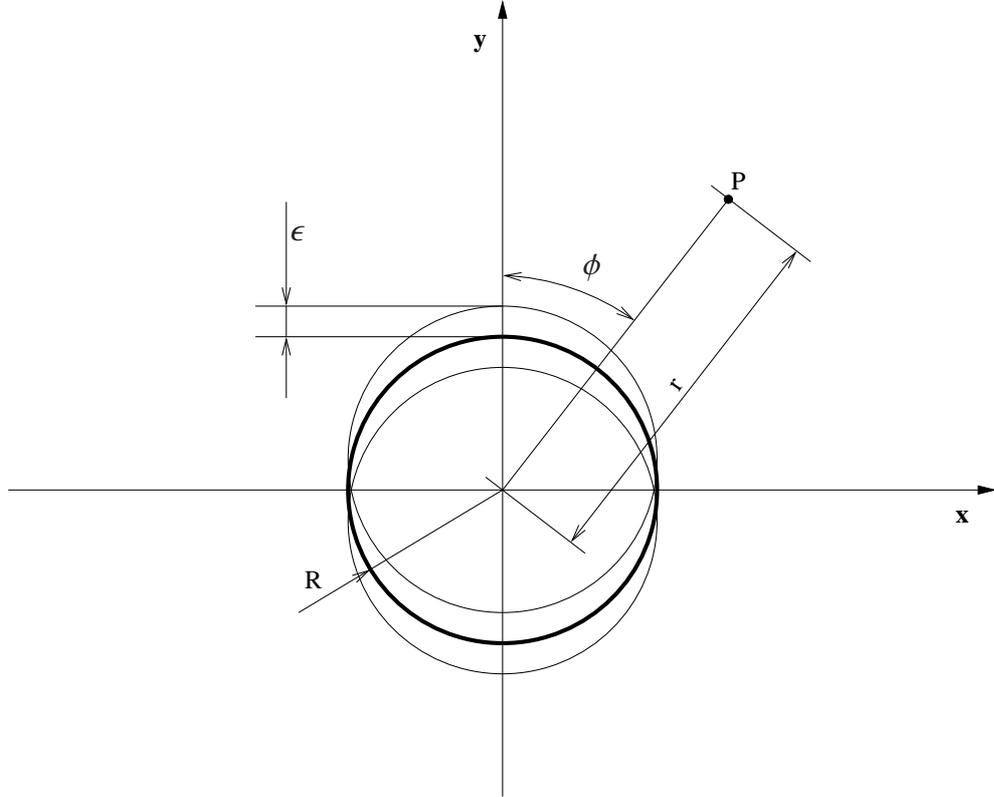


Figure 3.1: Schematic of the problem as in Kim & Lee [2004]. Oscillating cylinder in a still air.

time derivative are equal to

$$\begin{aligned}\tilde{U}_{wall} &= \epsilon c_{\infty} \cos \phi \sin(\omega t), \\ \left. \frac{d\tilde{U}}{dt} \right|_{wall} &= \epsilon \omega c_{\infty} \cos \phi \cos(\omega t).\end{aligned}\tag{3.2}$$

Kim & Lee [2004] run four simulations with different values of the mean flow velocity ( $M_{\infty} = 0, 0.1, 0.2,$  and  $0.4$ ). An analytic solution is known only for the case with no mean flow. The perturbation in the pressure field is given by Dowling &

### 3. TEST CASE APPLICATIONS

---

Ffowcs Williams [1983]

$$p'(r, \phi, t) = -\rho_\infty \epsilon c_\infty^2 \cos \phi e^{i\omega t} \frac{H_0^{(2)'}(\omega r/c_\infty)}{H_0^{(2)'}(\omega R/c_\infty)}, \quad (3.3)$$

where  $\rho_\infty$  is the reference ambient density,  $R = D/2$  is the radius of the cylinder, and  $H_0^{(2)'}$  and  $H_0^{(2)''}$  are respectively the first ( $H_0'(z) = \frac{dH_0}{dz}$ ) and the second ( $H_0''(z) = \frac{d^2H_0}{dz^2}$ ) derivatives of the zero-th order Hankel function of the second kind. These functions can be easily computed exploiting one of their properties that relates the derivative of the Hankel function to the function itself and to the function of order immediately higher

$$\frac{d}{dz} H_p(\alpha z) = -\alpha H_{p+1}(\alpha z) + \frac{p}{z} H_p(\alpha z). \quad (3.4)$$

By applying Equation (3.4), Equation (3.3) becomes

$$p'(r, \phi, t) = -\rho_\infty \epsilon c_\infty^2 \cos \phi e^{i\omega t} \frac{H_1^{(2)}(\omega r/c_\infty)}{\frac{c_\infty}{\omega R} H_1^{(2)}(\omega R/c_\infty) - H_2^{(2)}(\omega R/c_\infty)}. \quad (3.5)$$

This work considers the reciprocal problem of a still cylinder immersed in an oscillating flow field as sketched in Figure 3.2. This can be modelled by setting the body forces vector  $\mathbf{f} = [f_x, f_y, f_z]^T$  in Equations (2.1) or Equations (2.2) equal to

$$\mathbf{f} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} 0 \\ \epsilon c_\infty \omega \cos(\omega t) \\ 0 \end{bmatrix}. \quad (3.6)$$

This changes the value of the  $\mathbf{S}$  vector in the set of Euler equations either in the conser-

### 3. TEST CASE APPLICATIONS

---

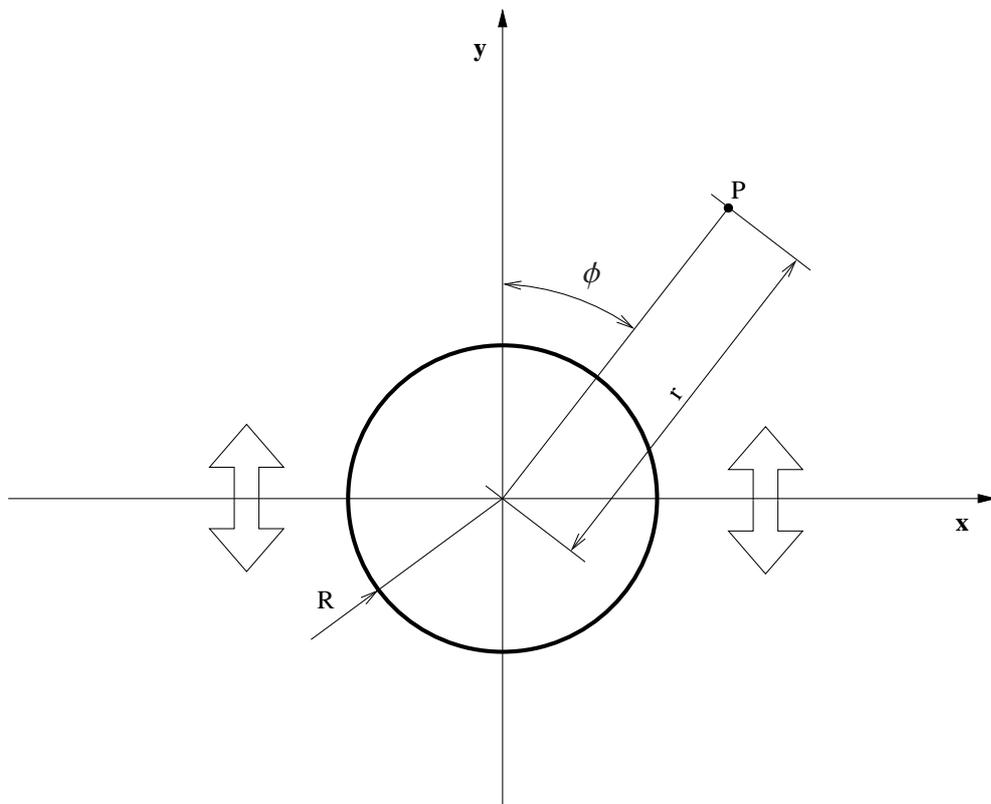


Figure 3.2: Schematic of the modified problem. Oscillating field around a still cylinder.

### 3. TEST CASE APPLICATIONS

---

vative form of Equation (2.5) or in the primitive form of Equation (2.9) and the value of vectors  $\mathbf{S}_v$  and  $\mathbf{S}_v^*$  in the Kim and Lee boundary treatment of Section 2.5.1.

Let subscript ‘‘of’’ define the oscillating flow field. Then

$$\mathbf{Q}_{\text{of}} = \begin{bmatrix} \rho \\ u \\ v \\ w \\ p \end{bmatrix} = \begin{bmatrix} \rho_0 \\ 0 \\ \epsilon c_\infty \sin(\omega t) \\ 0 \\ p_0 \end{bmatrix}, \quad (3.7)$$

where  $\rho_0$  and  $p_0$  are the constant reference values for density and pressure.  $\mathbf{Q}_{\text{of}}$  is a valid solution of the Euler set of equations that includes the body forces  $\mathbf{f}$ . This can be demonstrated by substituting Equation (3.7) into Equation (2.8b) with  $\mathbf{f}$  from Equation (3.6).

The analytical solution of Equation (3.3) to the original problem presented by Kim and Lee is also the analytical solution to the modified problem. The former can be transposed into the latter by the means of a change in the system of reference. The amplitude of the oscillation is taken to be vanishingly small as this is an inherent assumption in the analytical solution of Equation (3.5) which is only valid for linear oscillations.

### 3. TEST CASE APPLICATIONS

---

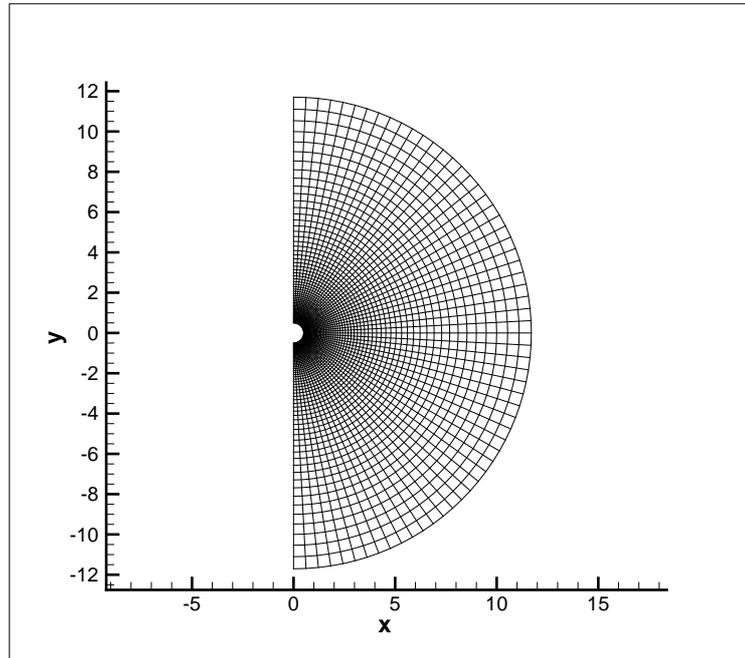
#### 3.1.2 Numerical methods

A set of increasingly refined meshes was created to model the test case of Section 3.1.1. The coarsest grid, which is referred to as grid number 1 further in the text, is shown in Figure 3.3. It has an O-grid topology and, to reduce the computational cost of the simulation, the symmetry of the problem is exploited by modelling only half of the field, so that the part located in the negative half-plane along the  $x$  axis is neglected. All the dimensions are normalised with respect to the cylinder diameter  $D$ . The characteristics of the five grids employed are summarised in Table 3.1. Grid number 1 was designed to have approximately squared cells near the central cylinder. This allows to minimise the error due to the grid distortion. In the circumferential direction (curvilinear coordinate  $i$  in detail (b) of Figure 3.3) the nodes are equally spaced. Once the first two rows of nodes around the cylinder are determined, a constant stretching factor (the ratio between the distances of two successive pair of nodes) in the radial direction is applied to compute the position of all the nodes. The external radius ( $R_e = 11.7049D$ ) is kept constant for all the five grids, making the results comparable in every area of the field. Every grid is obtained from the previous one and includes all its nodes. New nodes are added by reducing the stretching factor as shown in Table 3.1.

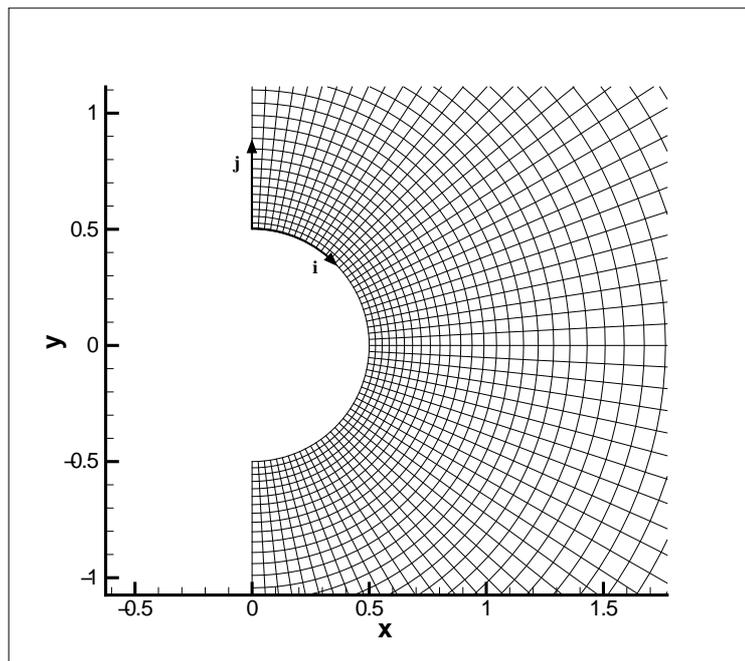
For this test case the flow governing equations are solved in conservative form to allow coupling with the Kim and Lee type of boundaries of Section 2.5.1. They are non-dimensionalised for the computation following the rules given in Section 2.1.4. The set of reference values shown in Table 3.2 is used. The normalisation of the input data and of the results (in primitive form) uses  $l_\infty$ ,  $\rho_\infty$ , and  $c_\infty$  for normalising length, density and velocity. Pressure is normalised by  $p_{ref} = 101325$  Pa to facilitate the comparison with published data.

### 3. TEST CASE APPLICATIONS

---



(a) General structure of the grid.



(b) Detail of the area surrounding the circle representing the still cylinder. Curvilinear coordinates  $i$  and  $j$  are shown.

Figure 3.3: Coarsest grid (number 1 in Table 3.1) used for the solution of the first test case. All dimensions are normalised with respect to the cylinder diameter  $D$ .

### 3. TEST CASE APPLICATIONS

---

Table 3.1: The five grids used for the solution of the first test case. The value of the stretching factor refers to the radial direction. The smallest cell values refer to the minimum node distance along the two curvilinear coordinates and is non-dimensionalised with respect to  $D$ .

Grid no.	Nodes		Stretching factor	Smallest cell [nd]		
	in $i$ dir.	in $j$ dir.		total	in $i$ dir.	in $j$ dir.
1	61	61	3721	1.0539	2.62E-02	2.70E-02
2	121	121	14641	1.0266	1.31E-02	1.33E-02
3	241	241	58081	1.0132	6.54E-03	6.61E-03
4	481	481	231361	1.0066	3.27E-03	3.29E-03
5	961	961	923521	1.0033	1.64E-03	1.65E-03

Table 3.2: Reference values used for the code internal non-dimensionalisation.

Length ( $l_\infty$ ) [na]	Static speed of sound ( $c_\infty$ ) [m/s]	Static density ( $\rho_\infty$ ) [kg/m <sup>3</sup> ]
Cylinder diameter $D$	340.2939905434	1.225

At the beginning of the simulations, the flow field is initialised with

$$\mathbf{Q} = [\rho, u, v, p]^T = [1, 0, 0, 1]^T. \quad (3.8)$$

The surface of the cylinder is modelled with the stationary inviscid wall boundary condition proposed by Kim and Lee and reported in Section 2.5.1.2. The external semi-circular surface, that is an inflow or an outflow depending on the phase of the field oscillation, is computed with a customised closure that is function of the direction of the velocity normal to the boundary and enforces either the soft inflow or the non-reflecting outflow condition by Kim and Lee reported in Section 2.5.1.1. The non-dimensional reference values required by Equations (2.83), (2.85) and (2.86) for the

### 3. TEST CASE APPLICATIONS

---

computation of the boundary conditions are as follows:  $l = 25$  (characteristic length of the domain),  $u_\infty = 0$ ,  $v_\infty = \epsilon \sin(\omega t)$ ,  $p_\infty = 1$ .

The condition of symmetry applied to the vertical boundaries is enforced as described in Section 2.5.3. This particular type of boundary closure can only be applied to plane surfaces normal to either the  $x$  or the  $y$  axis as in this case.

When the computation is run in parallel on multiple processes, standard inter-block boundary conditions are used for the communication as described in Section 2.6.

The linearised formulation of the Euler Equations (2.8a) is an appropriate choice for the solution of this problem, so that predictions can be directly compared against the reference analytical solution by Dowling & Ffowcs Williams [1983], which is based on a linear perturbation assumption. The problem is also solved using the non-linear set of equations.

The time-integration is performed using the classical fourth-order Runge-Kutta scheme with the implementation described in Section 2.3.2.

The types of finite-difference spatial filters used for removing high-wavenumber spurious numerical waves from the computation of this test case are summarised in Table 3.3. Note that when the implicit filter is used, the value of  $\alpha_f$  is constant near the boundary as, if increased, the computations become unstable.

### 3. TEST CASE APPLICATIONS

---

Table 3.3: Characteristics of the filters used for the first test case.

<b>Filter type</b>	<b>NBT</b>	<b><math>\alpha_f</math> coeff.</b>	<b>Application frequency</b>
explicit	LOC	0	every time step
implicit	LOC	0.4	every time step

#### 3.1.3 Results

The non-dimensional duration of the simulations  $t_{fin}$  is fixed equal to 23 to allow for the waves that originate near the cylinder to propagate beyond the external boundaries. The convergence of the results is checked between time  $t = 22$  and 23. In all the simulations run, the maximum difference in non-dimensional pressure over the whole field is less than 0.3%.

The time-integration and parallelisation settings depending on the degree of mesh refinement are shown in Table 3.4. With the sixth-order prefactored compact scheme and the classical fourth-order explicit Runge-Kutta time-integration, the stability and the accuracy limits for the Courant number are respectively equal to 0.4999 and 1.4217. The simulations on grid number 3, 4, and 5 in Table 3.1 use a relatively high number of nodes and time steps and therefore are run in a parallel fashion by dividing the model in equal parts as summarised in Table 3.4.

Four different sets of simulations have been run by varying the form of the equation (linearised or non-linear) and the type of filtering (explicit or implicit, see Table 3.3 for details). To check the order of the mesh convergence, each set is composed of five runs on the progressively refined grids of Table 3.1.

The linearised and non-linear sets of simulations lead to visually identical results,

### 3. TEST CASE APPLICATIONS

---

Table 3.4: Time integration and parallelisation settings for the numerical grids of Table 3.1. The time step is in non-dimensional form.

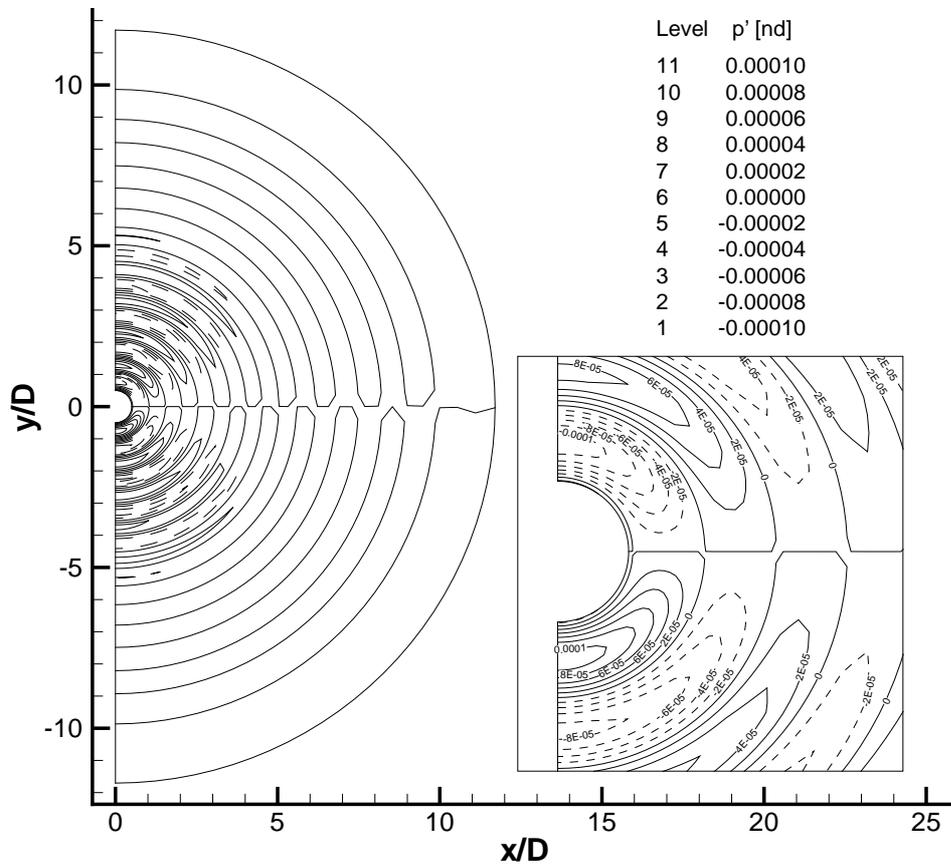
Grid no.	Time step [nd]	Steps	Courant no.	Processes		
				in $i$ dir.	in $j$ dir.	total
1	0.0125	1920	0.4775	1	1	1
2	0.00625	3840	0.4774	1	1	1
3	0.003125	7680	0.4774	2	1	2
4	0.0015625	15360	0.4774	2	2	4
5	0.00078125	30720	0.4774	4	2	8

so, only those relative to linearised equations with implicit filtering are shown in Figures 3.4-3.6. All the figures are obtained at the non-dimensional time  $t = 22$  and show flow variables in non-dimensional form.

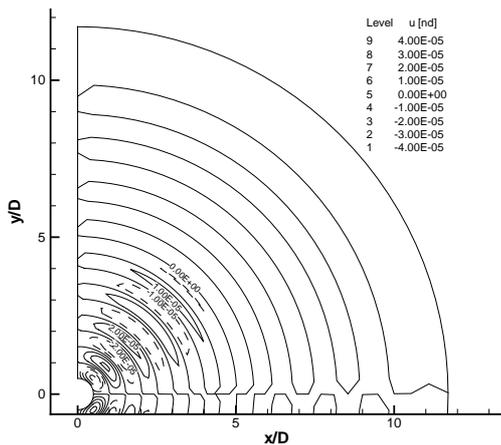
Figure 3.4 shows the flow state at  $t = 22$  from simulation with grid number 1. The general shape of the acoustic non-dimensional pressure field is that of a dipole with the main directivity along axis  $y$ . These results can be directly compared with those obtained from grid number 5 shown in Figure 3.5. The higher spatial resolution of the latter simulation is shown near the external boundary where the grids are locally more coarse, due to the proportional mesh stretching of Table 3.1. These observations are confirmed by the details (b) and (c) of the two figures that regard the non-dimensional velocity components  $u$  and  $v$ .

In Figure 3.6, the pressure distribution along positive axis  $y$  obtained from the five different grids are compared. The numerical results, represented by symbols, are clustered around the analytical reference solution which is shown by the continuous line. The detail allows to appreciate the small dependence of the predictions on the level of mesh refinement.

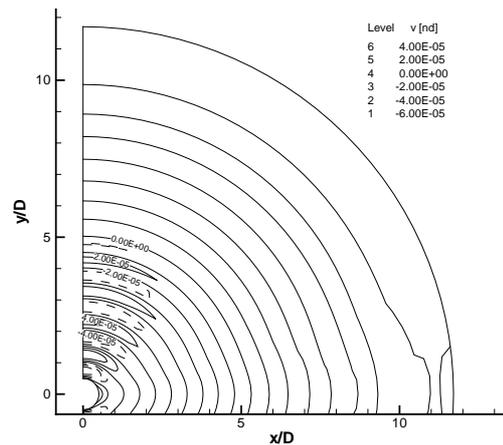
### 3. TEST CASE APPLICATIONS



(a) Non-dimensional pressure oscillation.



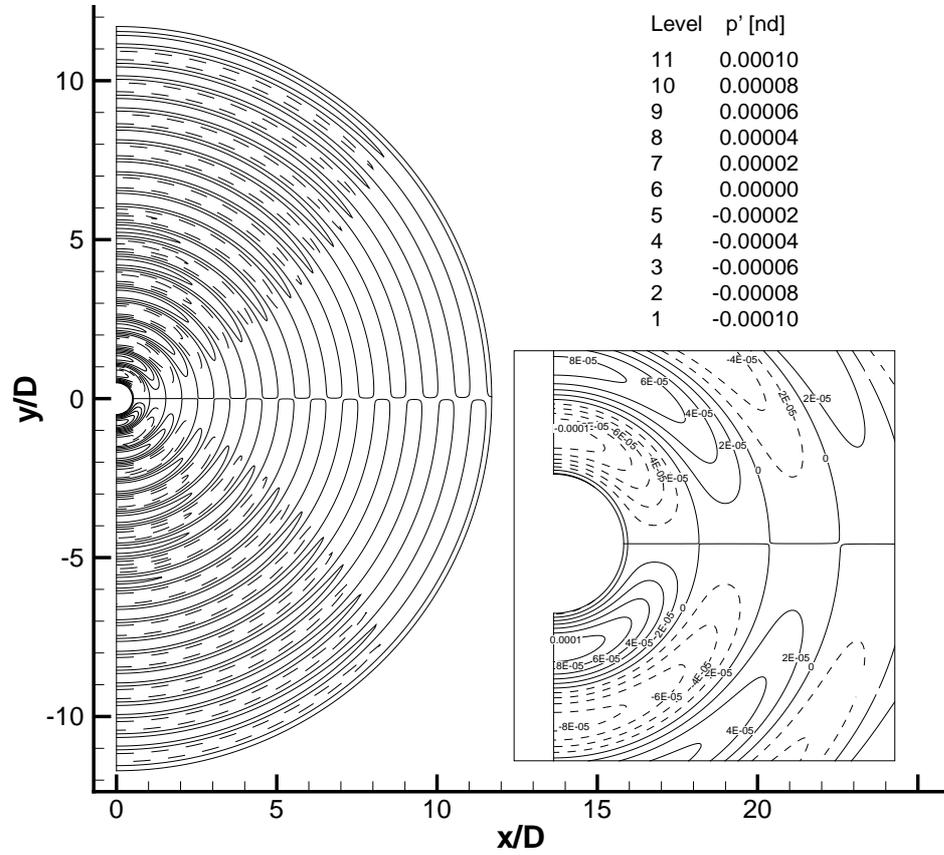
(b) Non-dimensional  $u$  velocity.



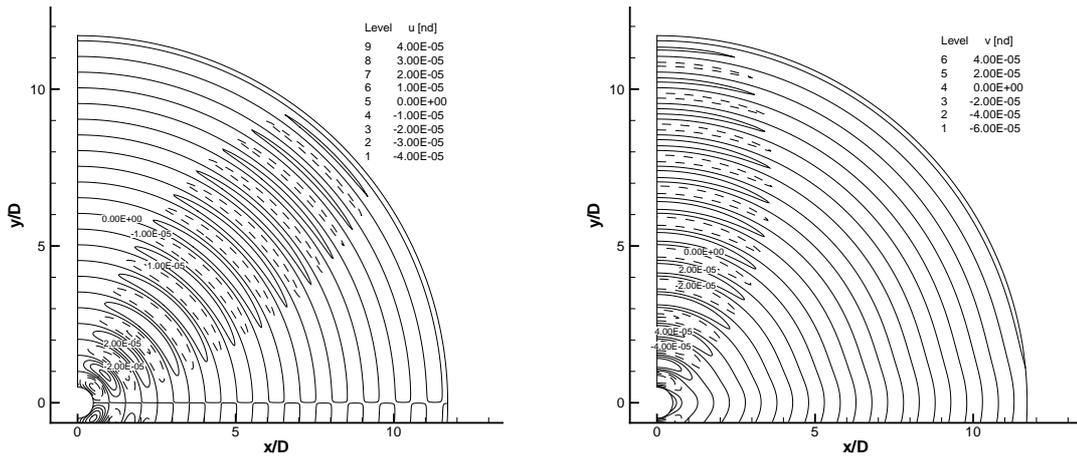
(c) Non-dimensional  $v$  velocity.

Figure 3.4: Non-dimensional predicted flow state at non-dimensional time  $t = 22$  using equations in linearised form and implicit filtering. Grid number 1. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS



(a) Non-dimensional pressure oscillation.



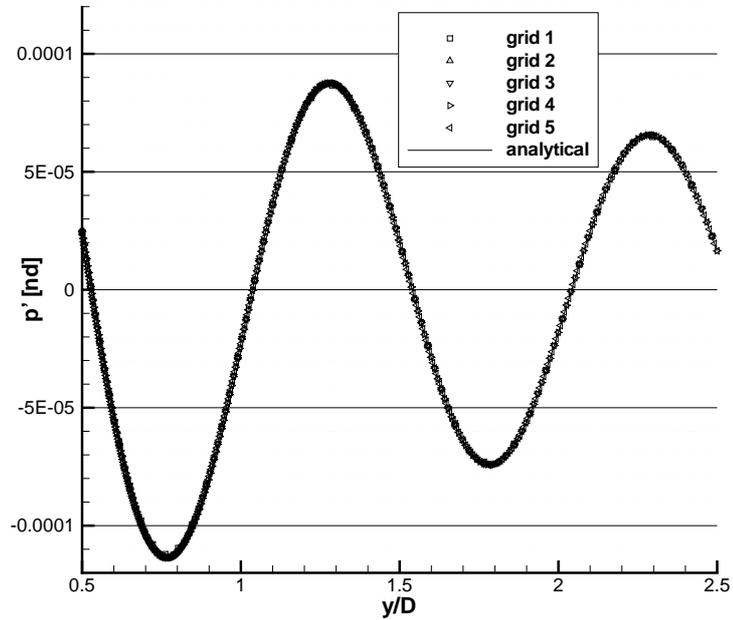
(b) Non-dimensional  $u$  velocity.

(c) Non-dimensional  $v$  velocity.

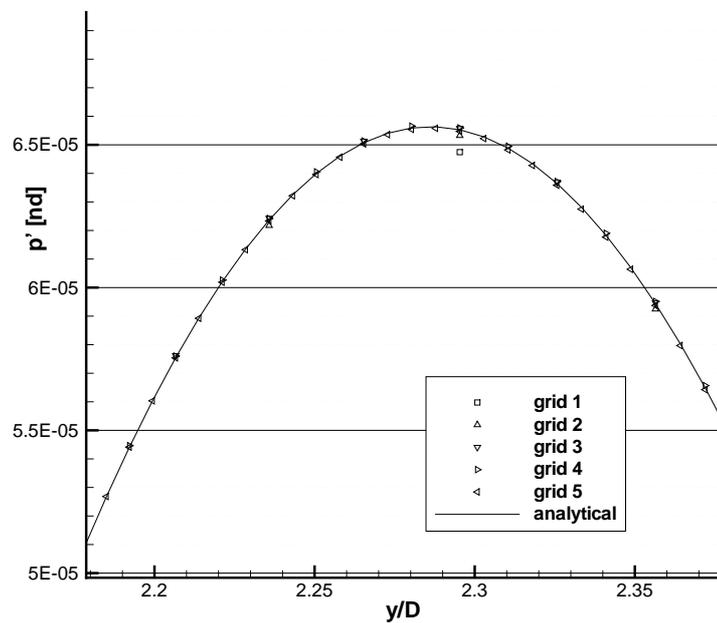
Figure 3.5: Non-dimensional predicted flow state at non-dimensional time  $t = 22$  using equations in linearised form and implicit filtering. Grid number 5. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---



(a) Non-dimensional pressure oscillation.



(b) Detail of the second peak.

Figure 3.6: Results at non-dimensional time  $t = 22$  along the positive  $y$  axis. The results from simulations on five different grids are compared to the analytical solution. Equations in linearised form with implicit filtering.

### 3. TEST CASE APPLICATIONS

---

In Table 3.5, the non-dimensional pressure error at time  $t = 6$  is shown as both the  $l_\infty$ -norm, defined by Equation (2.112), and the  $l_2$ -norm defined by Equation (2.111). The values are compared node-by-node with the analytical solution of Equation (3.3). The results are grouped in four sets, using the five different numerical grids of Table 3.1 in each set in order to evaluate the mesh convergence. Each set of results comes from a combination of filter type (implicit or explicit) and equation form (linearised or non-linear). At time  $t = 6$ , the front of the first wave propagating from the cylinder has not yet reached the external boundary. Thus the comparison between the analytical and the numerical solution is restricted to a circle of radius  $R = 1$  centred on the origin of the axes. This region includes the first pressure peak only, as shown in Figures 3.4, 3.5, and 3.6. The analytical solution of Equation (3.3) was obtained under the hypothesis of linearity. Therefore, correctly, the results indicate that the equations in linearised form give a lower error with respect to their non-linear counterparts. The implicit form of the spatial filter used to damp high-frequency numerical waves in the computational domain gives a flow simulation that is in closer agreement with the analytical solution in terms of absolute non-dimensional pressure value on all five grids.

The examination of the order of convergence of the error norms against the mesh density shows a second-order roll off (columns “conv.”). This result is not in accordance with the expected sixth order of the internal spatial scheme of Section 2.2 and can be explained by the use of the low-order centred (LOC) technique, described Section 2.2.3, to filter the flow field on the nodes near the boundaries. In fact, the first interior point of the computational domain at the circular cylinder boundary is treated with a second-order three-point stencil filter. In this type of problems the waves are originated on the cylinder surface and they necessarily have to pass through the low-order filter as they propagate inside the domain. This affects the wave amplitude.

### 3. TEST CASE APPLICATIONS

---

In Table 3.6, the results at the non-dimensional time  $t = 22$  with comparison restricted to a circle of radius  $R = 1$  are summarised. Here the influence of the error induced by the external boundary becomes apparent on the absolute accuracy of the non-dimensional pressure. The finer meshes are more affected by this effect.

At the non-dimensional time  $t = 22$  the waves have reached the external boundaries and the error comparison with the analytical solution can be extended to the whole field. The results are summarised in Table 3.7. Both error norms increase with respect to the norms computed over the range  $0.5 < R < 1$  of Table 3.6. The accuracy approaching the external boundaries is lower due to the influence of the boundary closure and to the lower node density. In contrast to Tables 3.5 and 3.6, the order of convergence of the  $l_\infty$ -norm and of the  $l_2$ -norm here behave differently. The  $l_\infty$ -norm, depends only on the error peak and is always located on the external boundary. This norm increases with the grid refinement level, while  $l_2$ -norm, which is function of all the nodes in the field, is more or less grid independent.

An attempt has been made to overcome the limitations given by the use of the LOC boundary treatment by the use of high-order one-sided filters (HOOS), but without success due to the instabilities moving from the boundary that eventually lead to the failure of the simulation. The spectral functions of these filters, shown in Equation (2.63) have a non-zero imaginary part [Visbal & Gaitonde, 1998] and, consequently, the amplification ratio over a range of wavenumbers is greater than 1. This makes the resulting numerical method less stable than with lower-order centred filters.

Other attempts to stabilising the computation with HOOS reducing the time step by up to ten times still give numerical instability.

A last attempt was made by reducing the non-dimensional amplitude of the oscil-

### 3. TEST CASE APPLICATIONS

---

Table 3.5: Comparison of the difference between the numerical and the analytical results for the non-dimensional pressure at time  $t = 6$  within a circle of radius  $R = 1$  centred on the origin of the axes. Results from combinations of linearised or non-linear equations and implicit or explicit filters. The absolute value of the error is shown both as  $l_\infty$ -norm and  $l_2$ -norm along with their order of convergence against the grids with increasing spatial refinement from Table 3.1.

Grid no.	Eq. form	Filter	$l_\infty$	conv.	$l_2$	conv.
1	non-lin.	expl.	2.81E-06		1.36E-06	
2	non-lin.	expl.	6.63E-07	2.11	3.18E-07	2.12
3	non-lin.	expl.	1.68E-07	2.00	7.74E-08	2.05
4	non-lin.	expl.	5.02E-08	1.75	2.04E-08	1.93
5	non-lin.	expl.	2.25E-08	1.16	7.31E-09	1.48
1	lin.	expl.	2.81E-06		1.36E-06	
2	lin.	expl.	6.60E-07	2.12	3.18E-07	2.12
3	lin.	expl.	1.65E-07	2.02	7.74E-08	2.05
4	lin.	expl.	4.68E-08	1.82	2.01E-08	1.95
5	lin.	expl.	1.82E-08	1.37	6.61E-09	1.61
1	non-lin.	impl.	1.43E-06		6.91E-07	
2	non-lin.	impl.	3.44E-07	2.08	1.63E-07	2.11
3	non-lin.	impl.	9.16E-08	1.92	4.06E-08	2.02
4	non-lin.	impl.	3.17E-08	1.54	1.17E-08	1.80
5	non-lin.	impl.	2.08E-08	0.61	5.61E-09	1.07
1	lin.	impl.	1.43E-06		6.91E-07	
2	lin.	impl.	3.41E-07	2.09	1.63E-07	2.11
3	lin.	impl.	8.86E-08	1.96	4.05E-08	2.02
4	lin.	impl.	2.83E-08	1.65	1.13E-08	1.85
5	lin.	impl.	1.36E-08	1.06	4.65E-09	1.29

### 3. TEST CASE APPLICATIONS

---

Table 3.6: Comparison of the difference between the numerical and the analytical results for the non-dimensional pressure at time  $t = 22$  within a circle of radius  $R = 1$  centred on the origin of the axes. Results from combinations of linearised or non-linear equations and implicit or explicit filters. The absolute value of the error is shown both as  $l_\infty$ -norm and  $l_2$ -norm along with their order of convergence against the grids with increasing spatial refinement from Table 3.1.

Grid no.	Eq. form	Filter	$l_\infty$	conv.	$l_2$	conv.
1	non-lin.	expl.	2.76E-06		1.32E-06	
2	non-lin.	expl.	7.06E-07	1.99	3.35E-07	2.01
3	non-lin.	expl.	1.98E-07	1.84	8.96E-08	1.91
4	non-lin.	expl.	1.63E-07	0.28	7.25E-08	0.31
5	non-lin.	expl.	1.52E-07	0.10	3.70E-08	0.97
1	lin.	expl.	2.76E-06		1.32E-06	
2	lin.	expl.	7.06E-07	1.99	3.35E-07	2.01
3	lin.	expl.	2.02E-07	1.82	9.00E-08	1.91
4	lin.	expl.	1.65E-07	0.29	7.28E-08	0.31
5	lin.	expl.	1.68E-07	-0.03	3.73E-08	0.96
1	non-lin.	impl.	1.41E-06		6.85E-07	
2	non-lin.	impl.	3.90E-07	1.88	1.93E-07	1.85
3	non-lin.	impl.	2.95E-07	0.41	1.12E-07	0.78
4	non-lin.	impl.	1.93E-07	0.61	6.42E-08	0.81
5	non-lin.	impl.	1.41E-07	0.46	2.94E-08	1.13
1	lin.	impl.	1.41E-06		6.85E-07	
2	lin.	impl.	3.91E-07	1.88	1.93E-07	1.85
3	lin.	impl.	2.94E-07	0.41	1.13E-07	0.78
4	lin.	impl.	2.11E-07	0.48	6.45E-08	0.81
5	lin.	impl.	1.54E-07	0.45	2.97E-08	1.12

### 3. TEST CASE APPLICATIONS

---

Table 3.7: Comparison of the difference between the numerical and the analytical results for the non-dimensional pressure at time  $t = 22$  on the whole field. Results from combinations of linearised or non-linear equations and implicit or explicit filters. The absolute value of the error is shown both as  $l_\infty$ -norm and  $l_2$ -norm along with their order of convergence against the grids with increasing spatial refinement from Table 3.1.

Grid no.	Eq. form	Filter	$l_\infty$	conv.	$l_2$	conv.
1	non-lin.	expl.	4.12E-05		9.15E-06	
2	non-lin.	expl.	3.30E-05	0.32	5.61E-06	0.71
3	non-lin.	expl.	2.97E-05	0.15	1.58E-06	1.84
4	non-lin.	expl.	1.48E-05	1.00	6.52E-07	1.28
5	non-lin.	expl.	5.46E-06	1.44	1.94E-07	1.75
1	lin.	expl.	4.12E-05		9.15E-06	
2	lin.	expl.	3.30E-05	0.32	5.61E-06	0.71
3	lin.	expl.	2.97E-05	0.15	1.58E-06	1.84
4	lin.	expl.	1.49E-05	1.00	6.52E-07	1.28
5	lin.	expl.	5.49E-06	1.44	1.96E-07	1.74
1	non-lin.	impl.	3.95E-05		7.49E-06	
2	non-lin.	impl.	2.80E-05	0.50	4.01E-06	0.91
3	non-lin.	impl.	2.66E-05	0.08	1.33E-06	1.61
4	non-lin.	impl.	1.32E-05	1.01	4.78E-07	1.47
5	non-lin.	impl.	4.98E-06	1.41	1.65E-07	1.53
1	lin.	impl.	3.95E-05		7.49E-06	
2	lin.	impl.	2.80E-05	0.50	4.01E-06	0.91
3	lin.	impl.	2.66E-05	0.08	1.33E-06	1.60
4	lin.	impl.	1.33E-05	1.01	4.79E-07	1.47
5	lin.	impl.	5.02E-06	1.40	1.67E-07	1.52

### 3. TEST CASE APPLICATIONS

---

lation  $\epsilon$  to  $10^{-7}$  while keeping a non-dimensional formulation and explicit filtering. The results at non-dimensional time  $t = 13$  from grids 1 to 4 are shown in Table 3.8. The non-dimensional pressure amplitude is correctly reduced by 3 orders of magnitude but, apart from a slight increase in the order of accuracy, probably due to the reduced influence of the non-linear effects, these results do not show any significant difference compared to the ones in Table 3.7.

Table 3.8: Comparison of the difference between the numerical and the analytical results for the non-dimensional pressure at non-dimensional time  $t = 13$  with the input  $v$  velocity oscillation amplitude  $\epsilon$  reduced to  $10^{-7}$ . The equations are solved numerically in non-dimensional form with explicit filtering. The radius over which the comparison is made is indicated by the value  $R_{lim}$ . The absolute value of the error is shown both as  $l_\infty$ -norm and  $l_2$ -norm along with their order of convergence against the grids with increasing spatial refinement from Table 3.1.

<b>Grid no.</b>	<b><math>R_{lim}</math></b>	<b><math>l_\infty</math></b>	<b>conv.</b>	<b><math>l_2</math></b>	<b>conv.</b>
1	1	2.96E-09		1.36E-09	
2	1	6.53E-10	2.20	3.17E-10	2.13
3	1	1.56E-10	2.07	7.61E-11	2.07
4	1	3.86E-11	2.02	1.88E-11	2.02
1	$R_{ext}$	4.13E-08		9.15E-09	
2	$R_{ext}$	3.31E-08	0.32	5.61E-09	0.71
3	$R_{ext}$	2.95E-08	0.17	1.57E-09	1.85
4	$R_{ext}$	1.47E-08	1.01	6.53E-10	1.27

## 3.2 Second test case: sound generation by interaction between an incident gust and a cascade of flat plates (2D)

### 3.2.1 Problem definition

The second test case is a benchmark problem from the second NASA computational aeroacoustics (CAA) workshop [Tam & Hardin, 1997]. It is the second problem under category 3: “Turbomachinery noise” and aims to model, in terms of tonal noise emission, the interaction between a rotor and a stator in turbomachinery.

The model is simplified with respect to the real turbomachine in a number of ways: (i) the 3D of a turbomachine stage is unrolled in two dimensions, (ii) the blade profile is modelled as a simple flat plate, (iii) the wakes from the upstream blade row are replaced by a vortical, divergence free, convected *frozen* gust defined at the left inflow boundary that aims to reproduce the turbulence contained in the wakes. The geometry of the problem is shown in Figure 3.7, where  $c$  is the chord length of the plate and  $u_\infty$  is the mean inflow velocity in the direction of the  $x$  axis. The inter-blade distance  $g$  is equal to  $c$  and the field extends from  $x = -2c$  to  $x = 3c$  and from  $y = 0$  to  $y = 4c$ , to include four plates. The initial flow field is uniform with mean static density  $\rho_\infty$ . The problem prescribes the use of the following set of reference variables for the non-dimensionalisation: velocity  $u_\infty$ , length  $c$ , time  $c/u_\infty$ , density  $\rho_\infty$ , and pressure  $\rho_\infty u_\infty^2$ . The Mach number at the inflow is  $M_\infty = 0.5$ .

### 3. TEST CASE APPLICATIONS

---

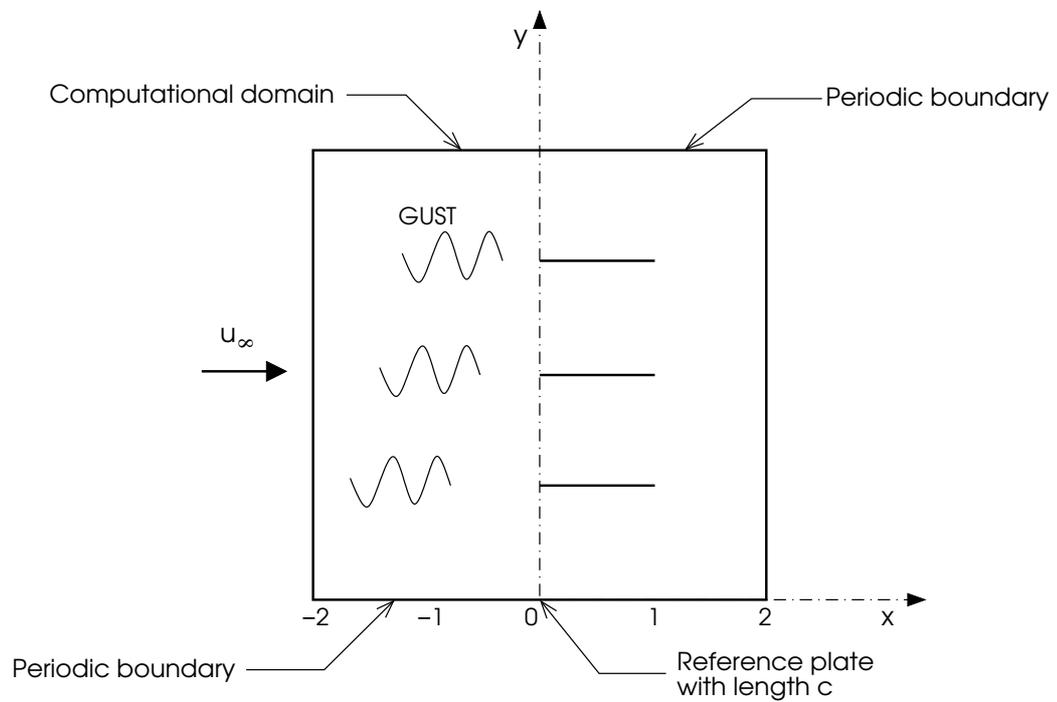


Figure 3.7: Problem geometry of test case number 2 (modified from Tam & Hardin [1997]).

### 3. TEST CASE APPLICATIONS

---

The incoming vorticity gust is a function of the coordinates and of time as

$$u'_g(x, y, t) = -\frac{v_G \beta}{\alpha} \cos(\alpha x + \beta y - \omega t), \quad (3.9a)$$

$$v'_g(x, y, t) = v_G \cos(\alpha x + \beta y - \omega t), \quad (3.9b)$$

$$\rho'_g(x, y, t) = 0, \quad (3.9c)$$

$$p'_g(x, y, t) = 0, \quad (3.9d)$$

where  $v_G = 0.01$  and the non-dimensional wavenumbers  $\alpha$  and  $\beta$  are equal to the non-dimensional angular frequency  $\omega$ . The problem requires the solution of two different input frequencies: a low-frequency gust with  $\omega = 5\pi/2$  and high-frequency gust with  $\omega = 13\pi/2$  but, similarly to other workshop contributors [Tam & Hardin, 1997], in this work only the low-frequency case is treated. Either the linearised or the non-linear formulation of the Euler equations can be used.

The problem requires to predict the pressure difference  $\Delta p = p_{lower} - p_{upper}$  between the upper and the lower surface of the reference plate along  $y = 0$ . In addition, the intensity of the radiated sound  $\overline{p^2}$  has to be determined along lines  $x/c = -2$  and  $x/c = +3$  and the final non-dimensional pressure distribution over the whole computational field at time  $t = 2\pi n/\omega$ , with  $n$  integer, has to be plotted. The problem requires the solution to be periodic in time but does not prescribe within which limits this periodicity has to be obtained.

An analytical solution for the problem was published by Smith [1972] but its numerical computation requires the truncation of infinite sums and numerical quadrature. This makes this benchmark approximate. Hall [1997a] provides a numerical solution in tabular form computed by a computer code called LINSUB based on Smith's the-

### 3. TEST CASE APPLICATIONS

---

ory. This solution is used in the results Section 3.2.3 as a reference to compare against both for the pressure jump across the blade and for the value of  $\overline{p^2}$  at  $x/c = -2$  and  $x/c = +3$ .

#### 3.2.2 Numerical methods

The equations and the data input and output are non-dimensionalised using the set of reference values defined in Tam & Hardin [1997] and reported in Section 3.2.1.

The problem definition does not give any constraint on the characteristics of the grids to be used, and only specifies a minimum computational domain extent. In the current study, only orthogonal Cartesian uniform meshes equally spaced in both the  $x$  and  $y$  direction are used. The spacing in the  $x$  direction is kept equal to that in the  $y$  direction, so that the grid geometry is fully determined by the inter-nodal distance  $\Delta x$  and by the total field length  $L$ . A third parameter ( $x_{in}$ ) is necessary to determine the position of the blades along the  $x$  axis, specifying the distance between the inflow plane and the leading edge of the plates.  $x_{in}$  does not affect the grid structure. In the  $y$  direction the position of the reference blade is equal to 0 so that no further parameter is needed. The length of the field  $L$  in the  $x$  direction is extended beyond the minimum defined by the problem description as preliminary tests have shown that placing the inflow and outflow on the monitoring planes  $x/c = -2$  and  $x/c = +3$  leads to predictions being adversely affected by the boundary closure treatment. Furthermore, some types of boundary conditions cannot be applied without the presence of additional nodes. Extending the computational domain beyond  $-2 < x/c < +3$  is a common practice among the authors that have published solutions to this problem.

Non-uniform grids are not used in the results presented in this section to minimise

### 3. TEST CASE APPLICATIONS

---

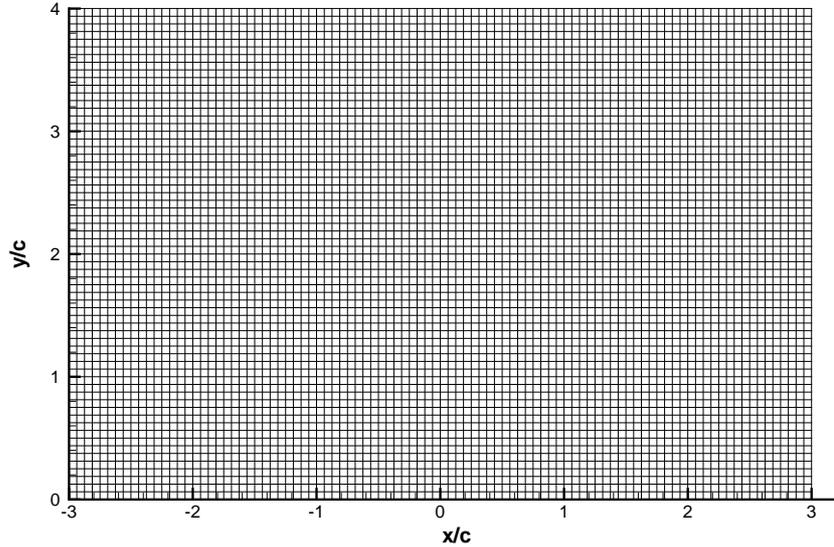


Figure 3.8: General structure of grid 6.1 in Table 3.9.

the number of variables affecting the accuracy of the results. Non-uniform meshes are used for the solution of test cases 1, 3 and 4 where the nature of the problem requires their adoption. This choice is not economical in terms of computational resources, as the nodes could be clustered around the plates, where the essential physical process of conversion from flow wake kinetic energy to radiating noise emission takes place.

The computational mesh parameters adopted in this study are summarised in Table 3.9. Each grid is labelled with the common notation  $L.i$  which identifies the values of  $L$  and  $\Delta x_i$  used to generate each grid.

The set of grids of length  $L = 6$  is used for preliminary tests on how the inlet gust convects across the computational domain in the absence of the blades. The remaining grids are used for the solution of the problem including the cascade. Figure 3.8 shows grid 6.1, which is the mesh of extent  $L = 6$  with the coarsest (level 1) discretisation.

In the  $y$  direction, an inviscid solid wall condition is applied over the range  $0 \leq$

### 3. TEST CASE APPLICATIONS

---

Table 3.9: Characteristics of the fifteen grids used for the solution of the second test case.  $L$ ,  $\Delta x$  and the nodes per unit length are in non-dimensional form. The name of the grid univocally determines its characteristics and it is given by the total length of the field and a progressive index related to the grid density. The fields related to the total number of nodes “model” and “computation” refer, respectively, to the nodes of the initial model and to the nodes of the actual parallel computation. The two values are different as the nodes on the common borders are duplicated in the neighbouring processes.

<b>Grid name</b>	<b>Length <math>L</math> [nd]</b>	<b><math>\Delta x</math> [nd]</b>	<b>Nodes per unit length [nd]</b>	<b>Total no. of nodes</b>	
				<b>model</b>	<b>computation</b>
<b>6.1</b>	6	0.0625	16	6305	6936
<b>6.2</b>	6	0.03125	32	24897	26136
<b>6.3</b>	6	0.015625	64	98945	101400
<b>7.1</b>	7	0.0625	16	7345	8092
<b>7.2</b>	7	0.03125	32	29025	30492
<b>7.3</b>	7	0.015625	64	115393	118300
<b>7.4</b>	7	0.0078125	128	460161	465948
<b>9.1</b>	9	0.0625	16	9425	10404
<b>9.2</b>	9	0.03125	32	37281	39204
<b>9.3</b>	9	0.015625	64	148289	152100
<b>9.4</b>	9	0.0078125	128	591489	599076
<b>11.1</b>	11	0.0625	16	11505	12716
<b>11.2</b>	11	0.03125	32	45537	47916
<b>11.3</b>	11	0.015625	64	181185	185900
<b>11.4</b>	11	0.0078125	128	722817	732204

$x/c \leq 1$  and a periodic boundary condition is imposed at  $x/c < 0$  and  $x/c > +1$ . The periodic boundary condition is numerically equivalent to a standard inter-block boundary exchanging the data of five rings of nodes with the neighbour. In the  $x$  direction, various boundary closures have been tested at the inlet and the outlet of the computational domain. At the inflow boundary, five different closures are available: (i) buffer zone, (ii) Kim and Lee soft (non-reflecting) inflow, (iii) Giles subsonic non-reflecting inflow, (iv) modified Giles subsonic non-reflecting inflow, (v) modified buffer zone.

### 3. TEST CASE APPLICATIONS

---

At the outflow, a variety of boundary condition treatments is available: (i) buffer zone, (ii) modified buffer zone, (iii) Kim and Lee non-reflecting outflow, (iv) Giles subsonic non-reflecting outflow.

For all the simulations presented in Section 3.2.3, the buffer length is varied based on the mesh density of the model so to keep the same spatial extension. For grid density 1, the buffer layer depth is 30 nodes, for grid density 2 it is 60 nodes, for grid density 3 it is 120 nodes and, finally, for grid density 4 it is 240 nodes.

The Kim and Lee soft inflow and non-reflecting outflow are described in detail in Section 2.5.1.1, the various versions of the Giles closures in Section 2.5.2, and the buffer layer and its modified version in Section 2.5.4.

The flow state in the modified buffer layer is set by Equation (2.108a) using a target flow state  $\mathbf{Q}_{\text{target}}$  that is equal to the sum of the uniform time-invariant  $M_\infty = 0.5$  plus the gust of Equation (3.9) superimposed on it.

The blade surface is modelled as an inviscid wall. Two inviscid wall boundary conditions have been included in the code which are the inviscid wall condition by Tam & Dong [1996] and that based on the theory of characteristics supplied by Kim & Lee [2004]. The former is used in association with the primitive form of the Euler equations and the latter with their conservative formulation.

For this test case, the specification of the boundary conditions and the introduction of the vortical gust are provided by a custom unsteady inflow condition, which can be implemented as either a modified Giles or a modified buffer inflow condition. A third technique is available following Lockard & Morris [1997]. In this case, the oscillation is not introduced from the boundary but it is directly generated in the computational domain interior by applying an appropriate body force vector  $\mathbf{f}$  (see Equations (2.1)

### 3. TEST CASE APPLICATIONS

---

and (2.2)). The problem consists in finding a function  $\Psi$  that satisfies the momentum equations in 2D primitive form when  $u$  and  $v$  include the vortical gust

$$\frac{Du}{Dt} + \frac{1}{\rho} \frac{\partial p}{\partial x} = \frac{\partial \Psi}{\partial y} \sin(\omega t), \quad (3.10a)$$

$$\frac{Dv}{Dt} + \frac{1}{\rho} \frac{\partial p}{\partial y} = -\frac{\partial \Psi}{\partial x} \sin(\omega t). \quad (3.10b)$$

This function is

$$\Psi = \frac{v_G}{\pi} \{1 + \cos [0.5\alpha (x - x_G)]\} \cos(\alpha x + \beta y), \quad (3.11)$$

and it is valid in the range:  $|x - x_G| < 2\pi/\alpha$ .  $x_G$  is the  $x$  coordinate of the midline of the zone where the gust is introduced.

The resulting body force components are

$$f_x = -\beta \frac{v_G}{\pi} \{1 + \cos [0.5\alpha (x - x_G)]\} \sin(\alpha x + \beta y) \sin(\omega t), \quad (3.12)$$

$$f_y = \alpha \frac{v_G}{\pi} \{(1 + \cos [0.5\alpha (x - x_G)]) \sin(\alpha x + \beta y) + 0.5 \sin [0.5\alpha (x - x_G)] \cos(\alpha x + \beta y)\} \sin(\omega t). \quad (3.13)$$

All the simulations presented in this work are run in parallel by subdividing the computational domain in equal squared parts connected by standard inter-block boundary conditions as detailed in Section 2.6. The squared shape of the parts minimises the amount of inter-communication while their approximately equal size in terms of nodes of the computational grid tends to homogeneously distribute the computational effort,

### 3. TEST CASE APPLICATIONS

---

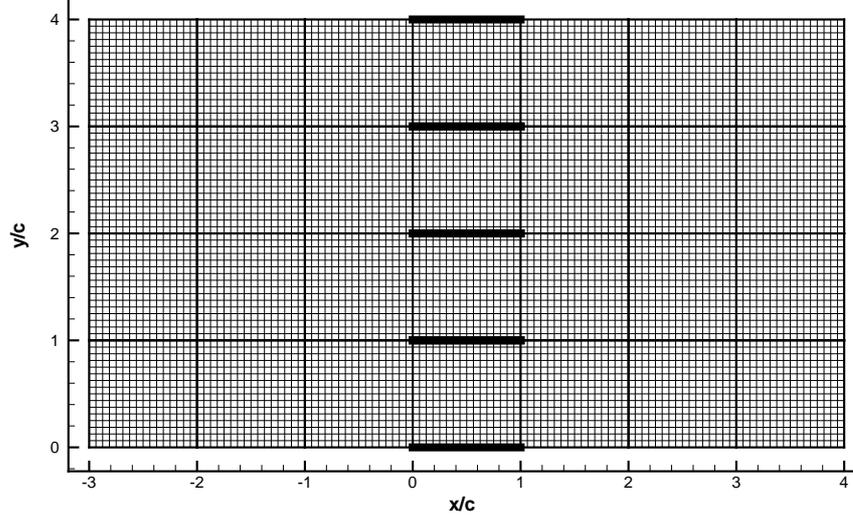


Figure 3.9: Grid 7.1 subdivided into 28 zones delimited by thicker borders. The position of the cascade is shown by horizontal black lines over the range  $0 \leq x/c \leq 1$ .

although the processes including the external boundaries have to perform additional computations with respect to the other processes. The parallel execution is also chosen to avoid exceptions in the treatment of the boundary conditions that, apart from the corrections related to the presence of the plate edges introduced only in some simulations, use a uniform condition along each block boundary edge. This allows to keep the code simple and efficient. In Figure 3.9 grid 7.1 is shown divided into 28 same sized blocks. The position of the cascade of flat plates is highlighted with thick black lines.

The initial conditions at the start of the computation depend on the type of gust generation method used. When the body force approach of Equations (3.12) and (3.13) is used, the initial conditions are homogeneous on the whole field and, in non-dimensional primitive form

$$\mathbf{Q}_\infty = [\rho_\infty, u_\infty, v_\infty, \rho_\infty u_\infty^2]^T = [1, 1, 0, 4/1.4]^T. \quad (3.14)$$

### 3. TEST CASE APPLICATIONS

---

On the other hand, when the gust is introduced from the boundary inflow using the modified Giles or the modified buffer inflow, it is advantageous to start from a field already including the vortical gust by superimposing the oscillating part of Equation (3.9) over the homogeneous field of Equation (3.14). It is found that the initial transient necessary to reach the periodicity in the computed flow field, with this modification was considerably shortened.

Some of the simulations run for this test case include a series of exceptions in the treatment of the nodes lying on the plates. This is required to deal with the strong discontinuity that appears as a singularity in the analytical solution of the problem at the plate leading edge, which affects the pressure distribution in the surrounding area.

There is also a second reason for the special treatment of these nodes close to the plate edges. Figure 3.9 shows that the leading and trailing edges of the cascade are always coincident to the corner nodes of four zones that belong to four different processes (those located between successive plates). While these processes apply the inviscid wall boundary condition on the corner nodes, their neighbours treat the same nodes as inter-block boundary points with no wall condition. This originates a difference in the flow field values at the nodes that should represent the same physical flow state. The situation is illustrated in Figure 3.10, where the nodes belonging to two neighbouring processes at a plate leading edge (bottom-side) are represented by black circles. Point -1 and point 1 represent the same physical locations in the flow field duplicated on two processes. These locations ought to have the same flow state, but for node -1 the spatial derivatives in the  $y$  direction are computed using the 11-point inter-block boundary stencils while for node 1 they are computed with the one-sided stencils and then corrected by the inviscid wall boundary condition that is used at the

### 3. TEST CASE APPLICATIONS

---

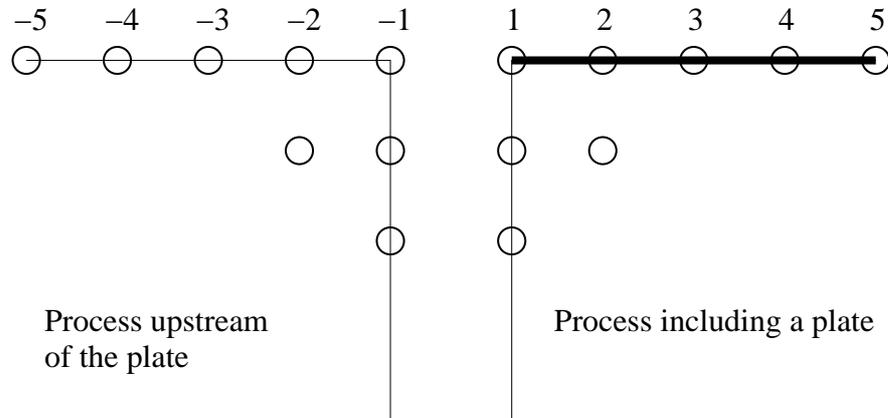


Figure 3.10: Schematic illustrating the node distribution over abutting processes upstream and below one plate.

plate leading edge. This causes a discontinuity in the flow field that interests mainly a small region around the edge but negatively affects the accuracy of the computation. It also contributes to the generation of high-frequency oscillations that the high-order finite-difference scheme, by its own nature, tends to develop in correspondence of geometrical discontinuities and steep gradients in the flow field.

For this reason, different types of numerical corrections have been tested and the best results were obtained using the following four stages:

1. The process that computes the inviscid wall condition, which is the right-hand side process in Figure 3.10, passes the half-derivatives computed in the  $y$  direction on the corner nodes (already corrected by the wall boundary treatment) to its neighbour process, which is the process upstream of the plate in Figure 3.10. This process overwrites any  $y$  derivative value it previously computed.
2. The processes upstream and downstream of the blades apply the filtering on the corner node in the  $y$  direction as if the node were part of a wall (and not an inter-block boundary).

### 3. TEST CASE APPLICATIONS

---

3. The derivatives in the  $x$  direction of the nodes near the plate edges, which are points -3, -2, 2 and 3 in Figure 3.10, are computed with one sided high-order stencils that do not “cross” the discontinuity represented by the corner. Specifically, the (5-1) 7-point stencil is applied on points -2 and 2 and the (4-2) 7-point stencil is applied on points -3 and 3. Details regarding these stencils and their coefficients can be found in Section 2.2.2.
4. Over the whole plate surface, the half-derivatives in the  $y$  direction of the pressure and the velocity component  $v$ , computed before the enforcement of the inviscid wall condition are not obtained with the typical high-order one-sided 7-point stencils. Instead, their order is dropped respectively to 1<sup>st</sup> and 3<sup>rd</sup> by the use of 2-point and 4-point one-sided stencils described in Section 2.2.2.

Stages 1 and 2 ensure that the flow field around the plate edges is matched on both processes while stages 3 and 4 aim to reduce the high-wavenumber spurious oscillations radiating from the surface discontinuity of the leading edge and are similar to the corrections adopted by Ragab & Salem-Said [2007]. In Section 3.2.3, the application of stage 1 and 2 will be referred to as correction “level 1”, while “level 2” also includes stage 3 and “level 3” includes all 4 stages.

These corrections have a computational cost related to the additional exchanges of data between the processes involved. Their cost can be evaluated by monitoring the duration of the runs.

For this test case, the linearised version of the Euler equations is used. The primitive formulation of the equations is used in conjunction with the Tam and Dong wall boundary condition, while the conservative formulation is used where the Kim and Lee inviscid wall boundary condition is enforced.

### 3. TEST CASE APPLICATIONS

---

#### 3.2.3 Results

The proceedings of the second NASA workshop include the contributions from four different research groups that have presented a solution to this problem [Hall, 1997b; Hu & Manthey, 1997; Lockard & Morris, 1997; Tam *et al.*, 1997]. Their numerical methods are both of the finite-elements and of the finite-difference type and include various techniques to avoid spurious wave reflections at the computational boundaries, introduce the gust and remove any spurious wave.

At the third and fourth NASA workshops, two groups used this problem as an initial test for their computational aeroacoustic solvers [Bin *et al.*, 2004; Wang *et al.*, 2000]. Other two studies [Hixon *et al.*, 2000; Ragab & Salem-Said, 2007] that included numerical solutions of this test case, were published independently. The study by Hixon *et al.* compares the performance of four different non-reflecting boundary conditions. Ragab & Salem-Said [2007] include the solution of this test case among other similar two-dimensional configurations.

The convergence of the non-dimensional pressure has been checked for all the simulations against the reference solution by Hall [1997a]. Over the whole computational field, the difference between the flow state at the end of the penultimate (non-dimensional time  $t = 500$ ) and of the last period (non-dimensional time  $t = 500.8$ ) is lower than 1% for all the results presented in this section, indicating that the predictions are statistically stationary.

For this test case, more than 200 simulations were performed varying a large number of computation parameters including: the type of boundary conditions, the time-

### 3. TEST CASE APPLICATIONS

---

integration scheme, the formulation of the equations (linearised/non-linear, primitive/-conservative), the type of filtering (explicit/implicit), its frequency of application and its near boundary treatment, the technique used to input the gust, the size of the computational domain, the special treatment of the plate edges, and the final time of the simulation. The aim was to determine the combination of techniques resulting in the highest accuracy. A selection of all the results is reported and discussed in two sections. Section 3.2.3.1 concerns the accuracy of the input vortical gust and Section 3.2.3.2 concerns the accuracy of the complete problem that includes the linear plates cascade.

For the preliminary simulations not including the plates, the time-integration is performed using the classical fourth-order Runge-Kutta scheme discussed in Section 2.3. On the other hand, the simulations regarding the complete problem use the Runge-Kutta scheme proposed by Berland *et al.* [2006] in the low-storage 2N form detailed in Section 2.3.2.

Due to the clearly superior results obtained in the preliminary tests with respect to the explicit formulation, all the simulations have been run applying the implicit filtering with a near boundary treatment of type low-order centred (LOC). The tunable coefficient  $\alpha_f$  linearly decreases with the distance from the boundary and assumes values between 0.495 on the first point from the wall, and 0.45. In this case the filtering is executed at the end of each Runge-Kutta stage.

#### 3.2.3.1 Accuracy of the vortical gust

The level of accuracy that can be reached in modelling the gust in the absence of a linear cascade is of interest since it tests the ability of the solver to convect vorticity waves since the amplitude of the waves impinging on the cascade is directly related to the noise emission that is studied in Section 3.2.3.2.

### 3. TEST CASE APPLICATIONS

---

Table 3.10: Grid-dependent explicit time-integration parameters for the simulations not including the linear cascade of flat plates. The time step is non-dimensionalised by  $c/u_\infty$ .

Grid no.	Time step [nd]	Steps	Courant no.
1	0.01	10000	0.4800
2	0.005	20000	0.4800
3	0.0025	40000	0.4800

The results are reported at the non-dimensional time  $t = 100$ , when the velocity disturbance has crossed the outflow boundary of the computational field so to test not only the ability to correctly input the vortical gust but also the level of error introduced by the boundary conditions at the inflow and at the outflow. When the buffer zones are used, Equation (2.108a) is applied at the end of each computational time step.

Table 3.10 shows the main parameters of the runs that are ordered by increasing computational mesh density. The CFL condition is satisfied as for this association of time-integration (classical Runge-Kutta) and of spatial scheme the accuracy limit is 0.4999 and the stability limit is 1.4217 (see Section 2.3 for details).

Table 3.11 shows the non-dimensional errors of the velocity components and of the pressure from various runs. The numerical solution is compared against the analytical definition of the gust of Equation (3.9). The error is shown as both the  $l_\infty$ -norm defined by Equation (2.112) and the  $l_2$ -norm defined by Equation (2.111). The order of convergence against the mesh points per unit length is shown for both error norms.

The analysis of the results shows that in all the cases the error in the velocity components and in the pressure is roughly two orders of magnitude lower than the amplitude of the gust  $\epsilon = 0.01$ . All the methods proposed for the introduction of the vortical gust appear engineering accurate predictions and can be deemed fit for purpose.

Table 3.11: Results from the simulations not including the linear cascade of flat plates. The non-dimensional errors of the velocity components and of the pressure with respect to the analytical definition of the gust of Equation (3.9) are shown as  $l_\infty$ -norm and  $l_2$ -norm along with their order of convergence against the mesh per unit length. The comparison is restricted to the nodes having  $x$  coordinate ranging from 0 to 1 inclusive as this is the area where the linear cascade will be introduced. The value  $x_G$  refers to the position of the midline of the zone where the gust is introduced with the method proposed by Lockard & Morris [1997] and does not apply to the other cases.

No.	Model name	Eq. form	Boundary conditions		Gust	$x_G$	u [nd]				v [nd]				p [nd]			
			$x$ low	$x$ high			$l_\infty$	conv.	$l_2$	conv.	$l_\infty$	conv.	$l_2$	conv.	$l_\infty$	conv.	$l_2$	conv.
1	6.1	prim.	buffer	buffer	Lockard	-1	5.56E-5		3.35E-5		5.77E-5		3.48E-5		3.45E-6		7.63E-7	
2	6.2	prim.	buffer	buffer	Lockard	-1	1.10E-4	-0.99	6.62E-5	-0.99	1.14E-4	-0.99	6.89E-5	-0.99	7.90E-6	-1.20	1.60E-6	-1.07
3	6.3	prim.	buffer	buffer	Lockard	-1	2.02E-4	-0.88	1.22E-4	-0.89	2.09E-4	-0.88	1.27E-4	-0.89	1.74E-5	-1.14	3.39E-6	-1.08
4	6.1	cons.	buffer	buffer	Lockard	-1	5.93E-5		3.35E-5		5.75E-5		3.46E-5		3.72E-5		2.17E-5	
5	6.2	cons.	buffer	buffer	Lockard	-1	1.13E-4	-0.93	6.61E-5	-0.99	1.13E-4	-0.99	6.87E-5	-1.00	3.96E-5	-0.09	2.16E-5	0.01
6	6.3	cons.	buffer	buffer	Lockard	-1	2.05E-4	-0.87	1.22E-4	-0.89	2.09E-4	-0.88	1.27E-4	-0.89	4.13E-5	14.19	2.14E-5	14.45
7	6.1	prim.	Giles	Giles	Lockard	-1	3.95E-4		2.74E-4		3.98E-4		2.73E-4		2.06E-5		1.06E-5	
8	6.2	prim.	Giles	Giles	Lockard	-1	2.13E-4	0.89	1.44E-4	0.93	2.16E-4	0.89	1.44E-4	0.93	2.17E-5	-0.08	1.11E-5	-0.07
9	6.3	prim.	Giles	Giles	Lockard	-1	1.18E-4	0.86	7.48E-5	0.95	1.17E-4	0.89	7.45E-5	0.96	2.21E-5	-0.02	1.12E-5	-0.01
10	6.1	cons.	Giles	Giles	Lockard	-1	3.99E-4		2.74E-4		3.99E-4		2.74E-4		5.98E-5		2.60E-5	
11	6.2	cons.	Giles	Giles	Lockard	-1	2.18E-4	0.88	1.45E-4	0.93	2.17E-4	0.89	1.45E-4	0.93	5.93E-5	0.01	2.54E-5	0.03
12	6.3	cons.	Giles	Giles	Lockard	-1	1.22E-4	0.84	7.51E-5	0.95	1.17E-4	0.89	7.46E-5	0.96	5.91E-5	0.01	2.50E-5	0.02
13	6.1	prim.	Giles	Giles	Giles	na	3.29E-4		2.32E-4		3.30E-4		2.33E-4		1.89E-6		8.94E-7	
14	6.2	prim.	Giles	Giles	Giles	na	9.89E-5	1.75	6.98E-5	1.75	9.93E-5	1.75	6.99E-5	1.75	6.32E-7	1.58	3.11E-7	1.52
15	6.3	prim.	Giles	Giles	Giles	na	2.62E-5	1.92	1.85E-5	1.92	2.63E-5	1.92	1.85E-5	1.92	1.72E-7	1.88	8.57E-8	1.86
16	6.1	cons.	Giles	Giles	Giles	na	3.30E-4		2.32E-4		3.31E-4		2.33E-4		2.14E-5		1.34E-5	
17	6.2	cons.	Giles	Giles	Giles	na	9.97E-5	1.74	6.98E-5	1.75	9.97E-5	1.74	6.99E-5	1.75	2.08E-5	0.04	1.41E-5	-0.07
18	6.3	cons.	Giles	Giles	Giles	na	2.70E-5	1.89	1.85E-5	1.92	2.66E-5	1.92	1.85E-5	1.92	2.06E-5	0.01	1.43E-5	-0.02
19	6.1	prim.	mod. buf.	mod. buf.	buffer	na	8.23E-7		4.75E-7		8.34E-7		4.75E-7		7.43E-9		1.92E-9	
20	6.2	prim.	mod. buf.	mod. buf.	buffer	na	1.35E-8	5.98	7.62E-9	6.01	1.31E-8	6.04	7.62E-9	6.01	1.10E-10	6.08	2.09E-11	6.53
21	6.3	prim.	mod. buf.	mod. buf.	buffer	na	3.04E-10	5.49	1.72E-10	5.49	2.98E-10	5.47	1.72E-10	5.49	2.02E-12	5.77	2.56E-13	6.35
22	6.1	cons.	mod. buf.	mod. buf.	buffer	na	8.48E-7		4.74E-7		8.42E-7		4.74E-7		5.01E-6		3.54E-6	
23	6.2	cons.	mod. buf.	mod. buf.	buffer	na	1.12E-7	2.94	5.69E-8	3.08	1.22E-7	2.81	6.02E-8	3.00	5.03E-6	-0.01	3.54E-6	0.00
24	6.3	cons.	mod. buf.	mod. buf.	buffer	na	1.46E-7	-0.38	7.73E-8	-0.44	1.54E-7	-0.34	7.85E-8	-0.38	5.04E-6	0.00	3.54E-6	0.00

### 3. TEST CASE APPLICATIONS

---

The comparison of the errors in Table 3.11 indicates that, when the equations are in primitive form, the accuracy is higher and the errors reach particularly low levels for the case in which the modified buffer layer is applied at both the inlet and the outlet. In this case, the convergence rate against the number of nodes per unit length matches the sixth-order roll off, which is the formal order of accuracy of the spatial differentiation of the interior scheme. This demonstrates that the error introduced with this time step by the fourth-order time-integration scheme is lower than that from the spatial discretisation and that, in the absence of the plates, introduced by the modified buffer is also lower than this threshold. The low-order filter applied near the boundaries has probably a negligible effect on the results in Table 3.11 as the flow in those points is overwritten with values almost exact at the end of each time step.

The method by Lockard & Morris [1997] for introducing the gust in the computational domain appears to generate higher values of  $l_2$  and  $l_\infty$  error norms and a lower error roll off than the inlet conditions. The order of convergence is negative when the Lockard & Morris [1997] gust generation is used with buffer layers, which is not a favourable combination.

Introducing the gust by a modified Giles boundary condition performs well as the error roll off is approximately second order. This rate is probably driven by the LOC near-boundary treatment of the implicit filter. The absolute level of the errors obtained appears to be adequate for the adoption in the complete problem including the flat plates cascade.

Figures 3.11, 3.12 and 3.13 show the distribution of the non-dimensional velocity component  $v$  and of the pressure oscillation  $p'$  with respect to the reference value  $\rho_\infty u_\infty^2$  at the end of simulations 3, 15 and 21 in Table 3.11. In detail (a) of Figure 3.11, the buffer zones at the inflow and outflow reduce the velocity amplitude to values very near

### 3. TEST CASE APPLICATIONS

---

Table 3.12: Grid-dependent explicit time-integration parameters for the two-dimensional linear cascade test case. The time step is non-dimensionalised by  $c/u_\infty$ .

Grid no.	Time step [nd]	Steps	Courant no.
1	0.01	50080	0.4816
2	0.005	100160	0.4816
3	0.0025	200320	0.4816
4	0.00125	400640	0.4816

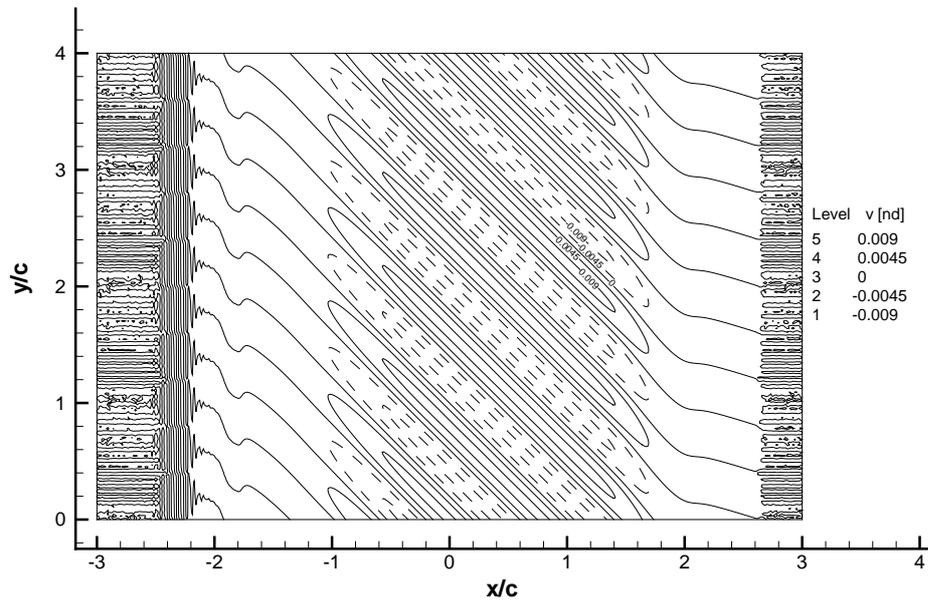
to zero, small oscillations with an amplitude about zero are still noticeable, and only in the central part of the field the gust has the correct value. Details (a) of Figures 3.12 and 3.13 show that, when the modified Giles and the modified buffer boundaries are used, the field is completely occupied by the gust. In details (b) of the figures, the non-dimensional amplitude of the pressure oscillation  $p'$  with respect to the reference value  $\rho_\infty u_\infty^2$  due to the introduction of the gust and to the influence of the boundaries, is shown for the three gust generation techniques. Here it is possible to appreciate the different levels of accuracy provided by the different gust generation methods. In Figure 3.13 (b), small oscillations related to the inter-block boundary approximations are visible.

#### 3.2.3.2 Results from the complete model

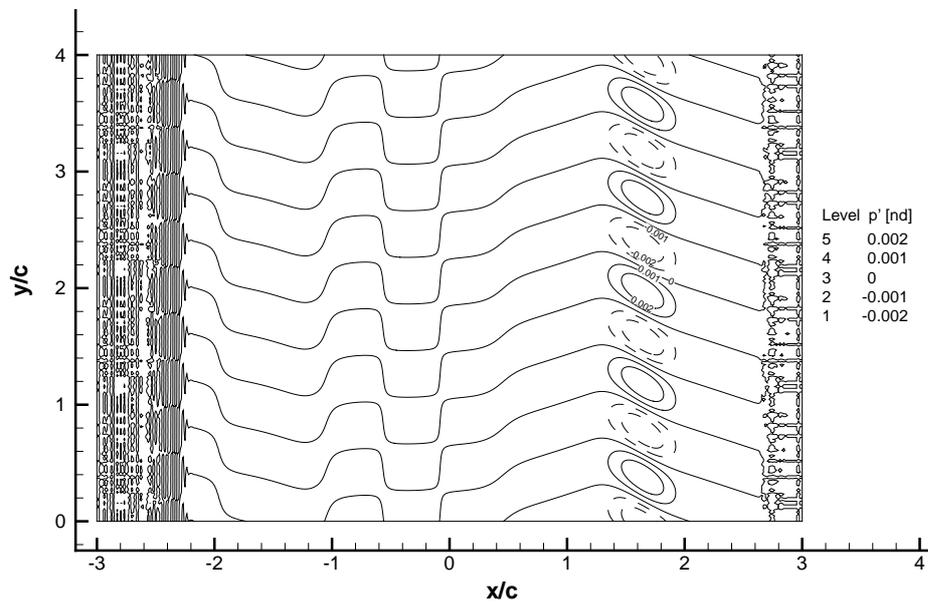
The results reported in this section are taken at a non-dimensional time  $t = 500.8$ .

Table 3.12 shows the main parameters of the time-integration scheme used as function of the computational grid. As in the simulations without the cascade, the accuracy limit is 0.4999 and the stability limit is 1.4217 (see Section 2.3 for details). These limits are met in all the runs reported in Tables 3.13 and 3.14.

### 3. TEST CASE APPLICATIONS



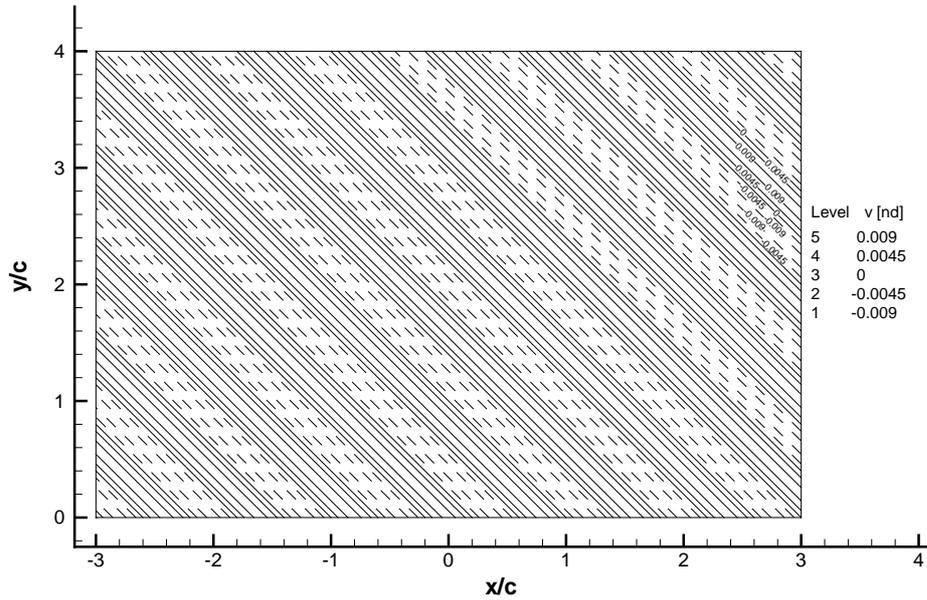
(a) Distribution of the non-dimensional  $v$  velocity.



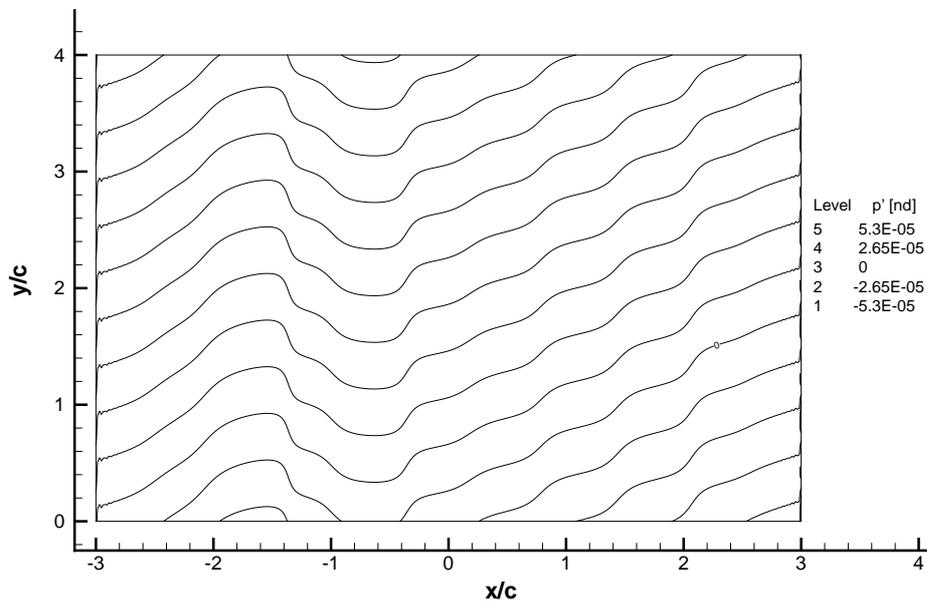
(b) Distribution of the non-dimensional amplitude of the pressure oscillation  $p'$  with respect to  $\rho_\infty u_\infty^2$ .

Figure 3.11: Simulation number 3 in Table 3.11 not including the linear cascade of flat plates. Gust introduced by the force field of Lockard & Morris [1997]. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS



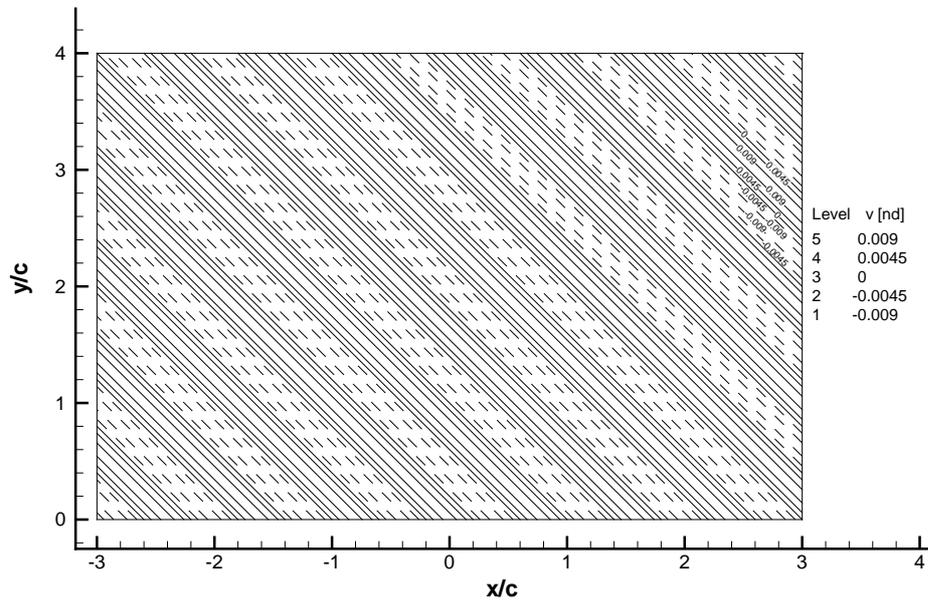
(a) Distribution of the non-dimensional  $v$  velocity.



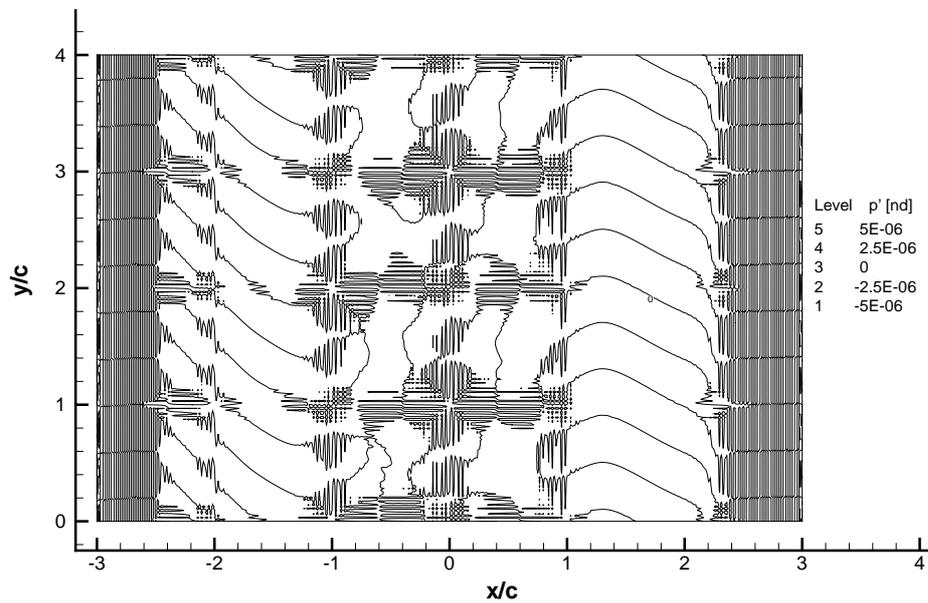
(b) Distribution of the non-dimensional amplitude of the pressure oscillation  $p'$  with respect to  $\rho_\infty u_\infty^2$ .

Figure 3.12: Simulation number 15 in Table 3.11 not including the linear cascade of flat plates. Gust introduced at the inflow boundary following Giles [1990]. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS



(a) Distribution of the non-dimensional  $v$  velocity.



(b) Distribution of the non-dimensional amplitude of the pressure oscillation  $p'$  with respect to  $\rho_\infty u_\infty^2$ .

Figure 3.13: Simulation number 21 in Table 3.11 not including the linear cascade of flat plates. Gust introduced by the modified buffer technique. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

Tables 3.13 and 3.14 list the results, as required by the problem definition, of the simulations that include the presence of the linear cascade of flat plates. The simulation was repeated by varying the length of the domain, the type of equation, the type of boundary closures, the method used for the introduction of the gust, and the relative position of the inflow plane and of the leading edge of the plates denoted by the non-dimensional value  $x_{in}$ . The non-dimensional errors of the pressure jump  $\Delta p$  between the faces of the reference plate, and of the radiated sound  $\overline{p^2}$  along the lines  $x/c = -2$  and  $x/c = +3$  are given from a representative set of these simulations.

The pressure difference  $\Delta p$  across the flat plates is evaluated at the non-dimensional time  $t = 500.8$ , which coincides with the end of a period and the error norms are given for both the real part of  $\Delta p$  and its imaginary part which is computed from the vectorial difference between the time-dependent oscillation amplitude recorded during the last period and the real part of  $\Delta p$ . The rate of convergence of the pressure difference  $\Delta p$  with increasing mesh refinement is between 0.7 and 1.5, which is lower than the sixth-order roll off that would be expected from the order of accuracy of the interior scheme. The radiated sound  $\overline{p^2}$  is evaluated between  $t = 500$  and  $t = 500.8$  by averaging  $p^2$  in time. The estimations of  $\overline{p^2}$  from grids with different levels of mesh refinement do not display a monotonic roll off. This suggests that, while some sources of error are reduced by incrementing the spatial refinement, other sources may increase.

Table 3.13 shows a summary of the results from the simulations where no modifications to the treatment of the plate edges and surfaces are introduced as detailed in Section 3.2.2. Table 3.14 shows results from the simulations for which the corrections are applied. In Table 3.14, field “Corr.” refers to the various levels of correction as defined in Section 3.2.2.

Table 3.13 shows that, by increasing the mesh density, the non-dimensional pres-

### 3. TEST CASE APPLICATIONS

---

sure difference  $\Delta p$  across the flat plates tends to converge towards the reference solution. The absolute values of the error are very similar among the various sets with the exception of simulations 9 to 11 in which the gust is generated by the force field of Lockard & Morris [1997], which is a rather coarse approximation and is the dominant source of error in the simulation as discussed in the context of Figure 3.11. Interestingly, the predicted pressure difference seems not to be significantly influenced by the length of the field as the sets 1 to 4, 16 to 19, and 20 to 23 show nearly the same results. The non-dimensional  $\overline{p^2}$  on the plane at  $x/c = -2$  in most cases gives results that tend towards the reference solution with an  $l_\infty$ -norm reducing with increasing mesh resolution. The best results are obtained by simulations 24 to 27 with a modified buffer condition at both the inflow and the outflow. On the plane at  $x/c = +3$  the situation is more involved and the convergence towards the reference values is obtained only in some cases.

Table 3.14 lists the results from the predictions in which corrections were introduced in the attempt to improve their accuracy. With Giles boundary conditions at both the inflow and the outflow (simulations 28 to 39) the pressure difference  $\Delta p$  error decreases with increasing level of correction and, with correction level 3 (simulations 36 to 39 in Table 3.14) the real part of the  $\Delta p$  error is lower than in the unmodified simulations (1 to 4 of Table 3.13). This is also the case for the simulations using the modified buffer at the external boundaries (simulations 24 to 27 and 56 to 59). Comparing the simulations from 36 to 47, it is evident that the length of the computational field has little influence on the pressure difference  $\Delta p$  when the Giles type of closure is enforced. In all the cases reported in Tables 3.13 and 3.14 the Kim and Lee inviscid wall, which uses conservative variables, performs better than the Tam and Dong wall closure for primitive variables, despite the primitive variables allowing the gen-

Table 3.13: Error analysis of the simulations including the linear cascade of flat plates. Predictions without special corrections for the treatment of the edges and the surfaces of the plates. Values are in non-dimensional form.

No.	Model name	Eq. form	Boundary conditions		Wall type	Gust	$x_G$	$x_{in}$ [nd]	$\Delta p$ real [nd]		$\Delta p$ imag. [nd]		$p^2$ [nd] at $x/c = -2$		$p^2$ [nd] at $x/c = +3$	
			$x$ low	$x$ high					$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$
1	7.1	cons.	Giles	Giles	Kim	Giles	na	-3	1.55E-2	3.27E-3	3.74E-2	7.38E-3	2.43E-7	1.87E-7	2.61E-7	2.01E-7
2	7.2	cons.	Giles	Giles	Kim	Giles	na	-3	1.12E-2	2.77E-3	2.87E-2	5.86E-3	3.05E-7	2.30E-7	7.70E-7	6.78E-7
3	7.3	cons.	Giles	Giles	Kim	Giles	na	-3	8.09E-3	2.20E-3	1.79E-2	4.28E-3	2.30E-7	1.64E-7	9.20E-7	7.89E-7
4	7.4	cons.	Giles	Giles	Kim	Giles	na	-3	6.40E-3	1.71E-3	6.28E-3	1.85E-3	1.33E-7	8.68E-8	8.95E-7	7.65E-7
5	7.1	prim.	Giles	Giles	Tam	Giles	na	-3	1.61E-2	3.40E-3	3.71E-2	7.36E-3	2.30E-7	1.74E-7	3.67E-7	3.19E-7
6	7.2	prim.	Giles	Giles	Tam	Giles	na	-3	1.18E-2	2.89E-3	2.88E-2	5.96E-3	3.72E-7	2.79E-7	9.22E-7	8.03E-7
7	7.3	prim.	Giles	Giles	Tam	Giles	na	-3	7.77E-3	2.33E-3	1.82E-2	4.40E-3	2.83E-7	2.03E-7	1.04E-6	8.91E-7
8	7.4	prim.	Giles	Giles	Tam	Giles	na	-3	7.27E-3	1.90E-3	7.07E-3	2.07E-3	1.60E-7	1.04E-7	9.91E-7	8.44E-7
9	9.1	cons.	buffer	Giles	Kim	Lockard	-1	-5	1.53E-2	3.43E-3	3.51E-2	7.08E-3	7.29E-7	5.82E-7	8.29E-7	7.56E-7
10	9.2	cons.	buffer	Giles	Kim	Lockard	-1	-5	1.11E-2	2.86E-3	2.78E-2	5.85E-3	5.04E-7	3.86E-7	9.29E-7	8.03E-7
11	9.3	cons.	buffer	Giles	Kim	Lockard	-1	-5	8.24E-3	2.25E-3	1.74E-2	4.29E-3	3.68E-7	2.75E-7	9.44E-7	8.06E-7
12	9.1	cons.	Giles	mod. buf.	Kim	Giles	na	-3	1.55E-2	3.26E-3	3.75E-2	7.40E-3	3.70E-7	3.03E-7	7.01E-8	4.87E-8
13	9.2	cons.	Giles	mod. buf.	Kim	Giles	na	-3	1.14E-2	2.78E-3	2.88E-2	5.88E-3	3.51E-7	2.62E-7	8.48E-7	7.09E-7
14	9.3	cons.	Giles	mod. buf.	Kim	Giles	na	-3	8.05E-3	2.22E-3	1.75E-2	4.26E-3	6.55E-8	3.88E-8	1.23E-6	1.08E-6
15	9.4	cons.	Giles	mod. buf.	Kim	Giles	na	-3	6.49E-3	1.78E-3	6.38E-3	1.77E-3	1.32E-7	8.23E-8	1.04E-6	9.32E-7
16	9.1	cons.	Giles	Giles	Kim	Giles	na	-4	1.52E-2	3.25E-3	3.74E-2	7.38E-3	2.42E-7	2.06E-7	1.86E-7	1.20E-7
17	9.2	cons.	Giles	Giles	Kim	Giles	na	-4	1.10E-2	2.76E-3	2.89E-2	5.87E-3	4.82E-7	3.78E-7	5.29E-7	4.15E-7
18	9.3	cons.	Giles	Giles	Kim	Giles	na	-4	8.47E-3	2.20E-3	1.82E-2	4.28E-3	4.52E-7	3.47E-7	5.94E-7	4.69E-7
19	9.4	cons.	Giles	Giles	Kim	Giles	na	-4	6.33E-3	1.74E-3	6.34E-3	1.84E-3	3.64E-7	2.72E-7	5.59E-7	4.39E-7
20	11.1	cons.	Giles	Giles	Kim	Giles	na	-5	1.56E-2	3.28E-3	3.74E-2	7.37E-3	2.63E-7	2.15E-7	2.82E-7	2.19E-7
21	11.2	cons.	Giles	Giles	Kim	Giles	na	-5	1.14E-2	2.77E-3	2.88E-2	5.88E-3	3.76E-7	2.94E-7	7.84E-7	6.27E-7
22	11.3	cons.	Giles	Giles	Kim	Giles	na	-5	7.95E-3	2.19E-3	1.81E-2	4.30E-3	3.14E-7	2.36E-7	8.86E-7	7.13E-7
23	11.4	cons.	Giles	Giles	Kim	Giles	na	-5	6.37E-3	1.70E-3	6.26E-3	1.88E-3	2.21E-7	1.59E-7	8.46E-7	6.82E-7
24	11.1	cons.	mod. buf.	mod. buf.	Kim	buffer	na	-5	1.54E-2	3.43E-3	3.52E-2	7.10E-3	8.87E-7	7.21E-7	7.26E-7	5.79E-7
25	11.2	cons.	mod. buf.	mod. buf.	Kim	buffer	na	-5	1.13E-2	2.86E-3	2.79E-2	5.87E-3	5.42E-7	4.18E-7	1.01E-6	8.37E-7
26	11.3	cons.	mod. buf.	mod. buf.	Kim	buffer	na	-5	8.19E-3	2.26E-3	1.70E-2	4.27E-3	1.64E-7	1.00E-7	1.25E-6	1.09E-6
27	11.4	cons.	mod. buf.	mod. buf.	Kim	buffer	na	-5	6.52E-3	1.84E-3	6.49E-3	1.80E-3	9.44E-8	6.26E-8	1.07E-6	9.64E-7

### 3. TEST CASE APPLICATIONS

---

eration of a gust at the inflow that is closer to the analytical solution as reported in Section 3.2.3.1. In general, the Giles boundary conditions and the modified buffer give satisfactory results with similar error magnitudes, as shown by comparing simulations 44-47 to 56-59. The Giles boundary closures are preferred because they are computationally more efficient as they do not require a layer of boundary nodes. The error norms of  $\overline{p^2}$  at the inflow show the expected trend with increasing mesh refinement but with absolute errors that only in certain cases compare favourably against the simulations without corrections. For the mesh with highest spatial resolution, the best results are obtained with the modified buffers. At the outflow, as for the case without corrections, a clear trend in  $\overline{p^2}$  error norms cannot be found. The length of the field seems to have a noticeable influence on  $\overline{p^2}$  at the inflow and at the outflow, but the results are difficult to interpret.

Figure 3.14 shows a representative set of results from the simulations without additional modifications. It refers to simulations 1 to 4 of Table 3.13. This set is obtained with the Euler equations in conservative form, the Giles subsonic non-reflecting boundaries at inflow and outflow, the Kim and Lee inviscid wall for the plate surfaces, and a computational domain of non-dimensional length 7 in the  $x$  direction. Figure 3.14 shows both the real and the imaginary part of the non-dimensional pressure difference  $\Delta p$  across the blade at  $y = 0$ . The symbols are predictions at different levels of grid refinement and the continuous line is the reference solution by Hall [1997a]. Spatial oscillations at the plate leading and trailing edges are shown that tend to increase in amplitude with increasing number of nodes. On the left-hand side of the graph, the real part of the predicted  $\Delta p$  converges towards the reference solution. On the right-hand side, a small offset can be noticed and the pressure difference magnitude is slightly over-estimated. The imaginary part seems to display the correct trend with an offset

Table 3.14: Error analysis of the simulations including the linear cascade of flat plates. Predictions with corrections for the treatment of the edges and the surface of the plates. The levels of correction as defined in Section 3.2.2 are shown in field “Corr.”. Values are in non-dimensional form.

No.	Model name	Eq. form	Boundary conditions		Wall type	Gust	Corr.	$x_{in}$ [nd]	$\Delta p$ real [nd]		$\Delta p$ imag. [nd]		$\overline{p^2}$ [nd] at $x/c = -2$		$\overline{p^2}$ [nd] at $x/c = +3$	
			$x$ low	$x$ high					$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$
28	7.1	cons.	Giles	Giles	Kim	Giles	1	-3	8.53E-3	2.39E-3	3.17E-2	7.30E-3	2.63E-7	1.70E-7	7.85E-7	6.81E-7
29	7.2	cons.	Giles	Giles	Kim	Giles	1	-3	4.74E-3	1.82E-3	1.96E-2	5.63E-3	4.33E-7	3.10E-7	9.03E-7	7.61E-7
30	7.3	cons.	Giles	Giles	Kim	Giles	1	-3	8.23E-3	2.13E-3	2.13E-2	5.61E-3	2.51E-7	1.73E-7	8.35E-7	7.05E-7
31	7.4	cons.	Giles	Giles	Kim	Giles	1	-3	1.08E-2	2.58E-3	2.31E-2	6.05E-3	1.08E-7	6.51E-8	7.78E-7	6.60E-7
32	7.1	cons.	Giles	Giles	Kim	Giles	2	-3	1.01E-2	2.55E-3	2.92E-2	6.49E-3	4.92E-7	3.55E-7	8.30E-7	7.11E-7
33	7.2	cons.	Giles	Giles	Kim	Giles	2	-3	4.05E-3	1.40E-3	1.45E-2	3.94E-3	4.78E-7	3.51E-7	7.36E-7	6.19E-7
34	7.3	cons.	Giles	Giles	Kim	Giles	2	-3	4.17E-3	1.17E-3	1.15E-2	3.29E-3	2.66E-7	1.93E-7	5.54E-7	4.74E-7
35	7.4	cons.	Giles	Giles	Kim	Giles	2	-3	8.16E-3	1.62E-3	1.93E-2	3.83E-3	1.26E-7	8.67E-8	4.45E-7	3.88E-7
36	7.1	cons.	Giles	Giles	Kim	Giles	3	-3	1.44E-2	3.32E-3	3.23E-2	7.02E-3	5.46E-7	3.99E-7	8.40E-7	7.20E-7
37	7.2	cons.	Giles	Giles	Kim	Giles	3	-3	8.52E-3	1.95E-3	1.98E-2	4.34E-3	4.62E-7	3.41E-7	6.65E-7	5.61E-7
38	7.3	cons.	Giles	Giles	Kim	Giles	3	-3	3.79E-3	1.13E-3	1.07E-2	3.01E-3	2.37E-7	1.72E-7	4.75E-7	4.10E-7
39	7.4	cons.	Giles	Giles	Kim	Giles	3	-3	5.29E-3	1.15E-3	1.37E-2	2.93E-3	1.09E-7	7.25E-8	3.74E-7	3.31E-7
40	9.1	cons.	Giles	Giles	Kim	Giles	3	-4	1.42E-2	3.25E-3	3.23E-2	7.01E-3	5.66E-7	4.05E-7	8.81E-7	6.78E-7
41	9.2	cons.	Giles	Giles	Kim	Giles	3	-4	8.19E-3	1.85E-3	2.00E-2	4.39E-3	6.21E-7	4.74E-7	4.18E-7	3.07E-7
42	9.3	cons.	Giles	Giles	Kim	Giles	3	-4	3.35E-3	1.03E-3	1.10E-2	3.08E-3	4.15E-7	3.24E-7	1.81E-7	1.27E-7
43	9.4	cons.	Giles	Giles	Kim	Giles	3	-4	5.22E-3	1.09E-3	1.39E-2	2.97E-3	2.82E-7	2.22E-7	8.33E-8	5.08E-8
44	11.1	cons.	Giles	Giles	Kim	Giles	3	-5	1.48E-2	3.40E-3	3.24E-2	7.00E-3	5.99E-7	4.40E-7	9.56E-7	7.35E-7
45	11.2	cons.	Giles	Giles	Kim	Giles	3	-5	8.72E-3	1.98E-3	1.99E-2	4.36E-3	5.40E-7	4.08E-7	6.64E-7	5.10E-7
46	11.3	cons.	Giles	Giles	Kim	Giles	3	-5	3.94E-3	1.15E-3	1.09E-2	3.04E-3	3.18E-7	2.42E-7	4.48E-7	3.46E-7
47	11.4	cons.	Giles	Giles	Kim	Giles	3	-5	5.35E-3	1.16E-3	1.37E-2	2.95E-3	1.86E-7	1.42E-7	3.38E-7	2.62E-7
48	9.1	cons.	Giles	mod. buf.	Kim	Giles	3	-3	1.45E-2	3.33E-3	3.25E-2	7.05E-3	6.69E-7	5.05E-7	7.53E-7	5.71E-7
49	9.2	cons.	Giles	mod. buf.	Kim	Giles	3	-3	8.82E-3	2.01E-3	1.99E-2	4.35E-3	5.26E-7	3.91E-7	7.25E-7	5.77E-7
50	9.3	cons.	Giles	mod. buf.	Kim	Giles	3	-3	3.92E-3	1.18E-3	1.03E-2	2.94E-3	6.98E-8	4.15E-8	7.59E-7	6.60E-7
51	9.4	cons.	Giles	mod. buf.	Kim	Giles	3	-3	5.02E-3	1.11E-3	1.36E-2	2.87E-3	1.30E-7	1.06E-7	5.09E-7	4.87E-7
52	7.1	prim.	Giles	Giles	Tam	Giles	3	-3	5.17E-3	2.66E-3	2.14E-2	6.26E-3	2.34E-7	1.35E-7	1.71E-6	1.44E-6
53	7.2	prim.	Giles	Giles	Tam	Giles	3	-3	1.17E-2	3.62E-3	1.40E-2	5.06E-3	7.09E-7	5.04E-7	1.44E-6	1.18E-6
54	7.3	prim.	Giles	Giles	Tam	Giles	3	-3	1.29E-2	4.01E-3	1.74E-2	5.17E-3	8.68E-7	6.64E-7	1.06E-6	8.50E-7
55	7.4	prim.	Giles	Giles	Tam	Giles	3	-3	1.53E-2	3.12E-3	2.20E-2	5.04E-3	6.87E-7	5.36E-7	7.41E-7	6.00E-7
56	11.1	cons.	mod. buf.	mod. buf.	Kim	buffer	3	-5	1.46E-2	3.63E-3	2.97E-2	6.46E-3	1.28E-6	9.69E-7	1.57E-6	1.22E-6
57	11.2	cons.	mod. buf.	mod. buf.	Kim	buffer	3	-5	8.72E-3	2.08E-3	1.87E-2	4.13E-3	7.40E-7	5.60E-7	8.85E-7	7.01E-7
58	11.3	cons.	mod. buf.	mod. buf.	Kim	buffer	3	-5	3.77E-3	1.19E-3	9.68E-3	2.85E-3	1.96E-7	1.30E-7	7.86E-7	6.76E-7
59	11.4	cons.	mod. buf.	mod. buf.	Kim	buffer	3	-5	4.91E-3	1.11E-3	1.35E-2	2.84E-3	7.98E-8	4.80E-8	5.43E-7	5.19E-7

### 3. TEST CASE APPLICATIONS

---

particularly pronounced between  $x/c = +0.05$  and  $x/c = +0.4$ .

In Figures 3.15 and 3.16 the non-dimensional pressure  $\overline{p^2}$  is shown for the same set of results as Figure 3.14, which are simulations 1 to 4 of Table 3.13. Upstream of the cascade, Figure 3.15 shows that, as the grid is refined, the predicted non-dimensional pressure amplitude, shown by symbols, tends to converge towards the reference solution by Hall [1997a], denoted by the continuous line. Downstream of the cascade, at  $x/c = +3$ , the values tend to converge towards a limit showing the correct sound directivity, displayed by the correct location of the  $\overline{p^2}$  maximum and minimum along the  $y$  direction, but that is over-estimated with respect to the reference solution. This is common to most of the other sets of solutions.

Figure 3.17 compares the results of the radiated noise at the outflow plane from simulations differing only for the extension of the field in the  $x$  direction. Increasing the distance between the cascade trailing edge and the computational domain outflow boundary affects the amplitude of the mean squared pressure fluctuation at  $x/c = +3$ . There is a rather small effect on the spatial phase of the mean squared pressure distribution, confirming that the directivity of the pressure perturbation is captured by the simulation, but for the  $\overline{p^2}$  mean level. The uniformly lower  $\overline{p^2}$  predicted by the computation with streamwise length  $L = 9$  may result from wave patterns generated by spurious wave reflections at the outflow boundary.

Figures 3.18, 3.19 and 3.20 show respectively the non-dimensional perturbation fields of pressure  $p'$  and velocity  $u'$  and  $v'$  at non-dimensional time  $t = 500.8$ . The perturbation is computed about the uniform reference flow  $\mathbf{Q}_\infty$  of Equation (3.14). The non-dimensional velocity fluctuation in the  $y$  direction  $v'$  is coincident with the value of the non-dimensional velocity component  $v$ . The distributions of pressure and velocity perturbations give patterns that are consistent with the predictions shown in Tam

### 3. TEST CASE APPLICATIONS

---

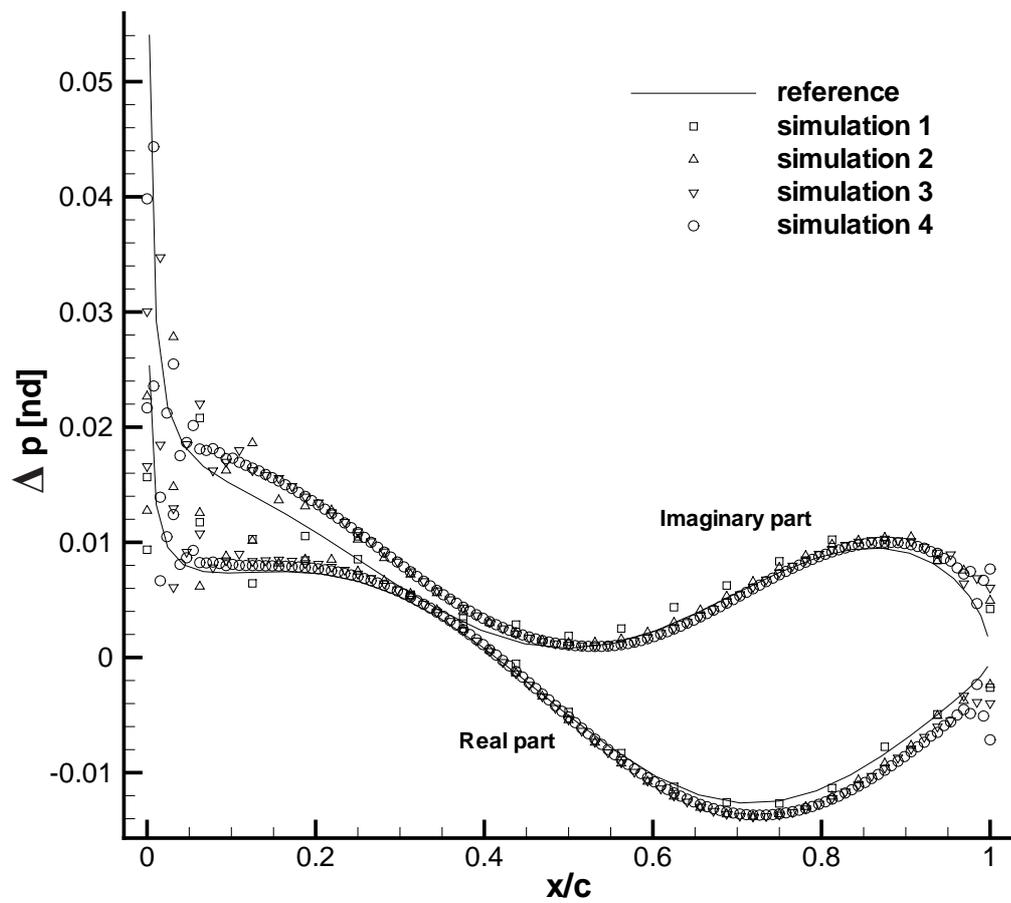


Figure 3.14: Simulations of the linear cascade of flat plates. Non-dimensional pressure difference across the  $y = 0$  blade at the non-dimensional time  $t = 500.8$ . Results from simulations 1 to 4 in Table 3.13 with increasingly refined grids which do not include the corrections of Section 3.2.2.

### 3. TEST CASE APPLICATIONS

---

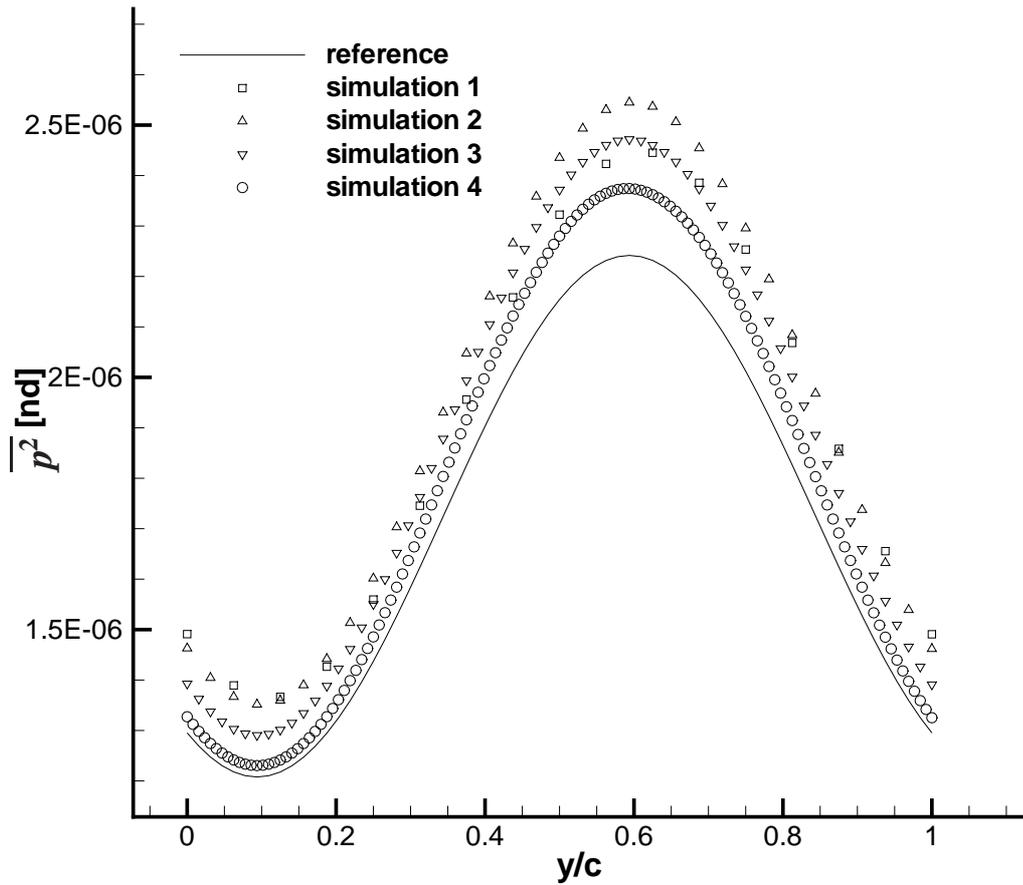


Figure 3.15: Linear cascade of flat plates with the inflow gust: non-dimensional pressure amplitude  $\overline{p^2}$  evaluated between non-dimensional time  $t = 500$  and  $t = 500.8$  upstream of the cascade at  $x/c = -2$ . Symbols represent simulations 1 to 4 in Table 3.13 with an increasingly refined mesh that do not include the corrections of Section 3.2.2. Only the range of  $y/c$  between 0 and 1 is included as the values from 2 to 4 are identical due to the cascade pitchwise periodicity.

### 3. TEST CASE APPLICATIONS

---

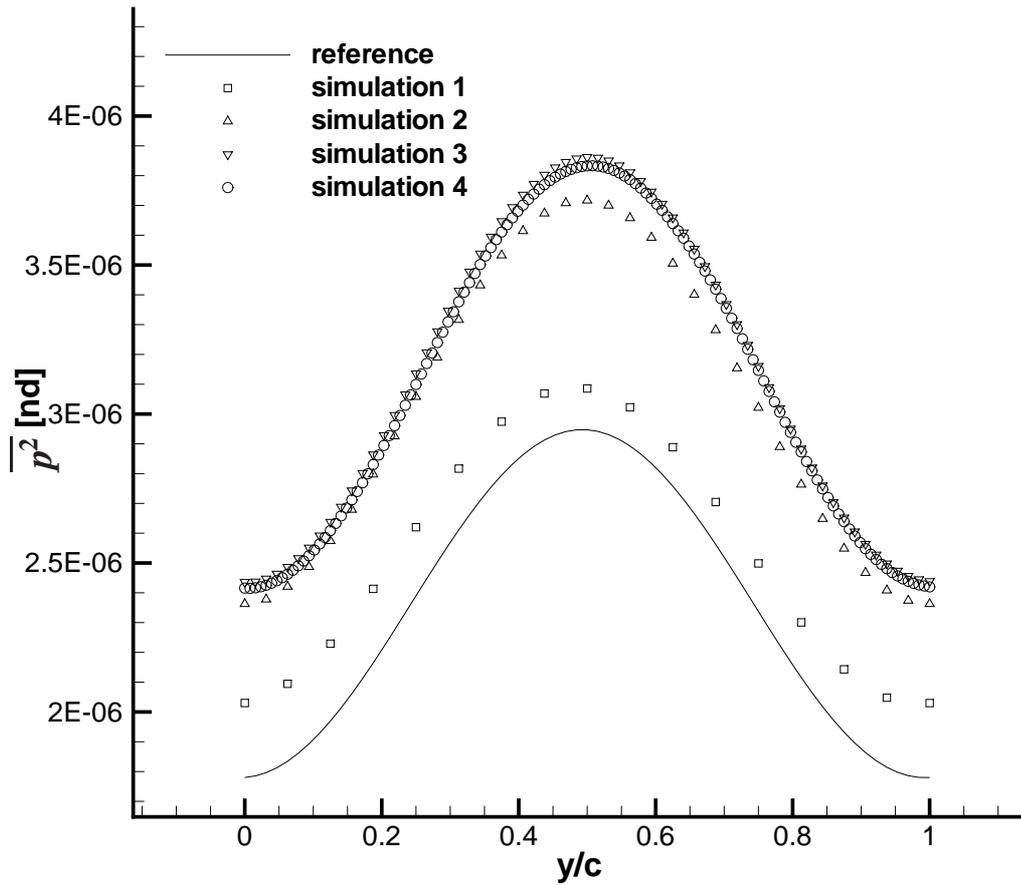


Figure 3.16: Linear cascade of flat plates with the inflow gust: non-dimensional pressure amplitude  $\overline{p^2}$  evaluated between non-dimensional time  $t = 500$  and  $t = 500.8$  downstream of the cascade at  $x/c = +3$ . Symbols represent simulations 1 to 4 in Table 3.13 with an increasingly refined mesh that do not include the corrections of Section 3.2.2. Only the range of  $y/c$  between 0 and 1 is included as the values from 2 to 4 are identical due to the cascade pitchwise periodicity.

### 3. TEST CASE APPLICATIONS

---

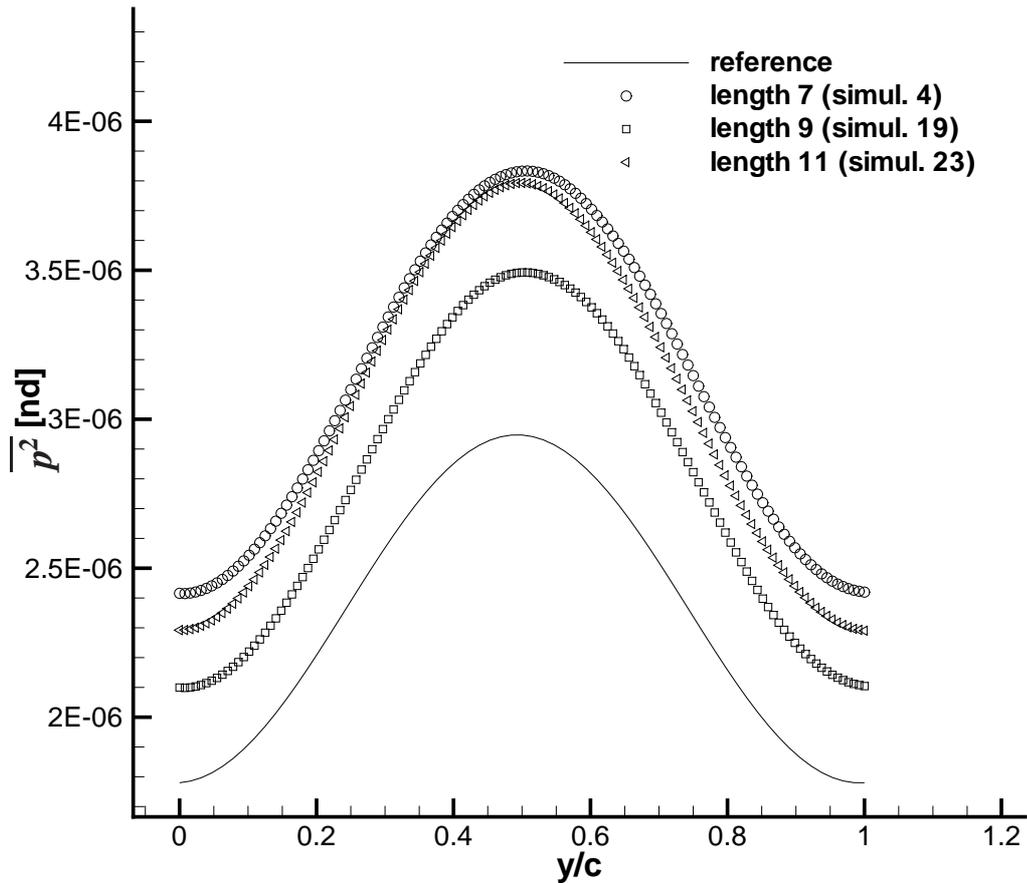


Figure 3.17: Linear cascade of flat plates with the inflow gust: non-dimensional pressure amplitude  $\overline{p^2}$  evaluated between non-dimensional time  $t = 500$  and  $t = 500.8$  downstream of the cascade at  $x/c = +3$ . Symbols represent simulations 4, 19 and 23 in Table 3.13 with increasing domain length  $L$ . The simulations do not include the corrections of Section 3.2.2. Only the range of  $y/c$  between 0 and 1 is included as the values from 2 to 4 are identical due to the cascade pitchwise periodicity.

### 3. TEST CASE APPLICATIONS

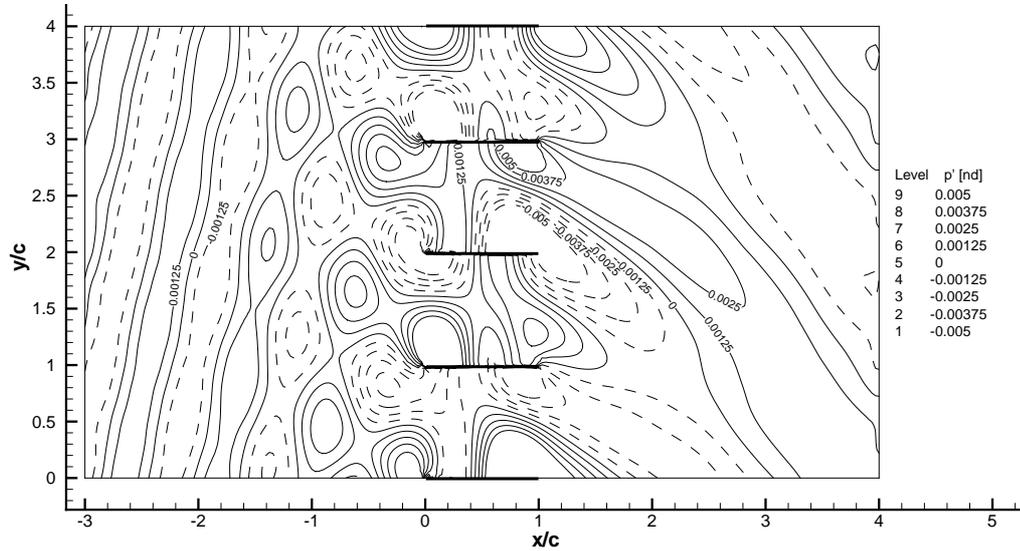


Figure 3.18: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional pressure perturbation  $p' = p - \rho_\infty u_\infty^2$  (where  $p$  is the non-dimensional pressure computed as required by the problem definition) over the entire computational domain at the end of simulation 2, Table 3.13. Negative contours are shown by dashed lines.

& Hardin [1997]. The velocity components in Figures 3.19 and 3.20 do not show any change in spatially periodic patterns approaching the computational domain boundaries where the Giles subsonic conditions are enforced, which indicates that the velocity perturbation is unaffected by the domain boundaries that are substantially non-reflecting for this components of the flow state. The pressure perturbation is slightly more irregular at the outflow boundary. At the inflow boundary small disturbances, that tend to disappear in the more refined grid simulations, are noticeable.

Figure 3.21 shows the distribution over the whole computational field of the non-dimensional mean pressure amplitude  $\overline{p^2}$  computed during the last period of simulation 2 in Table 3.13. The directivity of the radiating pressure is shown by the lobes of the  $\overline{p^2}$  contours in Figure 3.21 (a). Figures 3.21 (b) and (c) show the decay of  $\overline{p^2}$  along the

### 3. TEST CASE APPLICATIONS

---

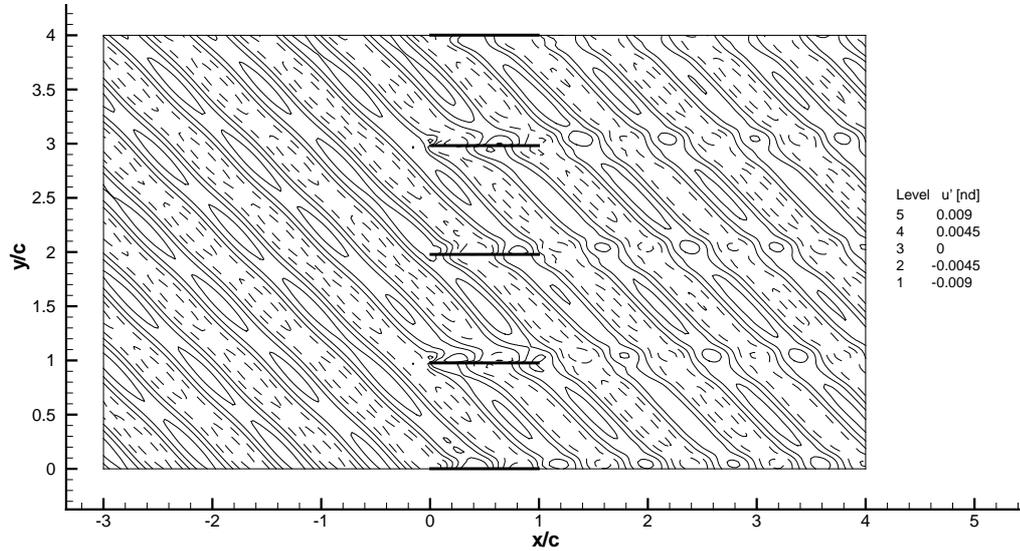


Figure 3.19: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional perturbation of the streamwise velocity  $u' = u - u_\infty$  over the entire computational domain at the end of simulation 2, Table 3.13. Negative contours are shown by dashed lines.

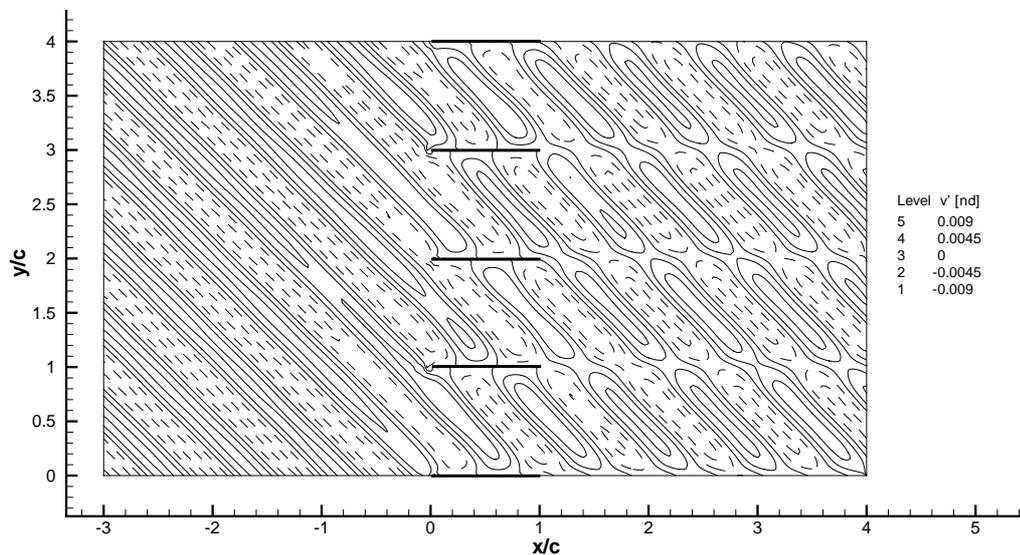


Figure 3.20: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional perturbation of the flow-normal velocity  $v' = v - v_\infty$  over the entire computational domain at the end of simulation 2, Table 3.13. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

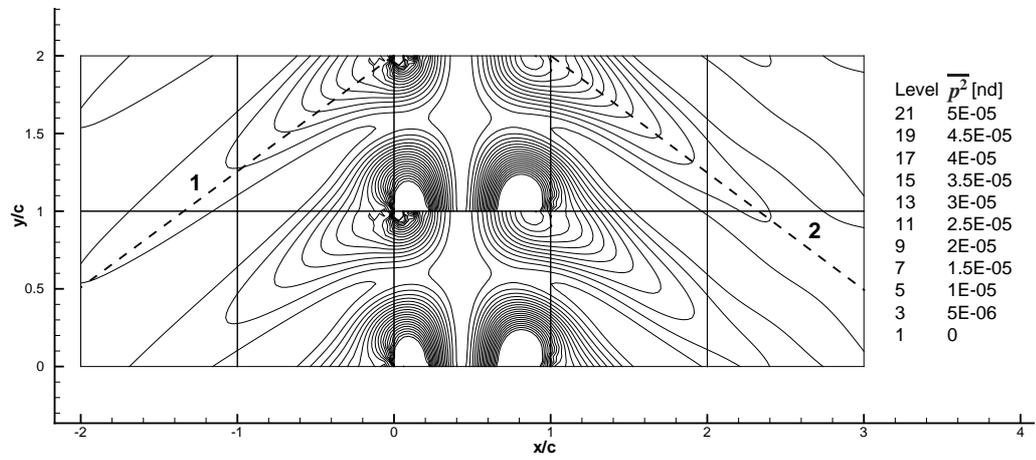
dashed lines 1 and 2 in Figure 3.21 (a). Two directivity peaks are shown at  $\theta = 220^\circ$  from the cascade leading edge and at  $\theta = -40^\circ$  from the cascade trailing edge, where  $\theta$  is measured anti-clockwise positive from the  $x$  axis. The decay is monotonic and, in logarithmic scale, non-linear. The non-linearity can be explained as in both the upstream and downstream direction the cascade response to the incoming gust is composed of a number of modes that are either cut-on (propagating) or cut-off (decaying). For the combination of the geometry and gust prescribed by the problem definition of the second test case, the dominating mode, the only clearly visible in Figure 3.18, is decaying, but other lower amplitude propagating modes are superimposed to it in the resulting pressure field. The sum of these contributions gives to the pressure profile along dashed lines 1 and 2 the shape of a blend of two straight lines. The pressure amplitudes related to the various modes of the cascade response could be obtained by performing a double (both in time and space) Fourier transform of the predicted pressure as done for the fourth test case of this study in Section 3.4. The mode decomposition for this test case can be found in Ragab & Salem-Said [2007].

The decay in the downstream direction, shown in Figure 3.21 (c), is characterised by changes in curvature that are absent from the pressure amplitude distribution in the upstream direction shown in Figure 3.21 (b). This is probably due to the fact that the outflow Giles subsonic boundary partially reflects the outgoing waves. The over-estimation of the  $\overline{p^2}$  amplitude noted in Figures 3.16 and 3.17, and directly from the data in Table 3.13, could be related to this phenomenon.

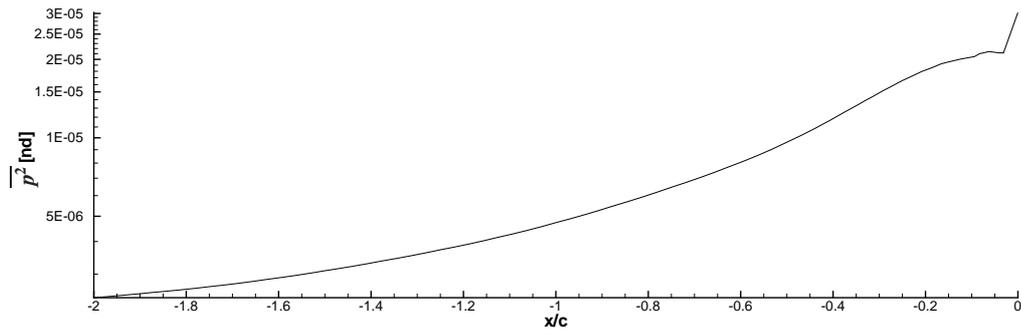
Figures from 3.22 to 3.24 refer to simulations 36 to 39 of Table 3.14 with correction level 3. They can be directly compared to Figures from 3.14 to 3.16.

Figures from 3.25 to 3.27 illustrate the results from simulation 37 in Table 3.14. These values can be directly compared to those from simulation 2 shown in Figures

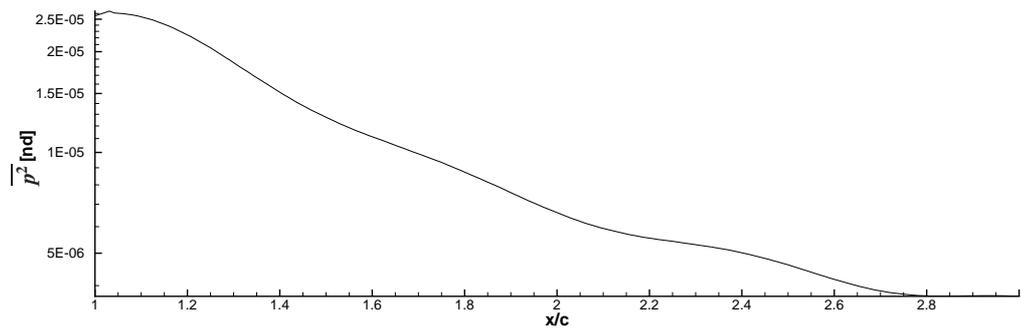
### 3. TEST CASE APPLICATIONS



(a) Contours of  $\overline{p^2}$  over two vanes showing the unsteady pressure directivity.



(b) Values along dashed line 1 in Figure 3.21 (a).



(c) Values along dashed line 2 in Figure 3.21 (a).

Figure 3.21: Linear cascade of flat plates with inflow gust. Non-dimensional unsteady pressure amplitude  $\overline{p^2}$  computed over the last period of simulation 2, Table 3.13.

### 3. TEST CASE APPLICATIONS

---

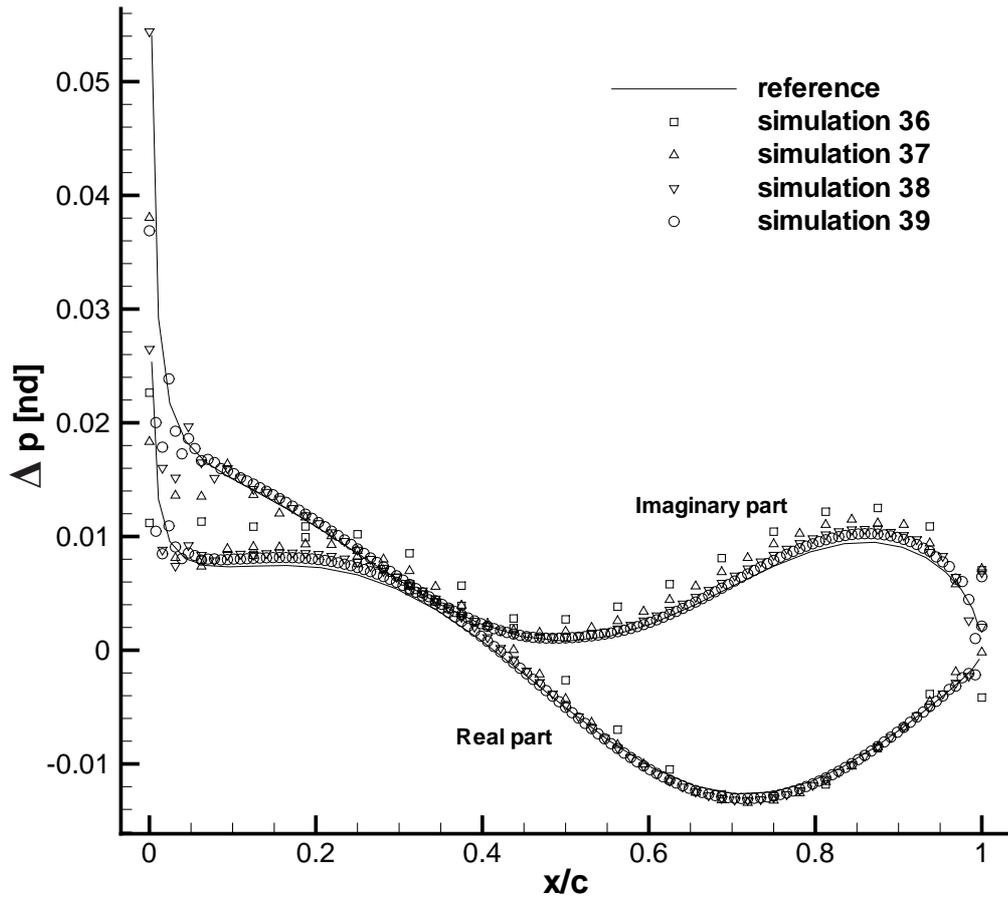


Figure 3.22: Linear cascade of flat plates with inflow gust. Non-dimensional pressure difference  $\Delta p$  across the  $y = 0$  flat plate at non-dimensional time  $t = 500.8$ . The results from four simulations with correction level 3 (36 to 39 in Table 3.14) and increasingly refined grids are shown. The same axis scales are used as in Figure 3.14 to facilitate the comparison among the results.

### 3. TEST CASE APPLICATIONS

---

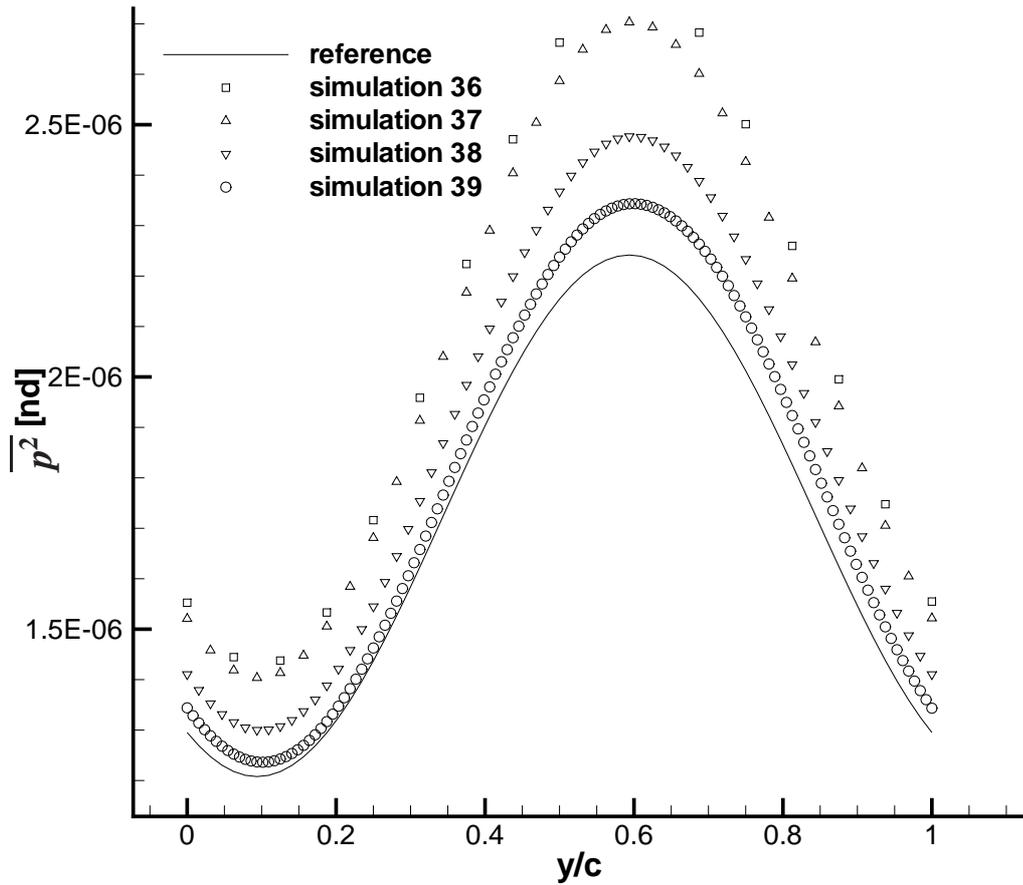


Figure 3.23: Linear cascade of flat plates with inflow gust. Flow-normal distribution of fluctuating pressure amplitude at  $x/c = -2$  upstream of the flat plate leading edge, averaged over the non-dimensional time  $500 < t \leq 500.8$ . Predictions with scheme correction level 3 at increasing levels of mesh refinement (symbols). Reference solution (continuous line). Simulations 36 to 39 in Table 3.14. The ordinate axis range is  $0 \leq y/c \leq 1$  as the pressure distribution is pitchwise periodic.

### 3. TEST CASE APPLICATIONS

---

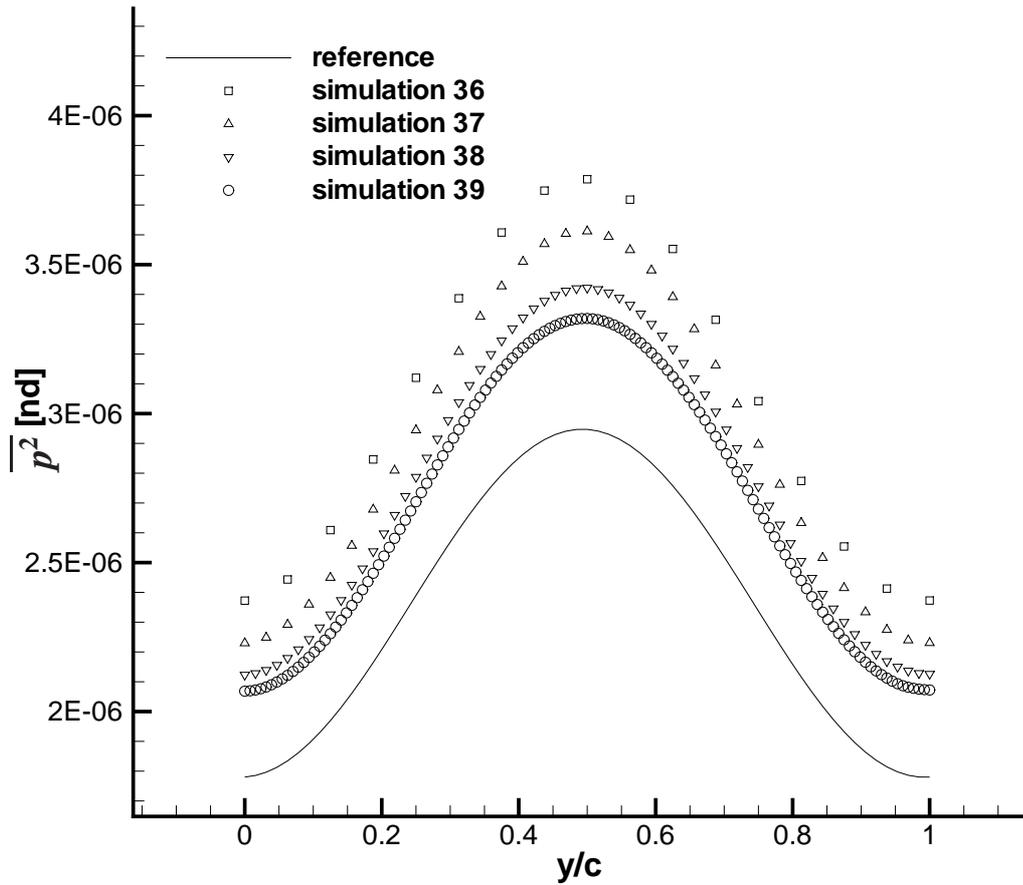


Figure 3.24: Linear cascade of flat plates with inflow gust. Flow-normal distribution of fluctuating pressure amplitude at  $x/c = +3$  downstream of the flat plate leading edge, averaged over the non-dimensional time  $500 < t \leq 500.8$ . Predictions with scheme correction level 3 at increasing levels of mesh refinement (symbols). Reference solution (continuous line). Simulations 36 to 39 in Table 3.14. The ordinate axis range is  $0 \leq y/c \leq 1$  as the pressure distribution is pitchwise periodic.

### 3. TEST CASE APPLICATIONS

---

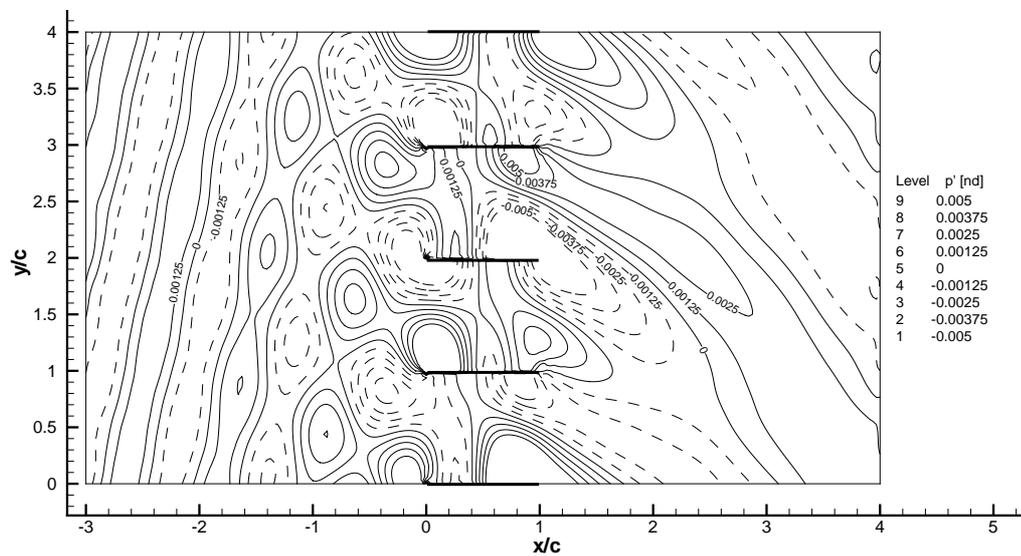


Figure 3.25: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional pressure perturbation  $p' = p - \rho_\infty u_\infty^2$  (where  $p$  is the non-dimensional pressure computed as required by the problem definition) over the entire computational domain at the end of simulation 37, Table 3.14. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

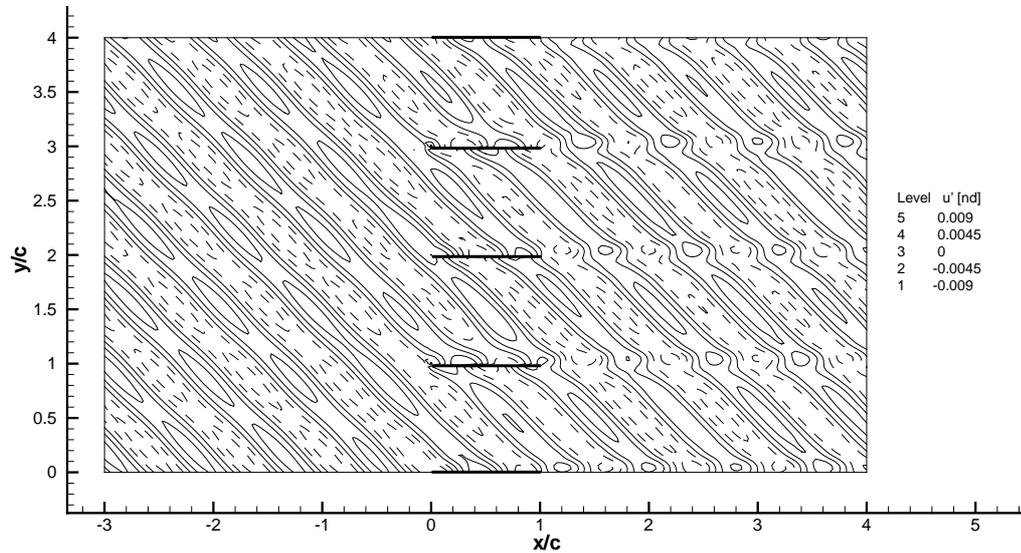


Figure 3.26: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional perturbation of the streamwise velocity  $u' = u - u_\infty$  over the entire computational domain at the end of simulation 37, Table 3.14. Negative contours are shown by dashed lines.

from 3.18 to 3.20. Simulations 2 and 37 solve the same formulation of the Euler equations, using the same type of boundary conditions, gust introduction method, and computational domain field length. This allows to graphically evaluate the increase in the accuracy due to the introduction of the corrections applied at the blade edges and surfaces discussed in Section 3.2.2.

Figure 3.22 compares the flat plate surface pressure difference from predictions using different levels of mesh refinement (symbols) with the reference solution by Hall [1997a] (continuous line). Figure 3.22 shows that the real part of the non-dimensional pressure difference  $\Delta p$  across the  $y = 0$  blade from the simulations with higher node density (simulations 38 and 39) is nearly the same and overlaps the reference solution over the whole plate. The spatial oscillations in  $\Delta p$  observed in Figure 3.14, especially near the leading edge, where the pressure gradients are steeper, are shown to be sub-

### 3. TEST CASE APPLICATIONS

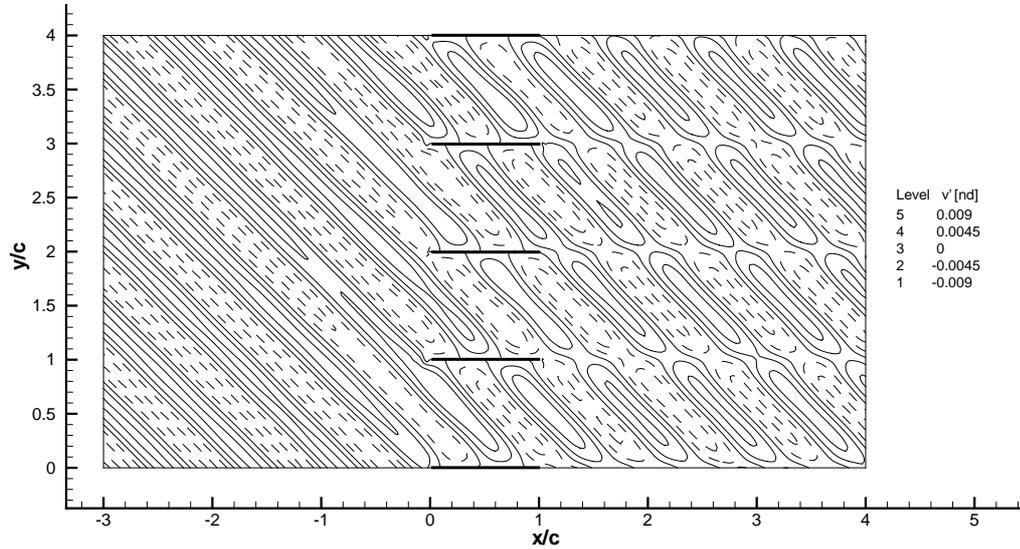


Figure 3.27: Linear cascade of flat plates with inflow gust. Distribution of the non-dimensional perturbation of the flow-normal velocity  $v' = v - v_\infty$  over the entire computational domain at the end of simulation 37, Table 3.14. Negative contours are shown by dashed lines.

stantially attenuated in Figure 3.22. The imaginary part of  $\Delta p$  in Figure 3.22 is also nearly coincident with the reference solution and the offset along the left-hand side of the plate noted in Figure 3.14 is shown to have been substantially reduced by the level 3 corrections applied by the numerical method at the flat plate edges and surfaces.

Figure 3.23 shows the flow-normal distribution of the pressure fluctuation amplitude  $\overline{p^2}$  at  $x/c = -2$ , upstream of the flat plate leading edge. These results are predictions obtained with the level 3 corrections of Section 3.2.2. In Figure 3.23, the non-dimensional pressure fluctuation amplitude  $\overline{p^2}$  from the coarser mesh simulations ( $\square$ ,  $\Delta$ ) shows a lower agreement with the reference solution (continuous line) when compared to Figure 3.15. However, the predictions from the more refined meshes ( $\nabla$ ,  $\circ$ ) show an improved agreement with the reference solution compared to Figure 3.15.

Figure 3.24 shows the flow-normal distribution of unsteady pressure predicted at

### 3. TEST CASE APPLICATIONS

---

$x/c = +3$ , downstream of the plate trailing edge, using the same level 3 correction as for Figure 3.23. At  $x/c = +3$ , the prediction shown by symbols displays a vertical offset with respect to the reference solution (continuous line) similarly to Figure 3.16. However, this offset is lower than in the predictions without the level 3 correction. The level 3 predictions also display a monotonically decreasing difference among them as the computational mesh is refined, which is a strong indication of mesh convergence in the computation.

Figures 3.25, 3.26 and 3.27 show the distributions of fluctuating pressure, stream-wise, and flow-normal velocity components across the full computational domain at non-dimensional time  $t = 500.8$ . These predictions are obtained using the level 3 correction in the scheme. The displayed predictions are substantially equivalent to those without the corrections.

Table 3.15 summarises the computational resources used to run each simulation. Due to the scheduler of the high-performance computer that was used, the runs have not been made under identical conditions, such as by reserving entire blades of processors to maximise the locality of the memory and to minimise the latencies related to the data exchange. This resulted in a significant data dispersion in the recorded cost of the computation that does not allow to accurately evaluate the additional computational cost from the corrections implemented to improve the quality of the results.

The results show that both the modified buffer condition and the Giles non-reflecting subsonic inflow and outflow, in association with the inviscid wall boundary by Kim and Lee, are suitable for the solution of the second test case. The Giles boundary conditions have to be preferred over the modified buffer because they do not require additional grid nodes to host the buffer layer and are, therefore, more efficient. The

### 3. TEST CASE APPLICATIONS

Table 3.15: Computational cost and duration of the linear cascade simulations of flat plates with inflow gust.

No.	Walltime [hh:mm:ss]	CPU time [hh:mm:ss]	Total mem. [MB]	No.	Walltime [hh:mm:ss]	CPU time [hh:mm:ss]	Total mem. [MB]
1	00:05:01	01:36:35	692	32	02:00:16	01:48:25	694
2	00:12:55	05:20:20	735	33	00:11:00	05:05:20	727
3	03:16:42	88:39:38	980	34	03:32:37	95:48:21	1001
4	33:27:53	921:30:45	1841	35	41:17:34	1111:10:40	1894
5	00:05:13	01:35:14	686	36	00:06:14	01:58:55	694
6	00:10:19	04:45:35	739	37	00:14:25	06:37:14	732
7	01:10:23	32:48:43	866	38	04:12:04	113:38:40	1057
8	35:51:50	980:45:13	1782	39	36:06:30	990:29:41	1678
9	00:04:23	01:41:56	873	40	00:05:30	02:20:09	838
10	00:19:11	11:09:54	943	41	00:12:48	07:31:52	919
11	03:24:05	118:34:20	1281	42	04:04:16	139:53:43	1293
	na	na	na	43	38:31:07	1359:45:07	2172
12	00:03:16	01:48:53	879	44	00:03:28	02:26:36	1047
13	00:13:31	07:57:59	941	45	00:15:34	11:11:34	1162
14	03:41:05	128:56:12	1344	46	04:11:26	174:24:32	1574
15	29:58:06	1078:13:50	2041	47	47:48:21	1987:32:49	2953
16	00:03:30	01:59:01	867	48	00:03:29	01:55:31	873
17	00:13:03	06:54:58	937	49	00:13:59	08:15:56	946
18	04:07:02	139:50:23	1135	50	02:55:07	104:55:43	1122
19	51:39:29	1692:20:37	2334	51	29:45:45	1070:50:42	2022
20	00:03:15	02:17:08	1060	52	00:03:02	01:20:31	654
21	00:10:58	07:59:08	1138	53	00:12:08	05:34:29	729
22	03:20:09	142:55:11	1487	54	01:43:28	48:07:55	879
23	37:05:53	1583:54:32	2986	55	35:17:53	947:43:28	1817
24	00:12:09	07:38:56	1086	56	00:11:38	07:23:10	1079
25	00:21:04	14:22:21	1149	57	00:21:20	14:39:09	1161
26	02:59:57	129:59:46	1378	58	03:09:26	132:09:42	1419
27	37:51:35	1617:47:45	2472	59	42:45:36	1809:57:03	2662
28	00:05:43	01:38:55	696				
29	04:00:27	05:35:45	732				
30	04:17:04	114:28:26	977				
31	36:57:44	997:28:22	1589				

### 3. TEST CASE APPLICATIONS

---

length of the computational domain seems to have an influence locally concentrated around the outflow area and needs to be established depending on the problem to be solved.

The results from this test case demonstrate that the high-order prefactored finite-difference solver, with appropriate wall boundary and plate edge and surface corrections can provide engineering accurate predictions of the unsteady wall pressure and of the near-field radiating pressure amplitude in an aeroacoustic problem of noise generated by the interaction with a straight solid wall. Therefore, the Giles non-reflecting inflow and outflow, and the Kim and Lee inviscid wall boundary conditions are surely the first choice for the application in the test cases described in the following sections. Although no direct numerical comparison is given with the results from other authors, the level of agreement with the reference solution, as defined in the proceedings of the second NASA computational aeroacoustics (CAA) workshop [Tam & Hardin, 1997] seems comparable with that of the results by Ragab & Salem-Said [2007] that is certainly the most accurate prediction available in literature for this problem.

## 3.3 Third test case: sound generation by interaction between an incident gust and a cascade of flat plates (3D)

### 3.3.1 Problem definition

The third test case is a benchmark problem from the third NASA computational aeroacoustics (CAA) workshop [Dahl, 2000]. It is the third problem under category 3: “Sound generation by interacting with a gust” and aims to reproduce the interaction of an incoming convected vortical gust with a cascade of flat aerofoils having a finite span and bounded in the  $z$  spatial direction by two parallel impermeable walls, as sketched in Figure 3.28. The coordinate system origin is on the lower  $z$  wall at the leading edge of the reference aerofoil lying on the  $x$ - $z$  plane. The  $y$  axis is normal to the aerofoil surface. The  $z$  axis forms an angle  $\alpha$  with the aerofoil leading edge of the reference blade defining the sweep of the blades. The chord of the aerofoil is equal to  $c$ , the distance between two successive plates in the  $y$  direction is  $h = c$  and the two bounding plates are separated by a distance  $l = 2.6c$ .

The mean flow is uniform across the domain. The mean velocity components in the  $y$  and  $z$  direction are equal to zero and the component in the  $x$  direction is equal to  $u_\infty = 0.5c_\infty$ , where  $c_\infty$  is the reference speed of sound, so that the mean flow Mach number  $M_0$  is equal to 0.5.

The problem is solved in non-dimensional form. The reference dimensional quantities used for non-dimensionalisation are: the speed of sound  $c_\infty$  for the velocity, the ambient flow density  $\rho_\infty$  for the density,  $\rho_\infty c_\infty^2$  for the pressure, the aerofoil chord  $c$  for the length, and  $c/c_\infty$  for the time.

### 3. TEST CASE APPLICATIONS

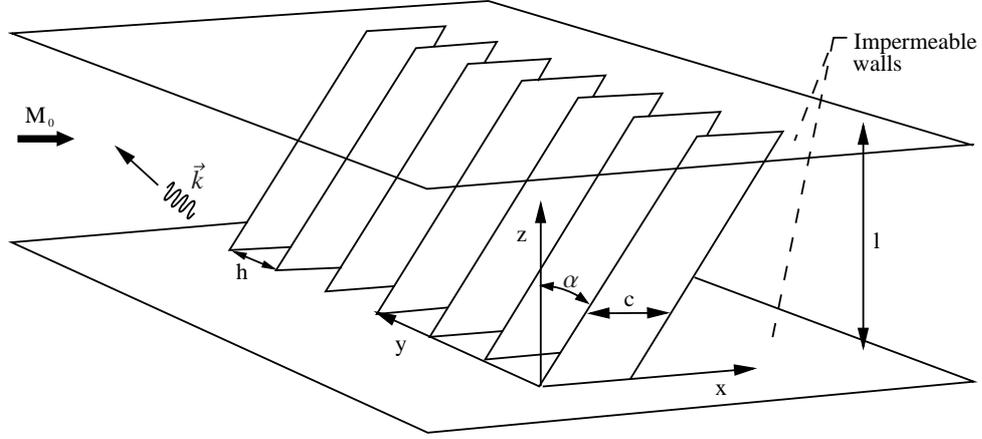


Figure 3.28: Problem geometry of test case number 3 (modified from Dahl [2000]).

The function describing the incoming gust is defined over the whole field and is function of the coordinates  $x$ ,  $y$ , and  $z$  and of time  $t$  as

$$u'_g(x, y, z, t) = 0, \quad (3.15a)$$

$$v'_g(x, y, z, t) = A \cos(k_x x + k_y y + k_z z - \omega t), \quad (3.15b)$$

$$w'_g(x, y, z, t) = 0, \quad (3.15c)$$

where  $A = 0.05$ , the wavenumbers  $k_x = 5.5/c$ ,  $k_y = \pi$  and  $k_z = 0$ , and the harmonic (angular) frequency  $\omega = k_x u_\infty / c = 2.75$ . The sweep angle  $\alpha$  ranges from  $0^\circ$  to  $30^\circ$  with steps of  $2.5^\circ$ .

The problem requires to determine the amplitude of the root mean squared radiated acoustic pressure  $p_{rms} = \sqrt{\overline{p^2}}$  in the upstream direction at point  $P = (-5c, 0, l/2)$  for each prescribed value of the sweep angle  $\alpha$ . The results are presented in dB using the  $p_{rms}$  value obtained for  $\alpha = 0^\circ$  as the reference level and are referred to as  $SPL$  in the results Section 3.3.3.

### 3. TEST CASE APPLICATIONS

---

To keep the incoming gust velocity field divergence free, and to avoid pressure and density oscillations coming from the inflow, in this work the gust definition also includes a component  $u'_g = -(k_y/k_x)v'_g$  in the  $x$  direction.

An exact analytical solution for this problem is not available. In the proceedings of the NASA workshop [Dahl, 2000] an approximate solution for the noise radiated in the upstream direction is given by Envia [2000], for which the contribution of the noise coming from the trailing edge of the aerofoils is neglected. This solution is used in Section 3.3.3 as the reference solution for comparing among the numerical results.

#### 3.3.2 Numerical methods

The equations and the data input and output are non-dimensionalised using the set of reference values defined by the problem in Section 3.3.1.

The problem definition in Section 3.3.1 does not prescribe the computational domain size or the spatial discretisation to be used. The computational domain is to be defined simply to include the blades and the point  $P$  from which the output is required. As for the second test case, clustering of the points around specific areas of the model, such as the aerofoil surface, is not used and the only distortion of the grid with respect to a uniform orthogonal Cartesian mesh is to accommodate to the sweep angle of the blades.

The overall dimension of the model in the  $z$  direction is fixed by the problem definition. In the  $y$  direction, the value of the transverse wavenumber  $k_y$  and of the inter-blade distance  $h$  determine the minimum number of vanes to be included in the model in order to obtain a pitchwise periodic flow. This gives a transverse domain extent of  $2h$ . In

### 3. TEST CASE APPLICATIONS

---

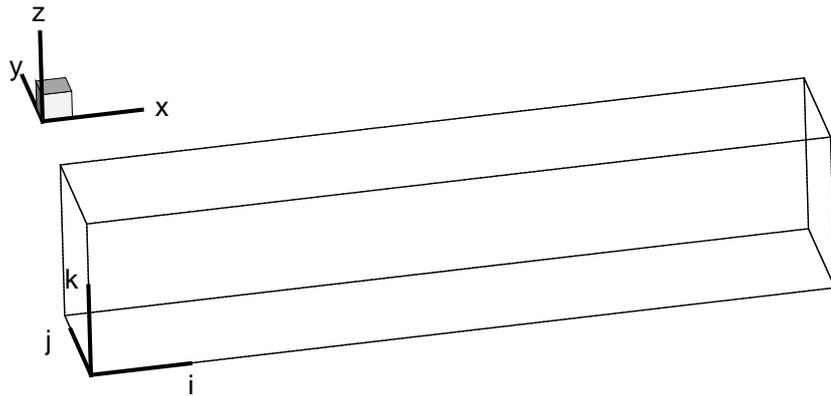
the  $x$  direction the length of the model is determined by the type of boundary closure applied at the inflow and at the outflow and by stand-off distance between the inflow and outflow boundaries to the aerofoils leading and trailing edges.

The node distribution in  $y$  and  $z$  directions over the whole model is kept as regular as possible imposing a constant distance between the successive layers of nodes. In the  $x$  direction, the faces corresponding to the inflow and the outflow can either be kept normal to the other external faces of the model, as sketched in Figure 3.29 (a), or they can follow the sweep angle of the blades, as sketched in Figure 3.29 (b). The former choice gives a “stretched” mesh and has the advantage of ensuring the orthogonality between the inflow and outflow boundaries and the curvilinear coordinate  $i$  shown in Figure 3.29 (a). The latter choice gives a “skewed” mesh and ensures a more regular distribution of nodes in the  $i$  direction for all values of  $\alpha$ , therefore simplifying the definition of the mesh and minimising the errors related to the grid deformation. Both types of mesh were used for the preliminary tests on the accuracy of the input gust.

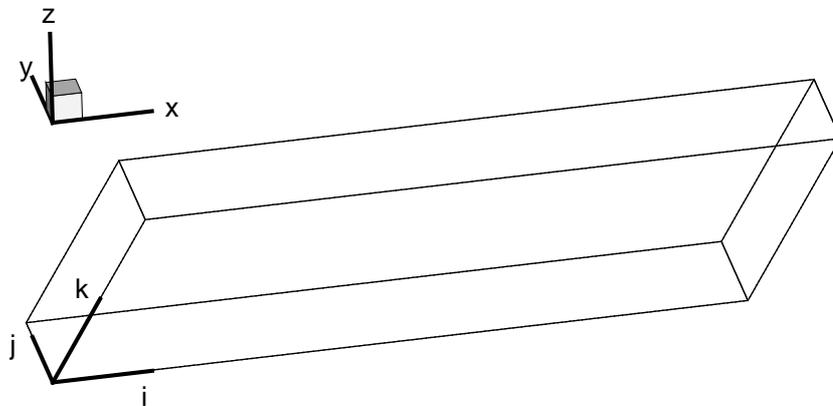
For both types of mesh, the distribution of the nodes along the  $i$  coordinate is dominated: by the position of point  $P$  where the acoustic pressure is monitored, which requires one mesh node to avoid data interpolation, by the position of the nodes on the edges of the blades, and by the position of the nodes on the outflow plane. In order to minimise the errors related to the mesh non-uniformity, a fifth-order polynomial is used to map between the physical and the computational  $i$  coordinate. The procedure is as follows. The model is divided into four zones as shown in Figure 3.30 for the “stretched” mesh. Zone 1 extends from the inflow boundary to the plane, parallel to the inflow, passing through point  $P$ . Zone 2 extends from point  $P$  to the plane defined by the blade leading edge. Zone 3 covers the area occupied by the flat plates. Zone 4 extends from the trailing edge of the plates to the outflow plane.

### 3. TEST CASE APPLICATIONS

---



(a) In the stretched mesh, the inflow and outflow planes are normal to the  $x$  axis.



(b) In the skewed mesh, the normal to the inflow and outflow planes forms an angle equal to the sweep angle  $\alpha$  with the  $x$  axis.

Figure 3.29: Computational domain of (a) the stretched mesh and (b) the skewed mesh.

### 3. TEST CASE APPLICATIONS

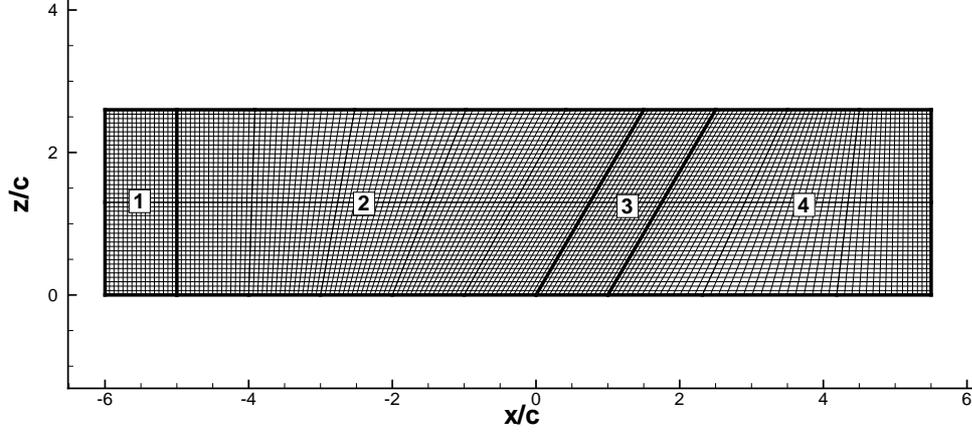


Figure 3.30: Schematic of the stretched grid highlighting the subdivision in the four zones used for the computation of the nodes along the  $x$  coordinate.

In each of the four zones, a different transformation is used to map the computational domain  $(i, j, k)$  on the physical domain  $(x, y, z)$ . For all zones,  $y = j\Delta y$  and  $z = k\Delta z$ , with  $\Delta y$  and  $\Delta z$  constant.

For zone 1, the stretched mesh streamwise coordinate  $x_s = x_u - 6c$ , where  $x_u = i\Delta x$  and  $\Delta x$  is constant.

For zone 2,

$$x_s = 6\frac{sh}{x_{le}^5}(i\Delta x)^5 - 15\frac{sh}{x_{le}^4}(i\Delta x)^4 + 10\frac{sh}{x_{le}^3}(i\Delta x)^3 + i\Delta x - 5c, \quad (3.16)$$

$$\frac{\partial x_s}{\partial i} = \Delta x \left[ 30\frac{sh}{x_{le}^5}(i\Delta x)^4 - 60\frac{sh}{x_{le}^4}(i\Delta x)^3 + 30\frac{sh}{x_{le}^3}(i\Delta x)^2 + 1 \right], \quad (3.17)$$

$$\frac{\partial^2 x_s}{\partial i^2} = (\Delta x)^2 \left[ 120\frac{sh}{x_{le}^5}(i\Delta x)^3 - 180\frac{sh}{x_{le}^4}(i\Delta x)^2 + 60\frac{sh}{x_{le}^3}(i\Delta x) \right], \quad (3.18)$$

where  $x_{le}$  is the  $x$  coordinate of the leading edge in a regular Cartesian grid and  $sh$  is the difference between  $x_{le}$  and the position of the aerofoil leading edge plane in the stretched grid for that particular row of nodes.

### 3. TEST CASE APPLICATIONS

---

For zone 3,

$$x_s = i\Delta x + sh, \quad (3.19)$$

$$\frac{\partial x_s}{\partial i} = \Delta x, \quad (3.20)$$

$$\frac{\partial^2 x_s}{\partial i^2} = 0. \quad (3.21)$$

For zone 4,

$$x_s = -6\frac{sh}{(L - x_{te})^5}(i\Delta x - x_{te})^5 + 15\frac{sh}{(L - x_{te})^4}(i\Delta x - x_{te})^4 - 10\frac{sh}{(L - x_{te})^3}(i\Delta x - x_{te})^3 + i\Delta x + sh + c, \quad (3.22)$$

$$\frac{\partial x_s}{\partial i} = \Delta x \left[ -30\frac{sh}{(L - x_{te})^5}(i\Delta x - x_{te})^4 + 60\frac{sh}{(L - x_{te})^4}(i\Delta x - x_{te})^3 - 30\frac{sh}{(L - x_{te})^3}(i\Delta x - x_{te})^2 + 1 \right], \quad (3.23)$$

$$\frac{\partial^2 x_s}{\partial i^2} = (\Delta x)^2 \left[ -120\frac{sh}{(L - x_{te})^5}(i\Delta x - x_{te})^3 + 180\frac{sh}{(L - x_{te})^4}(i\Delta x - x_{te})^2 - 60\frac{sh}{(L - x_{te})^3}(i\Delta x - x_{te}) \right], \quad (3.24)$$

where  $x_{te}$  is the position of the trailing edge in a regular Cartesian grid and  $L$  is the overall length of the undeformed model.

In order to avoid an excessive compression of the mesh in the top left-hand corner of zone 4, the overall length of the stretched type of grid was extended by the quantity  $sh_{max}$ , corresponding to the maximum value of  $sh$ , by modifying the equations used for

### 3. TEST CASE APPLICATIONS

---

zone 4 as

$$x_s = -6 \frac{(sh - sh_{max})}{(L - x_{te})^5} (i\Delta x - x_{te})^5 + 15 \frac{(sh - sh_{max})}{(L - x_{te})^4} (i\Delta x - x_{te})^4 - 10 \frac{(sh - sh_{max})}{(L - x_{te})^3} (i\Delta x - x_{te})^3 + i\Delta x + sh + c, \quad (3.25)$$

$$\frac{\partial x_s}{\partial i} = \Delta x \left[ -30 \frac{(sh - sh_{max})}{(L - x_{te})^5} (i\Delta x - x_{te})^4 + 60 \frac{(sh - sh_{max})}{(L - x_{te})^4} (i\Delta x - x_{te})^3 - 30 \frac{(sh - sh_{max})}{(L - x_{te})^3} (i\Delta x - x_{te})^2 + 1 \right], \quad (3.26)$$

$$\frac{\partial^2 x_s}{\partial i^2} = (\Delta x)^2 \left[ -120 \frac{(sh - sh_{max})}{(L - x_{te})^5} (i\Delta x - x_{te})^3 + 180 \frac{(sh - sh_{max})}{(L - x_{te})^4} (i\Delta x - x_{te})^2 - 60 \frac{(sh - sh_{max})}{(L - x_{te})^3} (i\Delta x - x_{te}) \right]. \quad (3.27)$$

Equations (3.25) to (3.27) were the transformation used in this work, although this results in sets of meshes having different overall length for the various values of the sweep angle  $\alpha$ .

The adopted mapping of Equations (3.16) to (3.21) and (3.25) to (3.27) ensures the continuity of both the first and the second derivative on the nodes located on the boundaries between the four zones.

The characteristics of the sets of computational grids used in this test case are summarised in Table 3.16. Each line of the table represents a different set characterised by the same overall nominal length in the  $x$  direction (as the actual length depends on the angle  $\alpha$ ), the same number of nodes and the same spatial orientation of the inflow and outflow planes. Each set is labelled with the common notation  $L.i.AB$  which identifies the extension in the  $x$  direction of the computational field  $L$ , the average inter-nodal distance  $\Delta x_i$  used to generate each grid and the orientation of the inflow ( $A$ ) and

### 3. TEST CASE APPLICATIONS

---

Table 3.16: Characteristics of the six sets of grids used for the solution of the third test case. The name of the set univocally determines its characteristics. The fields “Zones” indicate the number of parts in which the whole model is subdivided along the curvilinear coordinates  $i$ ,  $j$ , and  $k$  shown in Figure 3.29. The fields related to the total number of nodes “model” and “computation” refer, respectively, to the nodes of the initial model and to the nodes of the actual parallel computation. The two values are different as the nodes on the common borders are duplicated in the neighbouring processes.

Set no.	Set name	Zones			Nodes for each zone	Total no. of nodes	
		in $i$ dir.	in $j$ dir.	in $k$ dir.		model	computation
1	<b>10.1.NI</b>	10	2	2	6358	228459	254320
2	<b>10.1.NN</b>	10	2	2	6358	228459	254320
3	<b>10.1.IN</b>	10	2	2	6358	228459	254320
4	<b>10.1.II</b>	10	2	2	6358	228459	254320
5	<b>12.1.NN</b>	12	2	2	6358	273867	305184
6	<b>12.2.NN</b>	12	2	2	46827	2127125	2247696

the outflow ( $B$ ) planes. Letter N is used when the plane is normal to the  $x$  axis and letter I is used when it is inclined with respect to it.

For example, grid 10.1.II with sweep angle  $\alpha = 30^\circ$  is shown in Figure 3.31.

In all the models, the number of nodes along the three spatial directions is established to keep the cells as much as possible similar to a cube. Thus, each zone in which the complete model is subdivided has 17 nodes along the  $i$  and  $j$  directions and 22 nodes along the  $k$  direction. The only exception is set 12.2.NN in which the node linear density is doubled in all three spatial directions.

Table 3.17 shows the minimum and maximum value of stretching along the curvilinear coordinate  $i$  for each set of grids of Table 3.16. The values refer to the grids having sweep angle  $\alpha = 30^\circ$ .

The preliminary tests on the accuracy of the input vortical gust are run on the first 5 sets listed in Table 3.16, while the complete problem including the plate cascade is

### 3. TEST CASE APPLICATIONS

---

Table 3.17: Minimum and maximum stretching ratio along the curvilinear coordinate  $i$  for each set of grids in Table 3.16. The stretching ratio is computed as the ratio between two successive inter-nodal distances. For all the sets, the stretching values are computed with sweep angle  $\alpha = 30^\circ$ .

Set no.	Set name	Stretching ratio	
		min.	max.
1	<b>10.1.NI</b>	0.98238	1.01794
2	<b>10.1.NN</b>	0.95616	1.04585
3	<b>10.1.IN</b>	0.95616	1.04585
4	<b>10.1.II</b>	0.99035	1.00975
5	<b>12.1.NN</b>	0.95616	1.04585
6	<b>12.2.NN</b>	0.97778	1.02272

run on sets 5 and 6.

In the  $y$  direction, a periodic type boundary condition is enforced with the exception of the surface occupied by the plates. Numerically, the periodic boundary condition is equivalent to a standard inter-block boundary exchanging the data of five rings of nodes with the neighbour. The parallel impermeable walls normal to the  $z$  axis and the plate surfaces are modelled with the inviscid wall boundary by Kim & Lee [2004], detailed in Section 2.5.1.1. In the  $x$  direction, at the inlet, two different boundary closures have been tested: the extension to 3D of the Giles subsonic non-reflecting boundary including an additional term for the introduction of the gust, detailed in Section 2.5.2, and the extension to 3D of the modified buffer technique of Section 2.5.4, with coefficients  $\alpha$  and  $\beta$  equal to 1 and 3.5, which was already used, in its 2D version, for test case 2. At the outflow, the modified buffer technique is used as the extension of the Giles outflow to three dimensions shows a numerical instability that could not be controlled satisfactorily with the application of spatial filtering.

### 3. TEST CASE APPLICATIONS

---

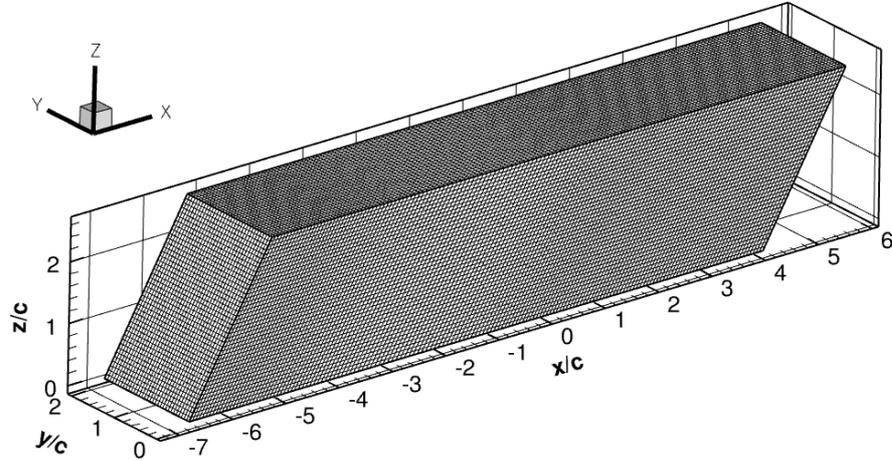


Figure 3.31: General structure of grid 10.1.II with sweep angle  $\alpha$  equal to  $30^\circ$ .

For all the simulations presented in the results Section 3.3.3, the buffer length depends on the mesh density of the model so to keep the same spatial extension: for density 1 (sets 1 to 5 in Table 3.16) it consists of 30 nodes, for density 2 (only set number 6 in Table 3.16) of 60 nodes.

In the three dimensional flat plate cascade with sweep, the specification of the boundary conditions and the introduction of the vortical gust are strictly correlated. Both the Giles and modified buffer inflow boundary conditions support the introduction of the velocity disturbance to model the inflow gust. The introduction of the gust through a body force vector  $\mathbf{f}$  following the method of Lockard & Morris [1997] is not implemented in this test case, as this approach was found to produce the largest error norms with respect to the reference analytical solution in the second test case of Chapter 3.2.

All the simulations presented in this work are run with the parallel scheme by subdividing the field in equal parts connected by the inter-block boundary condition of

### 3. TEST CASE APPLICATIONS

---

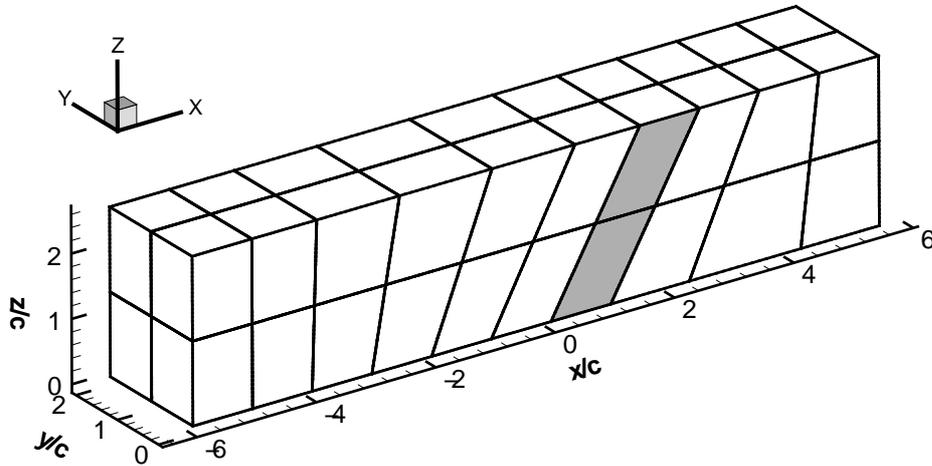


Figure 3.32: Grid 10.1.NN with sweep angle  $\alpha$  equal to  $30^\circ$  subdivided into 40 zones, which are shown by thick borders. The position of the reference blade of the swept-back cascade is highlighted in grey.

Section 2.6. The multi-block split in terms of number of nodes tends to evenly distribute the computational effort among the processors, maximising the efficiency of the parallel computation. By domain decomposition the resulting multi-block structure is such that only one type of condition is used on each block face. This aided the speed of the computation. Figure 3.32 shows the domain decomposition of grid 10.1.NN which is divided into 40 blocks of equal number of nodes. The position of the reference plate is highlighted in grey.

The computation is started from a uniform mean flow with the gust velocity per-

### 3. TEST CASE APPLICATIONS

---

turbations of Equations (3.15) superimposed on it as

$$\mathbf{Q}(x, y, z, 0) = \begin{bmatrix} \rho_\infty \\ u_\infty + u'_g(x, y, z, 0) \\ v'_g(x, y, z, 0) \\ 0 \\ \rho_\infty c_\infty^2 \end{bmatrix}, \quad (3.28)$$

where the mean field flow state  $(\rho_\infty, u_\infty, c_\infty)$  and the gust velocity perturbation are defined in Section 3.3.1.

Unlike what done for the second test case, in the third test case no exception is introduced in the algorithm for the special treatment of the nodes located on the plate surfaces and edges. This choice was dictated by the constraints on the computational resources and time available for this work.

For this test case, the linearised version of the Euler equations are solved in conservative form for compatibility with the use of the Kim and Lee inviscid wall boundary condition that require the conservative form.

Time integration is performed using the Runge-Kutta scheme proposed by Berland *et al.* [2006] in low-storage 2N form reported in Section 2.3.2.

#### 3.3.3 Results

The proceedings of the third NASA workshop include the contribution from Wang *et al.* [2000] that have presented a solution to this problem. In Wang *et al.* [2000] the problem is solved using the space-time conservation element and solution element (CE/SE) method that is based on the conservation of fluxes. Its formulation natively

### 3. TEST CASE APPLICATIONS

---

Table 3.18: Characteristics of the filters used for the third test case.

<b>Filter type</b>	<b>NBT</b>	<b><math>\alpha_f</math> coeff.</b>	<b>Application frequency</b>
explicit	LOC	0	every time step
explicit	LOC	0	every Runge-Kutta stage
implicit	LOC	0.4-0.495	every Runge-Kutta stage

allows for a simple implementation of non-reflecting boundary conditions without requiring the definition of the characteristic variables. The numerical implementation is second-order accurate in space and time.

Wang *et al.* introduce the following modifications with respect to the problem definition: (i) the non-dimensional gust amplitude  $A$  is reduced from 0.05 to 0.0001 to avoid any non-linear effect, (ii) the gust includes a component  $u'_g$  to keep the vorticity in two-dimensional form and divergence-free (also adopted in the present work), and (iii) the sweep angle  $\alpha$  ranges only from  $0^\circ$  to  $15^\circ$  with increments of  $5^\circ$  (although additional values are shown in the section of the proceedings where results are compared). The results presented in Sections 3.3.3.1 and 3.3.3.2 use the same parameters for comparison purposes.

Unless otherwise stated, the convergence of the non-dimensional pressure has been verified for each simulation. Convergence is attained when the difference between the values at the end of the penultimate (non-dimensional time  $t = 502.65$ ) and of the last period (non-dimensional time  $t = 504.93$ ) is lower than 1%.

The characteristics of the spatial filters used for this test case are shown in Table 3.18.

### 3. TEST CASE APPLICATIONS

---

The results for this test case are reported in two sections. Section 3.3.3.1 addresses the accuracy of the input vortical gust modelled without the cascade of flat plates, and Section 3.3.3.2 addresses the complete problem, including the effect of the swept-back solid boundaries flat plates cascade.

#### 3.3.3.1 Accuracy of the vortical gust

As for the second test case, a preliminary simulation is performed with the computational domain not including the linear cascade, to verify the level of accuracy that can be reached in the introduction of the vortical gust. This preliminary simulation was repeated using different models, configurations of the boundary conditions and filtering techniques. This is an important preliminary test to check the ability of the solver to convect vorticity waves on a 3D stretched grid. These results are directly correlated to the outcome of the simulation with the linear cascade, as the amplitude of the introduced gust directly affects the noise emitted in response from the plates cascade edges.

The flow in the simulations is time-marched to the non-dimensional time  $t = 102.81$ . At this time the velocity disturbance has propagated well beyond the borders of the computational field, so to test not only the ability to correctly input the vortical gust, but also the level of error introduced by the boundary conditions at the inflow and at the outflow. The non-dimensional time step is constant and equal to 0.01038, giving a total number of time steps of 9900. The minimum inter-nodal distance on the sets of meshes 1 to 5 in Table 3.16 is  $0.00625c$ . This gives a Courant number of 0.254 based on the mean flow conditions, which is below the accuracy limit of 0.846 and the stability limit of 1.918 given by the combination of the characteristics of the spatial and the time-integration schemes. When the buffer zones are used, the correction is applied

### 3. TEST CASE APPLICATIONS

---

at the end of each computational time step. The mesh skew angle  $\alpha$  at the position corresponding to the cascade leading edge is kept equal to  $30^\circ$  for all the simulations so to match the cascade sweep-back and test the effect of the maximum mesh skew on the predictions.

A selection of the simulations run to check the accuracy of the introduced vortical gust is shown in Table 3.19. The error norms are computed based on the nodes that are located between the aerofoils in the complete simulation that includes the flat plates cascade. Note that in this case, due to the dimensions of the model involved, no mesh convergence study is conducted.

The first four simulations in Table 3.19 are run on grids of nominal length  $10c$  applying the Giles non-reflecting boundary condition of Section 2.5.2 at the inflow and the modified buffer condition of Section 2.5.4 at the outflow. The simulations 3 and 4 with inclined inflow boundaries were found to be computationally unstable due to an instability arising from the inlet. Simulations 1 and 2 reached the non-dimensional end time  $t = 102.81$  but without convergence of the results on the pressure due to a numerical instability originating from the inflow. This phenomenon is illustrated in Figure 3.33, where the non-dimensional pressure oscillation on a node located in correspondence of the cascade leading edge in the complete simulations is shown for simulation 2 and 6 from Table 3.19. The unsteady pressure fluctuation amplitude from simulation 2 grows with time while the one from simulation 6 is constant. Figure 3.34 shows the iso-levels of non-dimensional pressure fluctuation  $p'$  over the external boundaries at the end of simulation 2. In the absence of the aerofoils the pressure fluctuation should be zero. Non-zero pressure fluctuation levels are shown.

To improve the numerical stability of the model, in the simulations 5 to 8 in Table 3.19 the inflow condition was changed from Giles non-reflecting inflow to the mod-

Table 3.19: Results from the simulations not including the skewed cascade of flat plates. The non-dimensional errors of the velocity components and of the pressure with respect to the analytical definition of the gust given by Equation (3.15) are shown as  $l_\infty$ -norm and  $l_2$ -norm. The norms are computed from the nodes that in the complete simulations are located between the plates. In field “Filter freq.” referring to the frequency of application of the filter, value “every RK” means that the filtering is executed at the end of every Runge-Kutta stage, while “every step” means that the application is made only when the time step is completed.

No.	Set name	Boundary conditions		Filter		Result	u [nd]		v [nd]		w [nd]		p [nd]	
		$I$ low	$I$ high	type	freq.		$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$	$l_\infty$	$l_2$
1	10.1.NI	Giles	mod. buf.	impl.	every RK	complete	4.24E-003	1.67E-003	7.68E-003	3.00E-003	6.90E-005	1.86E-005	3.55E-003	1.95E-003
2	10.1.NN	Giles	mod. buf.	impl.	every RK	complete	4.28E-003	1.68E-003	7.68E-003	3.04E-003	3.34E-005	9.82E-006	3.46E-003	2.02E-003
3	10.1.IN	Giles	mod. buf.	impl.	every RK	crashed	-	-	-	-	-	-	-	-
4	10.1.II	Giles	mod. buf.	impl.	every RK	crashed	-	-	-	-	-	-	-	-
5	12.1.NI	mod. buf.	mod. buf.	impl.	every RK	complete	1.25E-003	2.65E-004	2.19E-003	4.63E-004	2.60E-005	6.60E-006	3.97E-004	2.35E-004
6	12.1.NN	mod. buf.	mod. buf.	impl.	every RK	complete	1.24E-003	2.64E-004	2.17E-003	4.62E-004	2.62E-005	6.54E-006	3.82E-004	2.33E-004
7	12.1.NN	mod. buf.	mod. buf.	expl.	every RK	complete	2.05E-003	4.73E-004	3.59E-003	8.29E-004	4.60E-005	1.06E-005	3.68E-004	2.31E-004
8	12.1.NN	mod. buf.	mod. buf.	expl.	every step	complete	1.56E-003	3.48E-004	2.74E-003	6.10E-004	3.77E-005	8.54E-006	3.67E-004	2.32E-004

### 3. TEST CASE APPLICATIONS

---

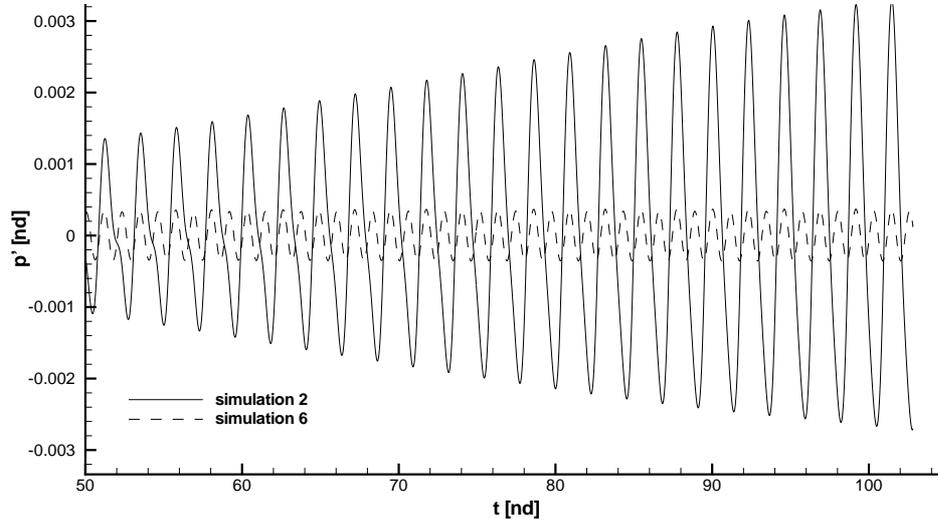


Figure 3.33: Simulations without swept-back cascade of flat plates. The time sequence of the pressure fluctuation, normalised with respect to  $\rho_{\infty}c_{\infty}^2$ , at a node located in correspondence of the cascade leading edge. Results from simulation 2 and simulation 6 in Table 3.19 are compared.

ified buffer condition of Section 2.5.4. This required extending the computational field in the upstream direction by  $2c$  to accommodate 30 additional nodes in  $i$  over which the buffer correction is applied. This change combined either with implicit filtering applied at every Runge-Kutta stage or with explicit filtering produced numerically stable simulations. Table 3.19 shows that, with the only exception of velocity component  $w$ , simulations 1 and 2 that use the Giles inflow are less accurate than simulations 5 and 6, which use the inflow buffer, by roughly one order of magnitude. Simulations 5, 6 and 8 provide similar error norms while the application of explicit filtering at every Runge-Kutta step in simulation 7 leads to error norms in the velocity components nearly twice as large as 5, 6 and 8. This is likely to be due to excessive filtering of the state variables that is corrected by lowering the frequency of application of the filter to once every time step, as shown in simulation 8. The conformation of the outflow

### 3. TEST CASE APPLICATIONS

---

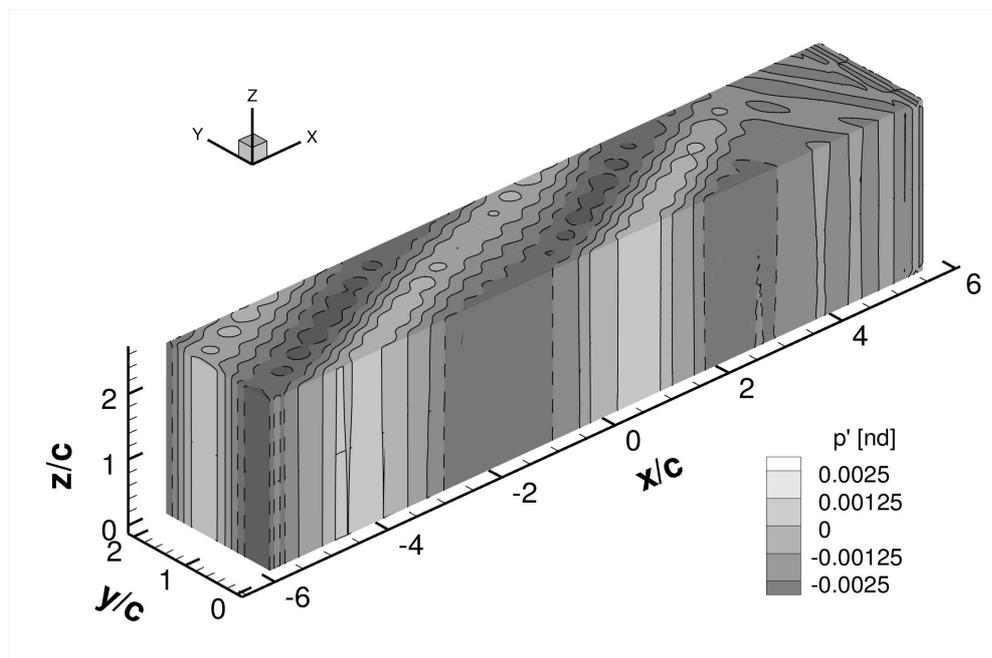


Figure 3.34: Simulation of a divergence-free velocity gust propagating in the computational domain of the skewed cascade without the plates. Pressure perturbation normalised with respect to  $\rho_\infty c_\infty^2$  over the external surfaces of model 10.1.NN at non-dimensional time  $t = 102.81$  from simulation 2 in Table 3.19. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

plane (compare simulations 5 and 6 in Table 3.19) seems to have little influence on the results. The complete problem including the flat plates cascade is only run on models with outflow plane normal to the  $x$  axis.

Based on the  $l_2$ -norm, the non-dimensional error level of the best performing simulations is two orders of magnitude smaller than the amplitude  $A = 0.05$  of the incoming gust. Therefore, even on grids relatively coarse, skewed and stretched as those used in this study, the gust propagation error is deemed adequate for attempting to model the complete problem of sound generated by the gust interacting with the swept-back cascade of aerofoils. It is useful to remind that the data shown in Table 3.19 refer to models with the maximum sweep angle  $\alpha = 30^\circ$  required by this test case [Dahl, 2000] and, therefore, the maximum level of mesh distortion, so the error level expected for all the other values of the sweep angle is lower.

Figure 3.35 shows the non-dimensional flow field obtained over the external surfaces of model 12.1.NN at the end of simulation 8 in Table 3.19, at non-dimensional time  $t = 102.81$ . The non-dimensional velocity perturbation components in axis  $x$  in Figure 3.35 (a) and in  $y$  in Figure 3.35 (b) show a regular pattern over the whole model and only small deviations can be noticed near the impermeable walls in coincidence with the peaks of the oscillations that are slightly reduced. Figure 3.35 (c) shows the pressure perturbation, normalised with respect to  $\rho_\infty c_\infty^2$ . The pressure perturbation peaks are one order of magnitude lower than the results from simulations of Figure 3.34 and thus closer to the analytical value of zero. This confirms the numerical results shown in Table 3.19 in that simulation 8 is more accurate than simulation 2 by about one order of magnitude. It can be noticed that near the computational inlet and the outlet, the modified buffer condition leads the error to a level near to zero. The results from simulations 5 to 7 give qualitatively similar results.

### 3. TEST CASE APPLICATIONS

---

Table 3.20: Computational requirements and duration of the simulations performed for the solution of the third test case in the absence of the flat plates cascade. The non-dimensional pressure error norms of table 3.19 are here replicated for the sake of completeness.

No.	Walltime	CPU time	Total mem.	p [nd]	
	[hh:mm:ss]	[hh:mm:ss]	[MB]	$l_\infty$	$l_2$
1	00:48:22	30:20:21	1793	3.55E-003	1.95E-003
2	00:36:53	23:20:41	1824	3.46E-003	2.02E-003
3	00:19:39	10:55:48	1801	-	-
4	00:18:09	09:57:18	1776	-	-
5	00:41:33	30:09:47	2292	3.97E-004	2.35E-004
6	00:46:56	31:33:08	2502	3.82E-004	2.33E-004
7	00:23:30	17:01:30	1956	3.68E-004	2.31E-004
8	00:27:47	20:11:33	2114	3.67E-004	2.32E-004

The reduction in error of one order of magnitude from the use of the modified buffer at the inflow is a positive outcome from the range of simulations in Table 3.19. This inflow condition together with a computational domain streamwise extent of  $12c$  are therefore selected for modelling the gust propagating through the swept-back cascade reported in Section 3.3.3.2.

Table 3.20 reports the computational resources used to complete the simulations listed in Table 3.19. Simulations 5 to 8 used larger memory allocation to include the inflow buffer. They also tended to use more CPU time, even if a significant variance among the computational time is shown that has probably more to do with the effect of concurrent jobs on the cluster (using communication bandwidth) than to the effect of using different solver options in the code.

### 3. TEST CASE APPLICATIONS

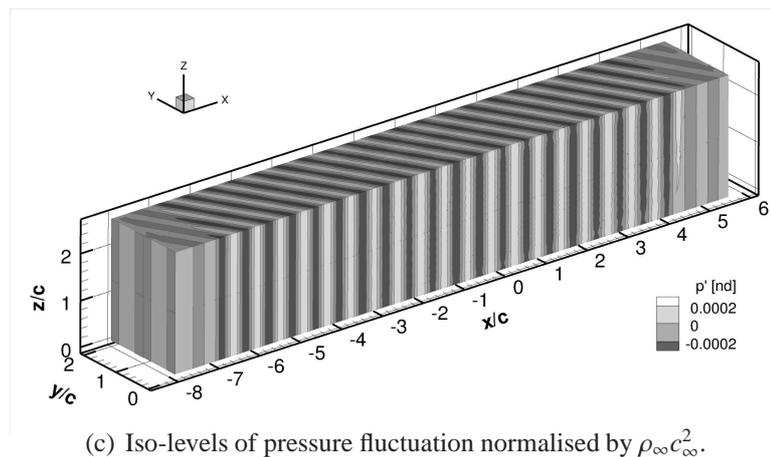
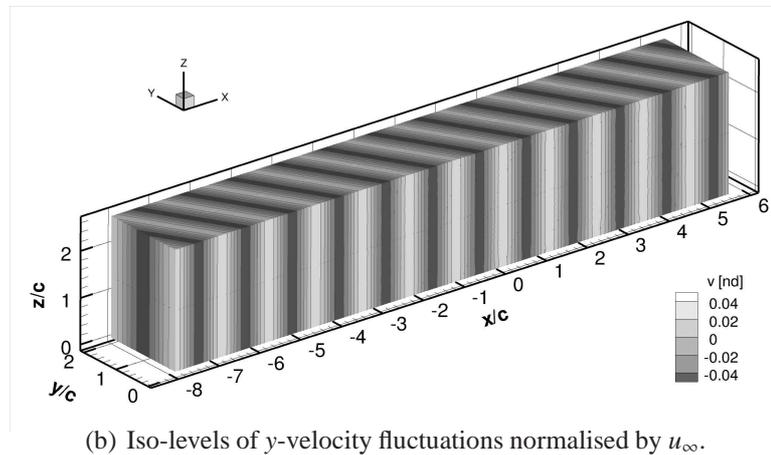
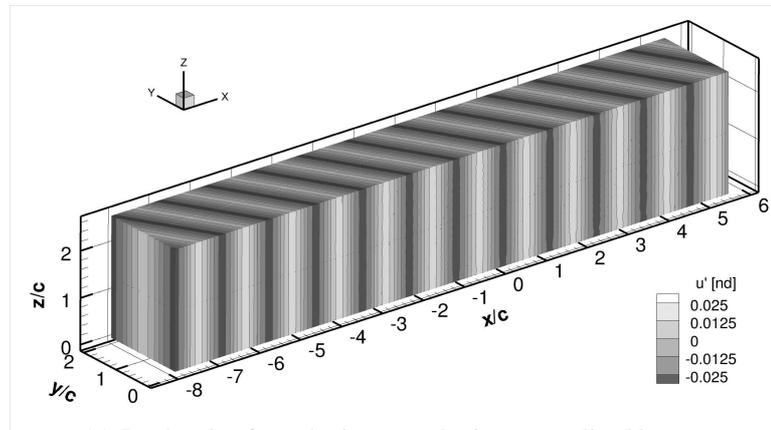


Figure 3.35: Convection of divergence-free velocity wake across the computational domain of test case 3 without the cascade walls. Instantaneous value of the flow state perturbation at non-dimensional computational time  $t = 102.81$ . Simulation 8 in Table 3.19.

### 3. TEST CASE APPLICATIONS

---

#### 3.3.3.2 Results from the complete model

This section reports the predictions of the gust of Equation (3.15) interacting with swept-back cascade of flat plates of Figure 3.28. The predictions are obtained from models that have a computational domain streamwise length of  $12c$  and use the modified buffer at the inflow and outflow.

Both the implicit and the explicit spatial filters of Section 2.2.3 have been tested, but the implicit filter resulted not capable of dealing with the high-frequency oscillations generated by the presence of the plate edges that represent a singularity of the analytical solution. Unlike the second test case, no special treatment is applied to these edges, therefore the filter performance is relied upon to keep the computation numerically stable. Hence, all the results shown in this section are obtained by explicit filtering, in its two versions shown in Table 3.18.

31 simulations were performed covering the sweep angle range  $0^\circ \leq \alpha \leq 30^\circ$  as listed in Table 3.21. All simulations are time-marched to the same non-dimensional end time  $t = 504.94$ . The simulations are grouped into three groups. Within each group the simulations differ only for the value of the sweep angle  $\alpha$ . The first group of simulations 1 to 13 and the second group of simulations 14 to 26 use the set of meshes 5 in Table 3.16. The two groups differ only for the frequency of application of the explicit filter that, for simulations 1 to 13 is applied at every Runge-Kutta stage, while for simulations 14 to 26 is applied only at the end of each time step. As prescribed by the problem definition,  $\alpha$  ranges from  $0^\circ$  to  $30^\circ$  with increments of  $2.5^\circ$ . All the simulations are time-advanced by a constant non-dimensional time step equal to 0.01038 over a total number of time steps of 48620. This satisfies the CFL condition, based on the uniform background flow state, for which the Courant number is equal to 0.254.

### 3. TEST CASE APPLICATIONS

---

The third group of simulations 27 to 31 uses the more refined set of meshes number 6 in Table 3.16. Due to the larger mesh size and computational cost of the third group of simulations,  $\alpha$  is increased in steps of  $5^\circ$  and the maximum value is equal to  $25^\circ$ . The third group of simulations 27 to 31 use a non-dimensional time step of 0.005192, resulting in 97240 steps, and a Courant number of 0.254.

Along with the characteristics of the three groups of simulations, Table 3.21 shows the computational cost of each simulation in terms of memory, CPU time, and actual duration of the run. The simulations within each group should have the same computational weight, but their actual duration shows a large dispersion. As noted for the previous test case, due to the high-performance computer job scheduler, the runs have not been made under controlled conditions.

Figure 3.36 shows the sound pressure level (*SPL*) in dB predicted at the upstream monitoring point *P* [Dahl, 2000] from the 31 simulations of Table 3.22. The dB levels are referenced to the predicted sound pressure at  $\alpha = 0^\circ$ . The results from the three groups of simulations in Table 3.22 are compared to the approximate reference analysis supplied by Envia [2000] and to the numerical results by Wang *et al.* [2000]. The three groups are referred to as “Ghillani 1” (1 to 13 in Table 3.22), “Ghillani 2” (14 to 26), and “Ghillani 3” (27 to 31). The data by Wang *et al.* [2000], not available in numerical form, were extracted from the graphs by the graph tracing program “g3data” version 1.5.3.

In general, the results compare favourably with the reference solution, with the Ghillani 1 and Ghillani 2 groups showing an improved match to the reference curve by Envia [2000] with respect to Wang *et al.* [2000]. The Ghillani 1 and Ghillani 2 predictions seem to follow the reference curve rather well over the ranges  $0^\circ \leq \alpha \leq$

### 3. TEST CASE APPLICATIONS

---

Table 3.21: Computational runs for the third test case of a wake impinging on a cascade of flat plates at different sweep angles  $\alpha$ . The list is divided into three groups. Within each group, the simulations differ only by the value of the sweep angle  $\alpha$ . Along with the characteristics of each simulation, the computational requirements and the duration of each run are shown.

No.	Model name	Angle $\alpha$ [°]	Filter type	Filter freq.	Walltime [hh:mm:ss]	CPU time [hh:mm:ss]	Total mem. [MB]
1	12.1.NN	0	expl.	every RK	01:58:25	90:01:03	1961
2	12.1.NN	2.5	expl.	every RK	02:24:18	106:40:59	2107
3	12.1.NN	5	expl.	every RK	02:10:56	100:51:03	2240
4	12.1.NN	7.5	expl.	every RK	02:33:59	111:24:45	2483
5	12.1.NN	10	expl.	every RK	01:40:55	79:10:28	2008
6	12.1.NN	12.5	expl.	every RK	02:36:41	115:43:45	2320
7	12.1.NN	15	expl.	every RK	02:36:54	115:20:35	2403
8	12.1.NN	17.5	expl.	every RK	02:15:44	102:50:18	2068
9	12.1.NN	20	expl.	every RK	02:16:33	103:42:50	2170
10	12.1.NN	22.5	expl.	every RK	02:07:11	96:51:53	2117
11	12.1.NN	25	expl.	every RK	02:38:26	116:11:27	2296
12	12.1.NN	27.5	expl.	every RK	01:50:24	86:15:28	2044
13	12.1.NN	30	expl.	every RK	01:38:12	76:35:51	1972
14	12.1.NN	0	expl.	every step	01:50:07	85:08:39	2156
15	12.1.NN	2.5	expl.	every step	01:50:25	84:08:56	2135
16	12.1.NN	5	expl.	every step	01:56:19	88:43:52	2101
17	12.1.NN	7.5	expl.	every step	01:34:27	74:34:12	2016
18	12.1.NN	10	expl.	every step	01:29:10	70:10:15	1942
19	12.1.NN	12.5	expl.	every step	02:00:07	90:38:10	2227
20	12.1.NN	15	expl.	every step	01:57:35	88:20:36	2240
21	12.1.NN	17.5	expl.	every step	01:54:29	86:12:54	2213
22	12.1.NN	20	expl.	every step	01:58:16	89:37:35	2140
23	12.1.NN	22.5	expl.	every step	02:13:49	97:40:22	2243
24	12.1.NN	25	expl.	every step	01:25:02	66:54:04	2018
25	12.1.NN	27.5	expl.	every step	01:27:52	68:02:30	2032
26	12.1.NN	30	expl.	every step	01:51:02	85:48:29	2122
27	12.2.NN	0	expl.	every RK	44:56:18	1963:42:54	5699
28	12.2.NN	5	expl.	every RK	38:35:37	1748:19:48	5695
29	12.2.NN	10	expl.	every RK	38:39:00	1725:09:23	5692
30	12.2.NN	15	expl.	every RK	33:00:52	1557:37:26	5703
31	12.2.NN	25	expl.	every RK	38:17:49	1728:25:09	5678

### 3. TEST CASE APPLICATIONS

---

$12.5^\circ$  and  $22.5^\circ \leq \alpha \leq 27.5^\circ$ , with values slightly over- and under-estimated on the right-hand side in correspondence with the secondary peak. The coarser agreement is found over the range  $15^\circ \leq \alpha \leq 20^\circ$  and at  $\alpha = 30^\circ$ , where the amplitude of the pressure fluctuation is possibly lower than the numerical noise floor in the simulation.

The Ghillani 3 group of simulations appears to clearly over-estimate the value of *SPL*. The analysis of the numerical values of the original  $p_{rms}$ , listed in Table 3.22, gives some insight on the probable cause of this over-estimation. The reference curve by Envia [2000] indicates a monotonic reduction in *SPL* with  $\alpha$  over the range  $0^\circ \leq \alpha \leq 15^\circ$ . The *SPL* predicted by the second and third groups of simulations reported in Table 3.22 shows an increase in  $p_{rms}$  over the range  $0^\circ \leq \alpha \leq 2.5^\circ$  in Ghillani 2 and over the range  $0^\circ \leq \alpha \leq 5^\circ$  in Ghillani 3. This increment appears to be a numerical artifact that impacts on the full set of Ghillani 2 and Ghillani 3 results shown in Figure 3.36, as the *SPL* levels are normalised by the  $p_{rms}$  at  $\alpha = 0^\circ$ .

Figure 3.37 shows the same data of Figure 3.36 but uses the  $p_{rms}$  at  $\alpha = 5^\circ$  from each simulation as the reference for the *SPL*, this normalisation partially removes the vertical bias in the predictions from the current numerical scheme. The apparent under-estimation of the results in the case of  $\alpha = 0^\circ$  can be explained by considering that in that case the problem is essentially 2D and the rows of nodes running along the  $k$  curvilinear coordinate of Figure 3.29 form a right angle with the bounding plates. This is an important advantage in the locations where the wall boundaries of the aerofoils and the impermeable bounding plates meet. In fact, from experience, when high-order schemes are used, and two walls meet with an angle different from  $90^\circ$ , numerical oscillations tend to originate from the corner adversely affecting the accuracy of the simulation. The first group of simulations Ghillani 1 could be less prone to this effect due to the higher frequency of application of the spatial filter.

### 3. TEST CASE APPLICATIONS

---

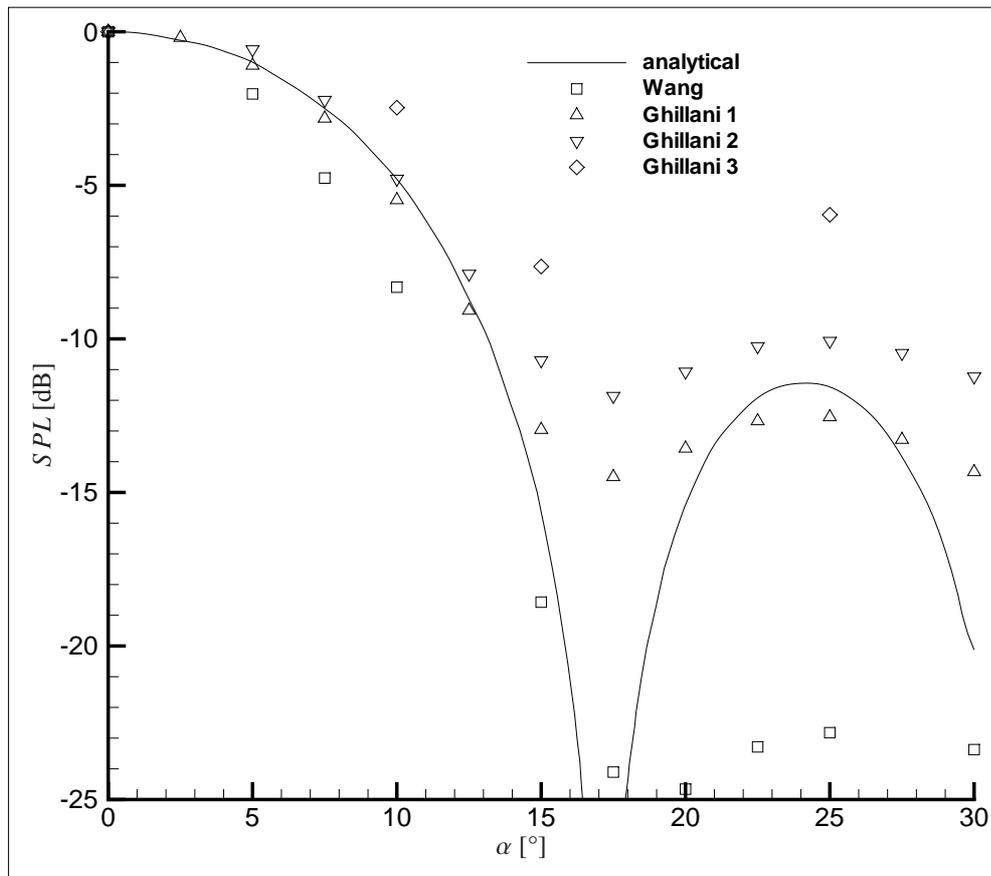


Figure 3.36: Simulations including the plates cascade: results from the three different groups (see Table 3.22) compared to the approximate reference by Envia [2000] and to the numerical solution provided by Wang *et al.* [2000].

### 3. TEST CASE APPLICATIONS

---

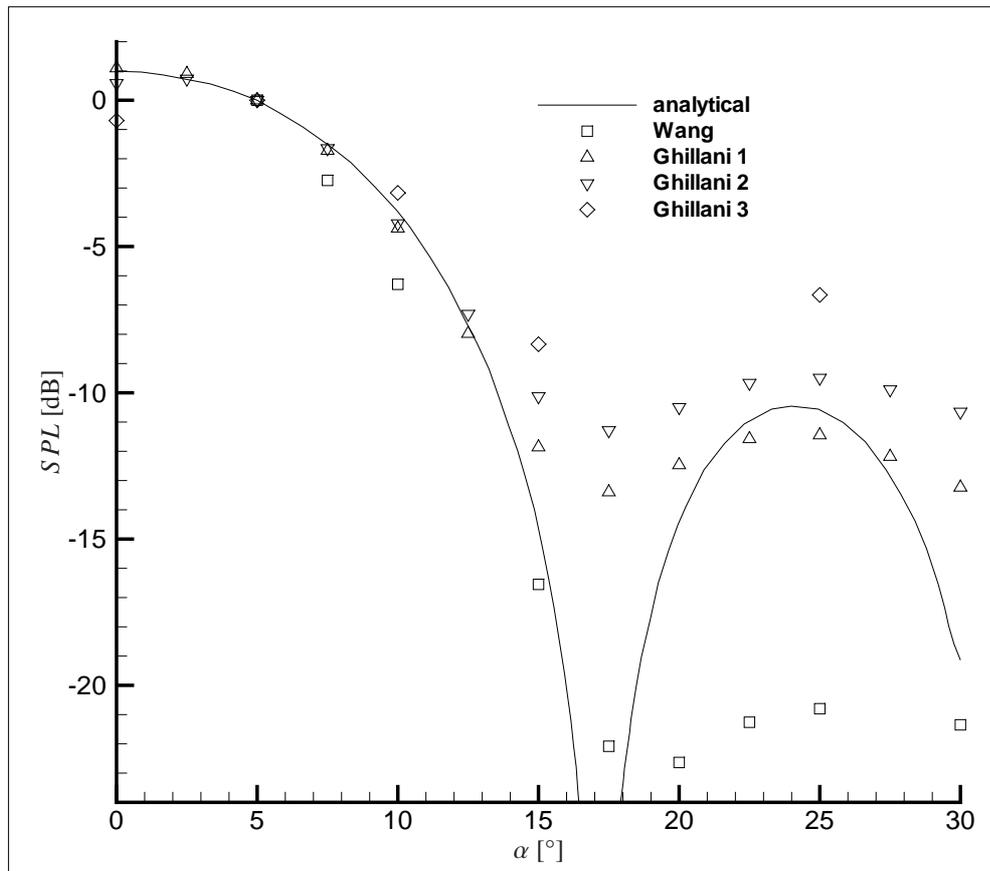


Figure 3.37: Simulations including the plates cascade: same results shown in Figure 3.36 but with  $SPL$  levels normalised by the  $p_{rms}$  at  $\alpha = 5^\circ$ .

### 3. TEST CASE APPLICATIONS

---

Table 3.22: Numerical results of the simulations executed for the solution of the third test case in the version including the plates cascade. The  $SPL$  results are illustrated in Figure 3.36. Here, in addition, the value of the non-dimensional  $p_{rms}$  is shown.

No.	Model name	Angle $\alpha$ [°]	Filter type	Filter freq.	$p_{rms}$ [nd]	$SPL$ [dB]
1	12.1.NN	0	expl.	every RK	0.02166	0.00
2	12.1.NN	2.5	expl.	every RK	0.02118	-0.19
3	12.1.NN	5	expl.	every RK	0.01908	-1.10
4	12.1.NN	7.5	expl.	every RK	0.01565	-2.82
5	12.1.NN	10	expl.	every RK	0.01153	-5.48
6	12.1.NN	12.5	expl.	every RK	0.00762	-9.08
7	12.1.NN	15	expl.	every RK	0.00487	-12.96
8	12.1.NN	17.5	expl.	every RK	0.00408	-14.49
9	12.1.NN	20	expl.	every RK	0.00454	-13.57
10	12.1.NN	22.5	expl.	every RK	0.00503	-12.68
11	12.1.NN	25	expl.	every RK	0.00511	-12.55
12	12.1.NN	27.5	expl.	every RK	0.00469	-13.28
13	12.1.NN	30	expl.	every RK	0.00416	-14.33
14	12.1.NN	0	expl.	every step	0.01373	0.00
15	12.1.NN	2.5	expl.	every step	0.01397	0.15
16	12.1.NN	5	expl.	every step	0.01285	-0.58
17	12.1.NN	7.5	expl.	every step	0.01063	-2.22
18	12.1.NN	10	expl.	every step	0.00791	-4.79
19	12.1.NN	12.5	expl.	every step	0.00554	-7.89
20	12.1.NN	15	expl.	every step	0.00400	-10.70
21	12.1.NN	17.5	expl.	every step	0.00350	-11.86
22	12.1.NN	20	expl.	every step	0.00384	-11.07
23	12.1.NN	22.5	expl.	every step	0.00422	-10.25
24	12.1.NN	25	expl.	every step	0.00431	-10.07
25	12.1.NN	27.5	expl.	every step	0.00411	-10.47
26	12.1.NN	30	expl.	every step	0.00377	-11.23
27	12.2.NN	0	expl.	every RK	0.01286	0.00
28	12.2.NN	5	expl.	every RK	0.01393	0.70
29	12.2.NN	10	expl.	every RK	0.00968	-2.47
30	12.2.NN	15	expl.	every RK	0.00533	-7.64
31	12.2.NN	25	expl.	every RK	0.00648	-5.96

### 3. TEST CASE APPLICATIONS

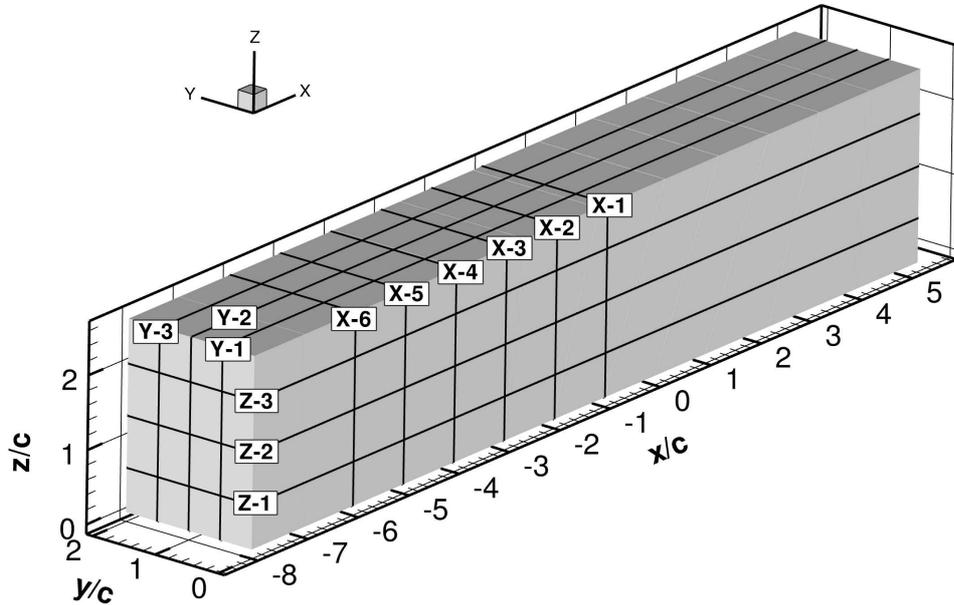


Figure 3.38: Schematic highlighting the planes from which the data illustrated in Figures from 3.39 to 3.50 are extracted.

The large quantity of data results collected can not be shown here in its entirety. Thus, only the results from simulation 31 in Table 3.22 are shown in detail. The simulation is characterised by a sweep angle  $\alpha = 25^\circ$ , mesh density level 2 in Table 3.16, and explicit filtering applied at every Runge-Kutta stage. Figures from 3.39 to 3.50 show the data obtained by slicing the model as schematically described in Figure 3.38. The six planes normal to the  $x$  axis are defined by the following equations:  $x = -c$  (labelled as X-1),  $x = -2c$  (X-2),  $x = -3c$  (X-3),  $x = -4c$  (X-4),  $x = -5c$  (X-5), and  $x = -6c$  (X-6). The three planes normal to the  $y$  axis are defined by equations:  $y = 0.5c$  (Y-1),  $y = c$  (Y-2), and  $y = 1.5c$  (Y-3). Finally, the three planes normal to the  $z$  axis are defined by equations:  $z = 0.6c$  (Z-1),  $z = 1.3c$  (Z-2), and  $z = 2c$  (Z-3).

These results show that the solver is able to address large 3D problems on grids

### 3. TEST CASE APPLICATIONS

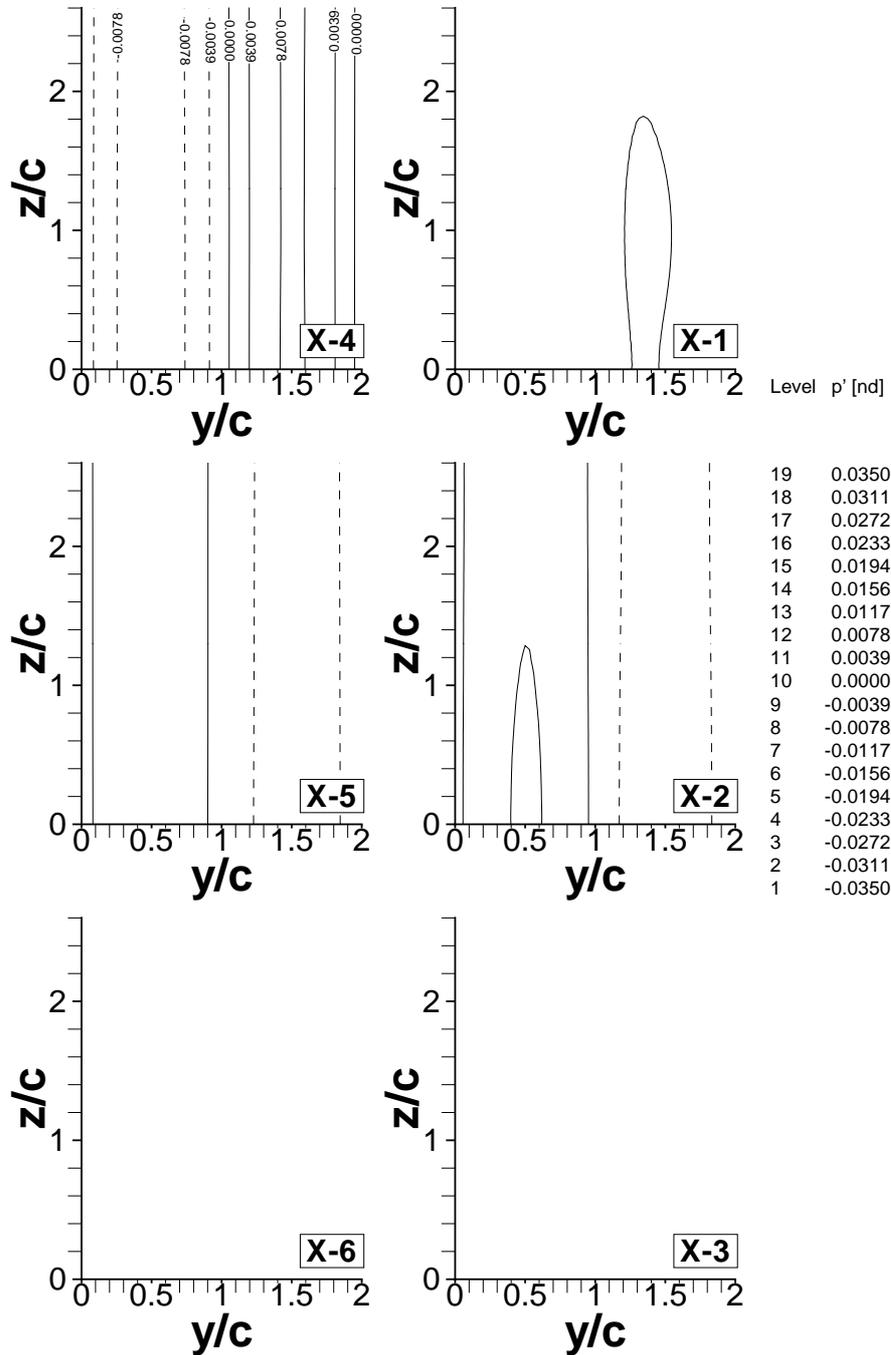


Figure 3.39: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional pressure oscillation (with respect to  $\rho_\infty c_\infty^2$ ) at the end of the run over planes normal to the  $x$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

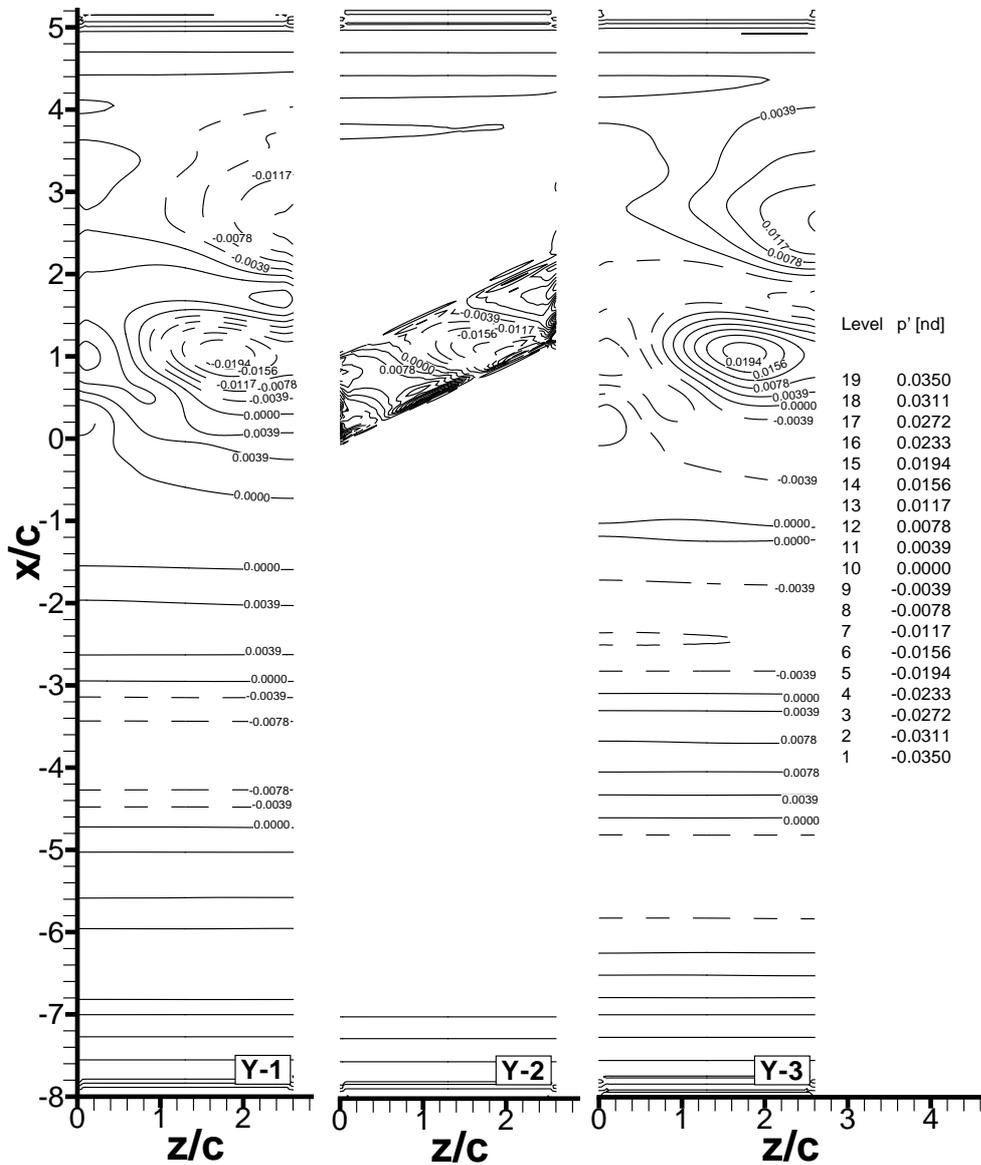


Figure 3.40: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional pressure oscillation (with respect to  $\rho_{\infty} c_{\infty}^2$ ) at the end of the run over planes normal to the  $y$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

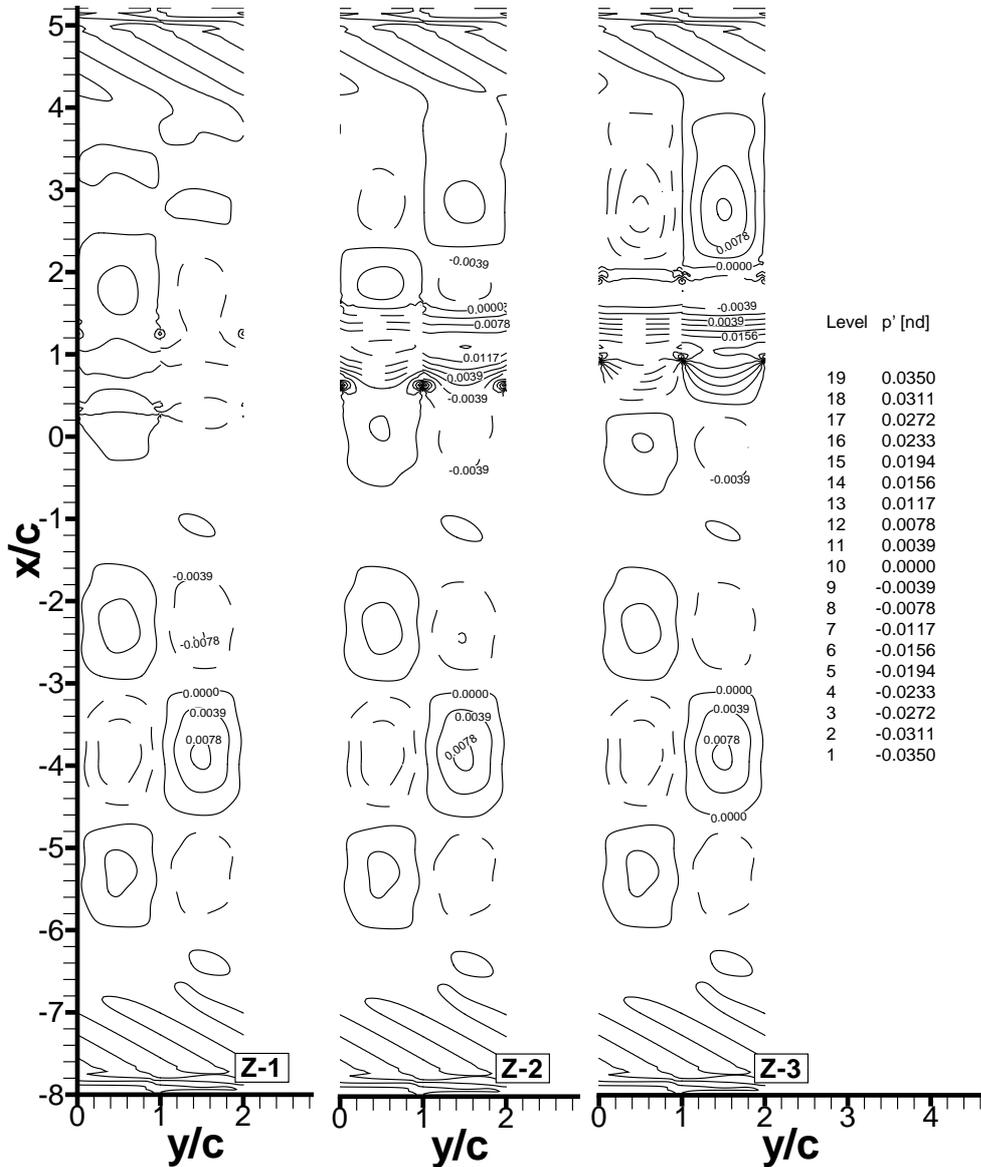


Figure 3.41: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional pressure oscillation (with respect to  $\rho_{\infty} c_{\infty}^2$ ) at the end of the run over planes normal to the  $z$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

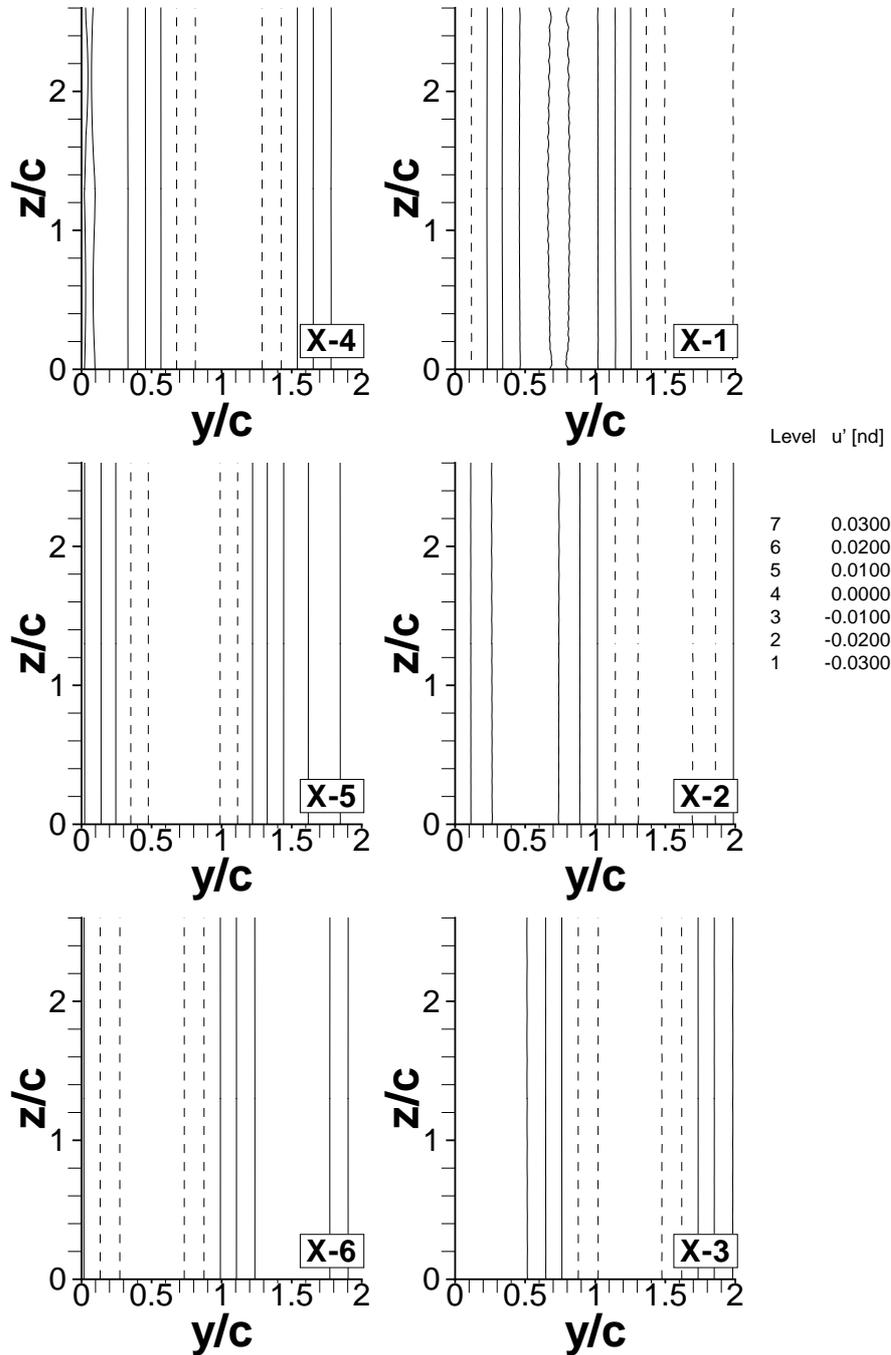


Figure 3.42: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $u$  velocity oscillation (with respect to  $u_\infty$ ) at the end of the run over planes normal to the  $x$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

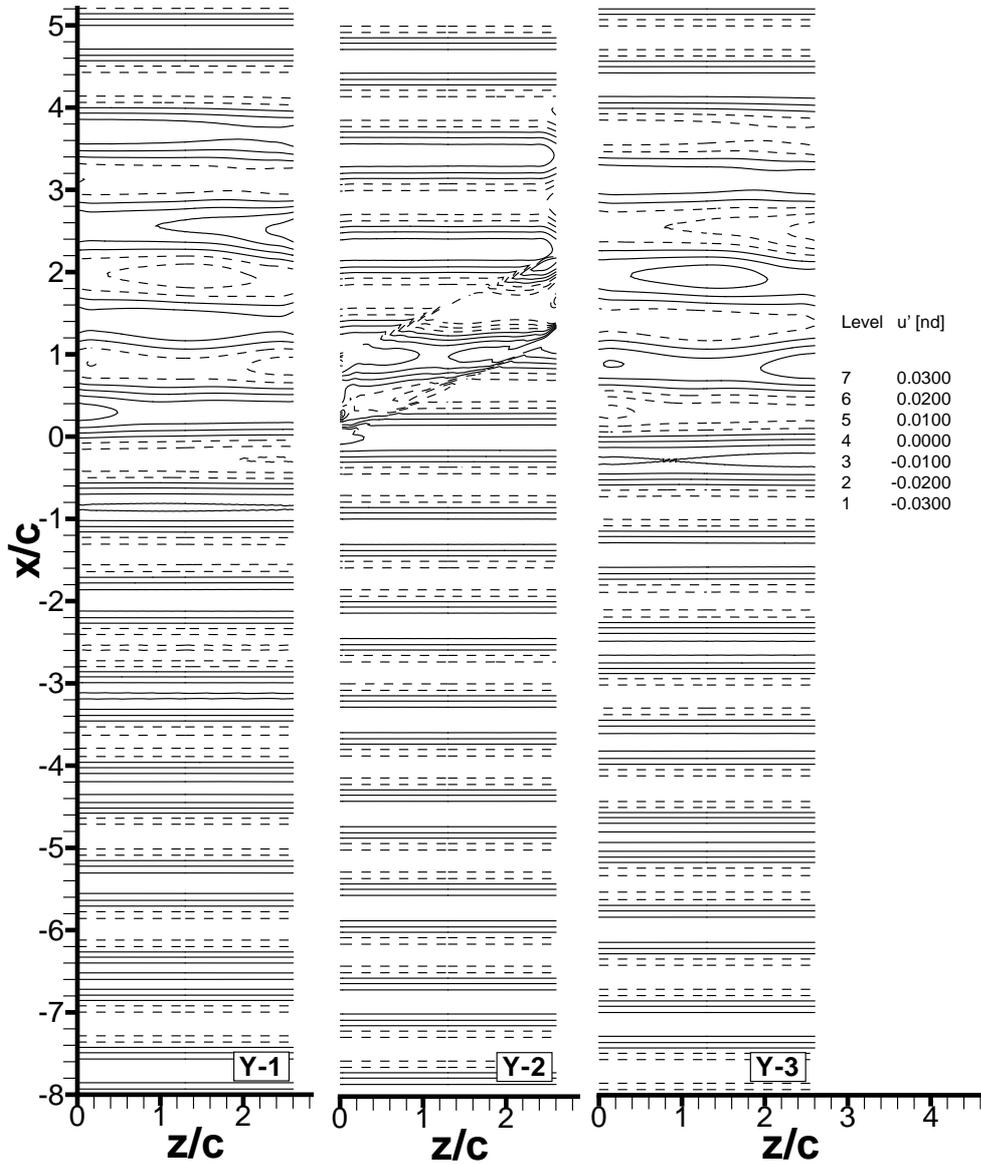


Figure 3.43: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $u$  velocity oscillation (with respect to  $u_\infty$ ) at the end of the run over planes normal to the  $y$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

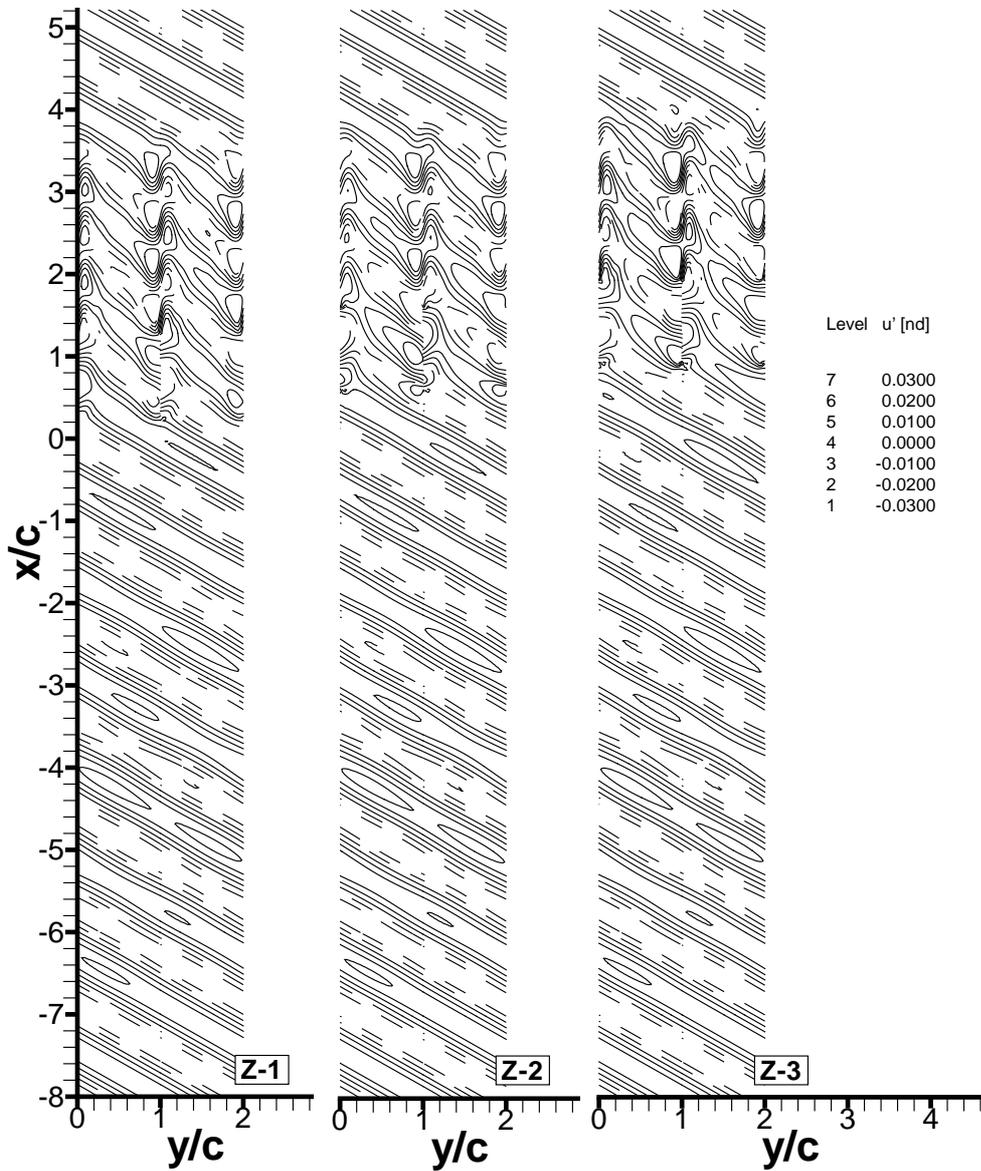


Figure 3.44: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $u$  velocity oscillation (with respect to  $u_\infty$ ) at the end of the run over planes normal to the  $z$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

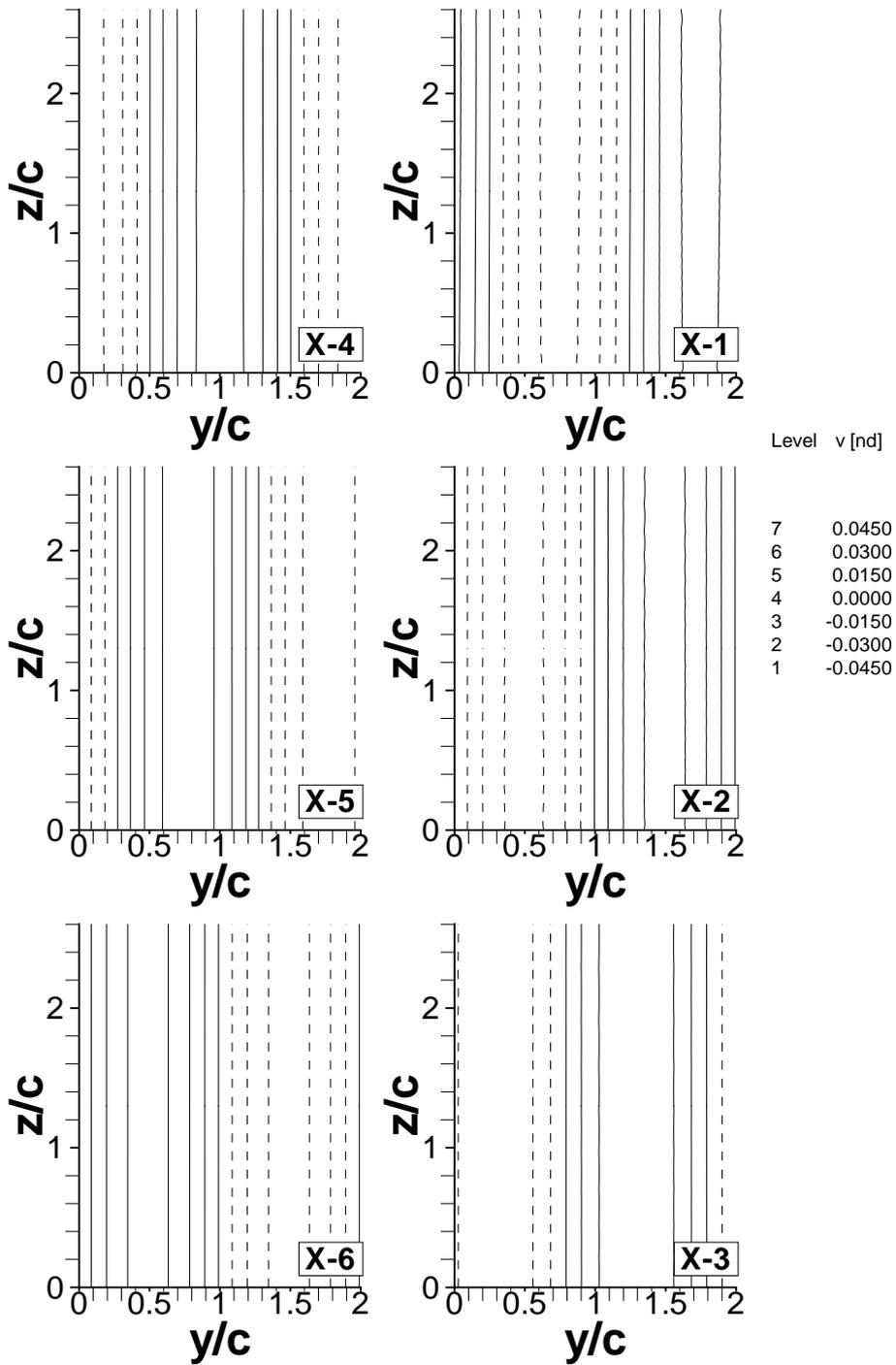


Figure 3.45: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $v$  velocity at the end of the run over planes normal to the  $x$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

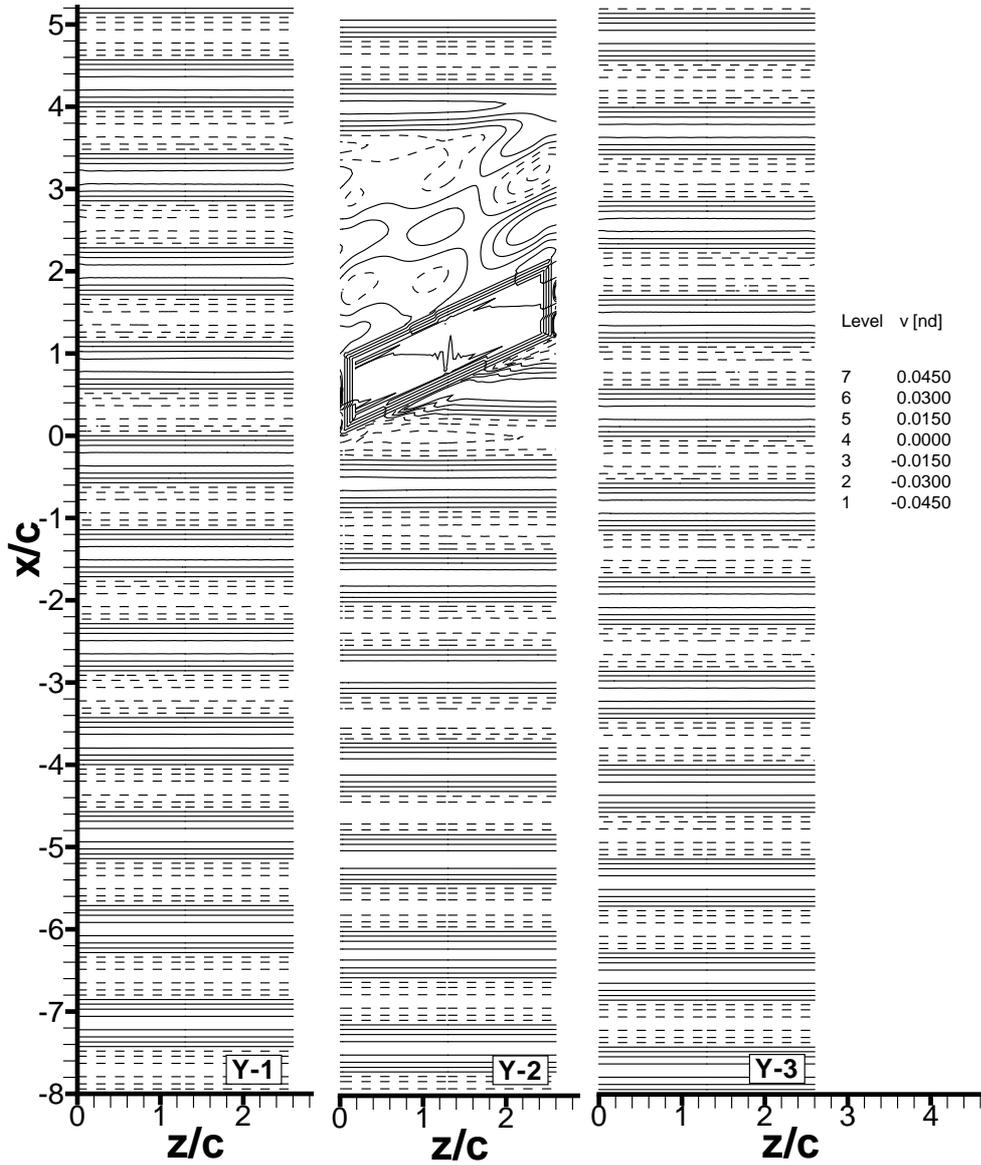


Figure 3.46: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $v$  velocity at the end of the run over planes normal to the  $y$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

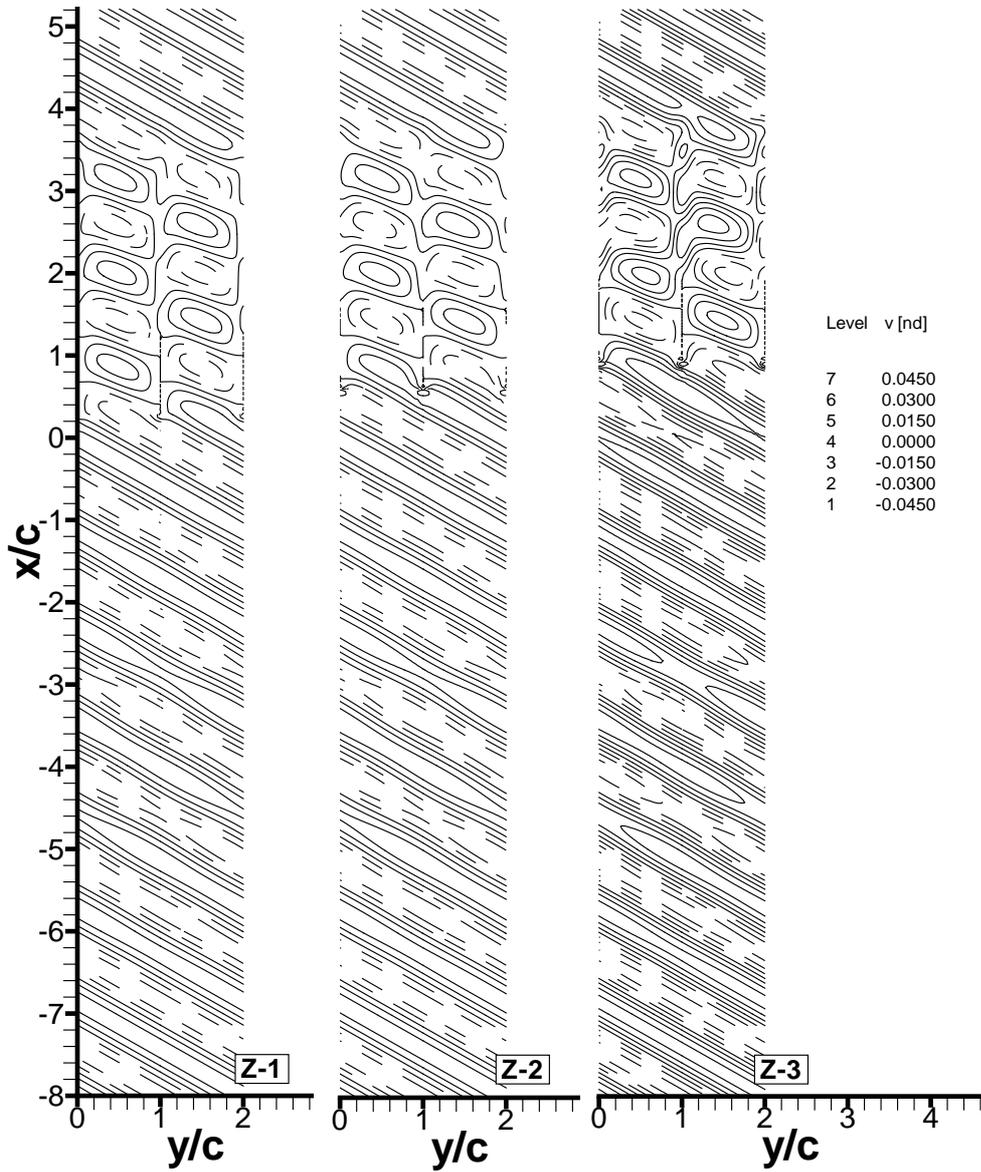


Figure 3.47: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $v$  velocity at the end of the run over planes normal to the  $z$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

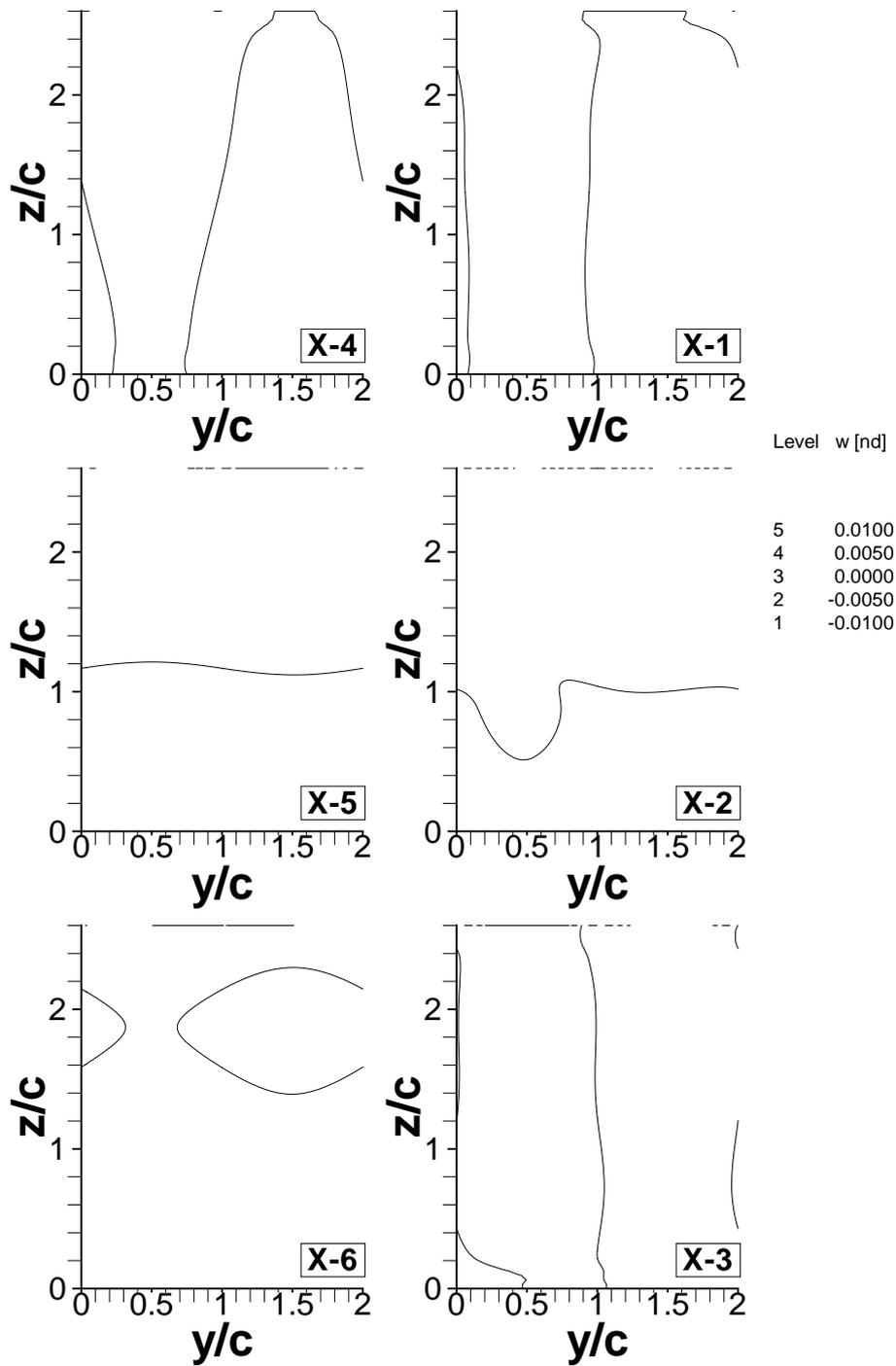


Figure 3.48: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $w$  velocity at the end of the run over planes normal to the  $x$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

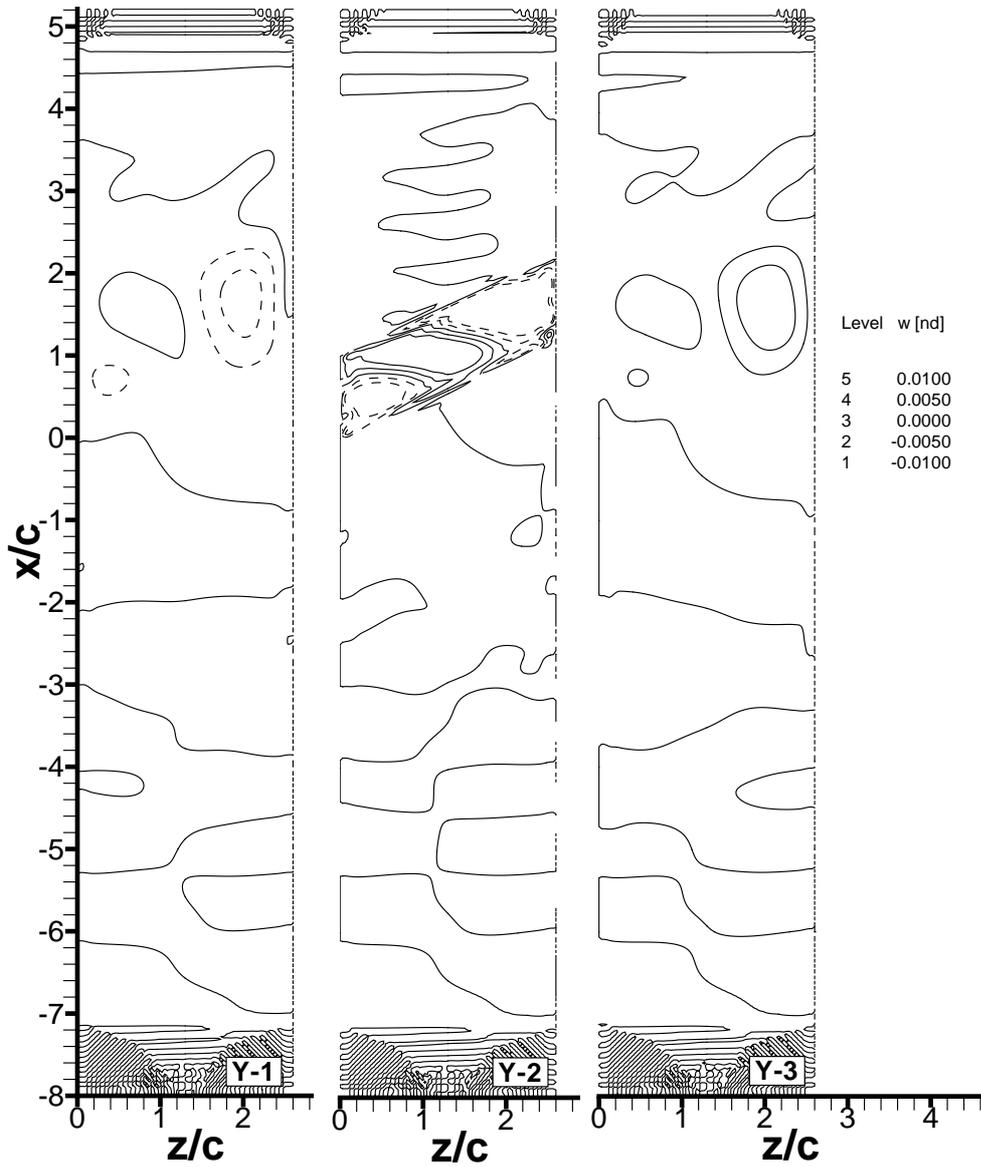


Figure 3.49: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $w$  velocity at the end of the run over planes normal to the  $y$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

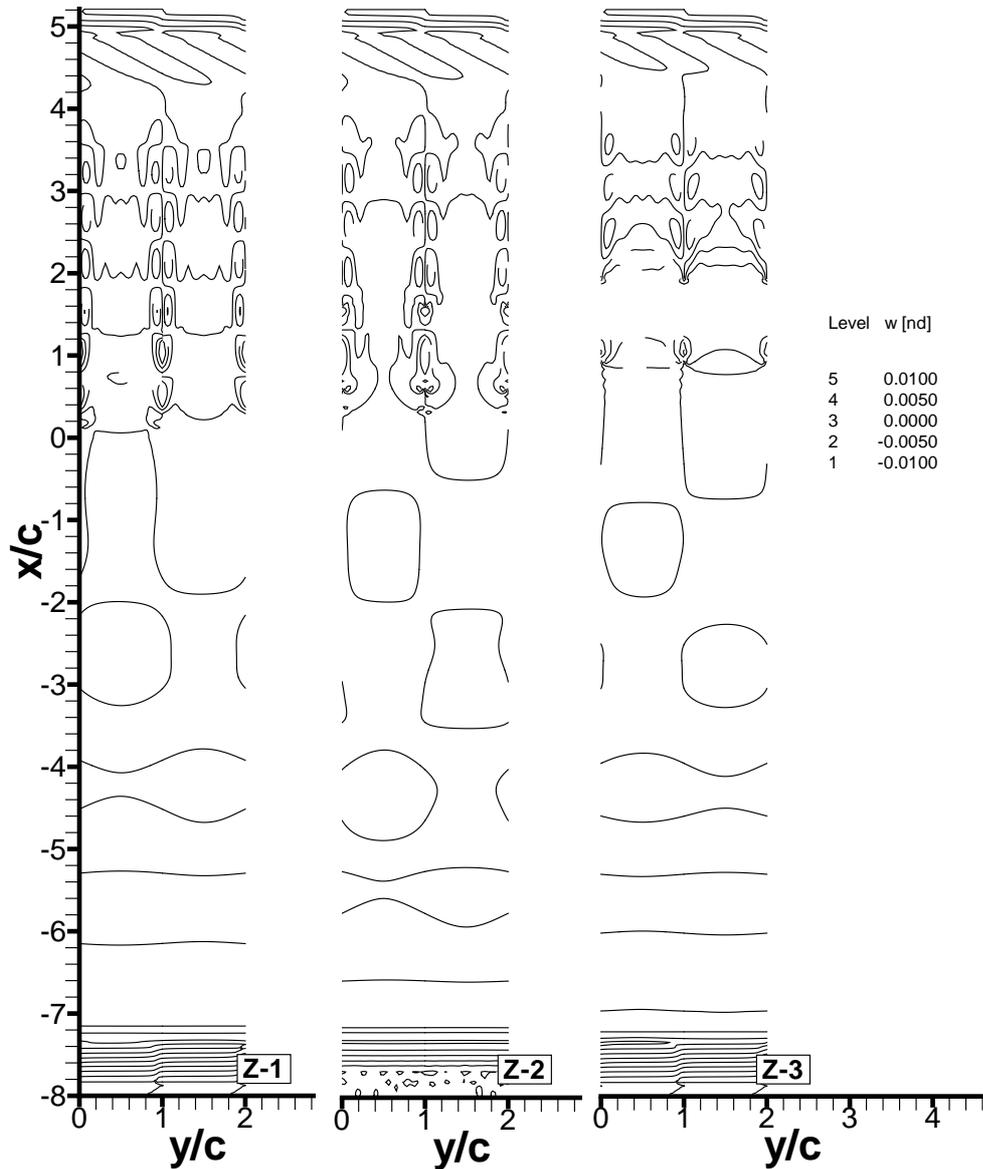


Figure 3.50: Third test case, complete problem with plates cascade. From simulation 31: distribution of the non-dimensional  $w$  velocity at the end of the run over planes normal to the  $z$  axis (see Figure 3.38). Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

containing high levels of skewness with an accuracy comparable to that of the best results available in literature [Wang *et al.*, 2000].

## 3.4 Fourth test case: sound generation by interaction between an incident gust and a cascade of aerofoils

### 3.4.1 Problem definition

The fourth test case is a benchmark problem from the fourth NASA computational aeroacoustics (CAA) workshop [Dahl, 2004]. It is the second problem under category 3: “Sound generation by interacting with a gust” and aims to study the impact of the turbulence coming from the wake of a rotor on a cascade of blades representing a guide vane stator. The problem is simplified by unrolling the profile of the blades in two dimensions and by modelling the turbulence as a vortical gust coming from the left boundary. The geometry of the problem is shown in Figure 3.51 where  $c$  is the aerofoil chord,  $d = (2/3)c$  is the cascade pitch or gap between the aerofoils, and  $\bar{\alpha}_i$  is the mean flow angle at the inflow. The profile of the blades is a representative compressor blade shape and is provided via an ASCII file containing the coordinates of 170 points lying on both the suction and the pressure sides of the aerofoil. The system of coordinates is located approximately at the geometric centre of mass of the reference aerofoil and the inflow and the outflow planes are at a distance of  $\pm 1.5c$  from the centre of the coordinate system along the  $x$  axis. The reference flow field is not specified and has to be computed starting from the mean (time-averaged) inflow and outflow boundary

### 3. TEST CASE APPLICATIONS

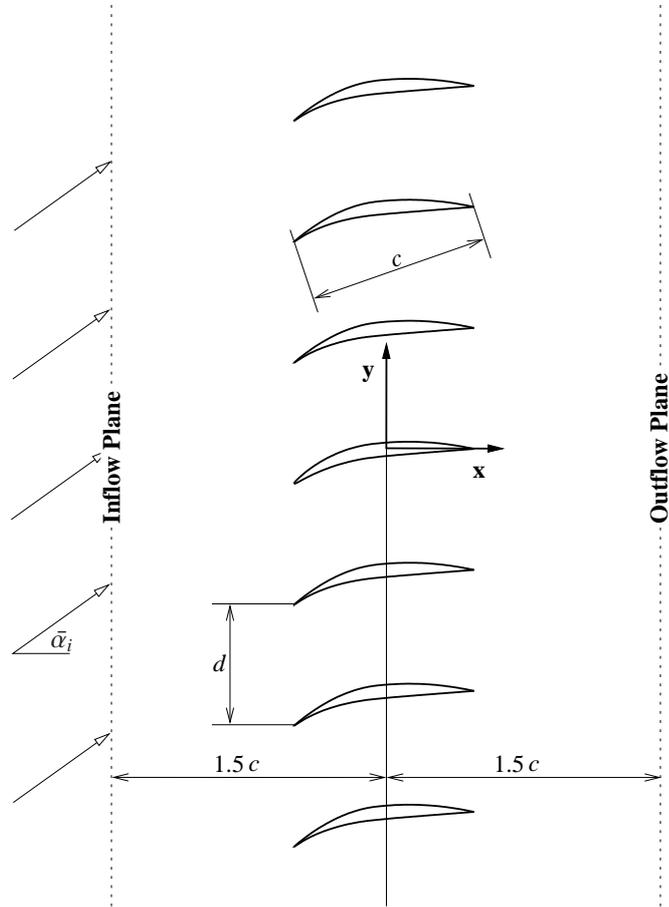


Figure 3.51: Problem geometry of test case number 4 (modified from Dahl [2004]).

conditions that are

$$\begin{cases} \bar{p}_{tot(inflow)} = 1 \\ \bar{T}_{tot(inflow)} = 1 \\ \bar{\alpha} = 36^\circ \end{cases}, \quad (3.29)$$

and

$$\bar{p}_{(outflow)} / \bar{p}_{tot(inflow)} = 0.92, \quad (3.30)$$

respectively, where  $\bar{p}_{tot(inflow)}$  and  $\bar{T}_{tot(inflow)}$  are the mean stagnation pressure  $p_{tot}$  and temperature  $T_{tot}$  at the inflow, and  $\bar{p}_{(outflow)}$  the mean static pressure at the outflow. All

### 3. TEST CASE APPLICATIONS

---

the values are normalised with respect to the following reference conditions:  $p_{ref} = 101352.93$  Pa,  $T_{ref} = 288.33$  K.

The vorticity gust is defined on the inflow plane and is a function of the  $y$  coordinate and of time  $t$

$$u'_g(y, t) = \sum_{n=1}^3 a_n \cos [n(k_y y - \omega t)] \cos \beta, \quad (3.31a)$$

$$v'_g(y, t) = -u'_g(y, t) \tan \beta, \quad (3.31b)$$

$$\rho'_g(y, t) = 0, \quad (3.31c)$$

$$p'_g(y, t) = 0, \quad (3.31d)$$

where the fundamental reduced angular frequency  $\omega = 3\pi/4$  normalised by the vane chord  $c$  divided by the speed of sound  $(\gamma RT_{ref})^{1/2}$ , the transverse wavenumber  $k_y = 11\pi/9$  normalised by the vane chord  $c$ , the harmonic amplitudes  $a_1 = 5 \times 10^{-3}$ ,  $a_2 = 3 \times 10^{-3}$ ,  $a_3 = 7 \times 10^{-4}$  normalised by the speed of sound, and  $\beta = 44^\circ$ .

The flow is assumed inviscid and isentropic and the problem requires to solve the time-dependent flow equations. The simulation has to be run until convergence (the spectra of two successive periods have to differ by less than 1% on inflow, outflow, and on the reference aerofoil at any of the three input frequencies) and, once this is achieved, the following quantities have to be computed and output:

1. pressure frequency spectra at the selected locations on the aerofoil surface having  $x$  coordinate equal to  $-0.25c$ ,  $0$ ,  $0.25c$  on both the upper (suction) and lower (pressure) side and on the inflow and outflow planes having  $y$  coordinate equal

### 3. TEST CASE APPLICATIONS

---

to  $-0.3c$ ,  $0$ , and  $0.3c$ , as sketched in Figure 3.52;

2. the harmonic pressure distribution on the inflow and outflow planes at the fundamental angular frequency  $\omega$  and at its multiples  $2\omega$  and  $3\omega$ . Then, applying a Fourier transform in the  $y$  direction, the spatial structure of the perturbations (depending on the mode order) has to be obtained.

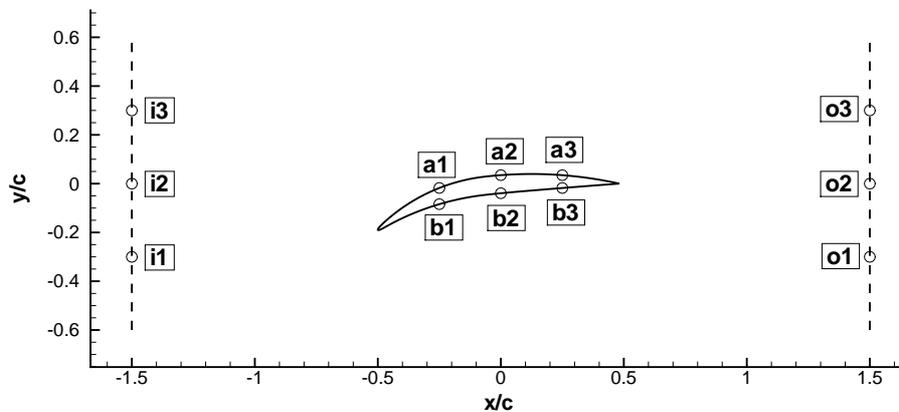


Figure 3.52: Points (indicated by circles) from which the flow spectra are output.

All the results have to be expressed in  $SPL$  in dB using as reference  $20 \mu\text{Pa}$ .

No analytical solution is known for this problem, so the authors of the problem provide a numerical solution that, for the number of nodes and for the characteristics of the employed solver can be used as a reference. To obtain this reference solution, the computations were carried out with a linearised Euler program called LINFLUX that operates in the frequency domain starting from a steady-state solution obtained from a non-linear code called TURBO that is part of the same software collection in use at NASA. To obtain the complete results, three runs at each of the frequency components of the incoming perturbation were performed by Envia [2004]. A detailed

### 3. TEST CASE APPLICATIONS

---

description of the method used to obtain the reference solution can be found in Envia [2004]. However, it has to be noted that the adopted solver is strictly three-dimensional axisymmetric and some approximations are introduced in the modelisation so that it can not be considered exact.

It is necessary to determine the number of vanes that are required in the model in order to be able to enforce pitchwise periodic boundary conditions in the  $y$  direction. In a single-stage turbomachinery CFD, the flow has a macroscopic pitchwise periodicity of  $B/V$ , where  $B$  is the number of rotor blades and  $V$  is the number of stator blades. In a rotary cascade model, pitchwise periodic planes can be  $2\pi B/V$  apart. In a linear cascade, this pitchwise periodicity can be modelled by imposing pitchwise periodic boundaries spaced  $Vd \times B/V$  apart. The  $B$  rotor blades shed  $B$  wakes at a wavenumber  $k = 2\pi B/(Vd)$ , thus  $B/V = (kd)/(2\pi)$ . The value of  $B/V$  is used to define the flow-normal extent of the simulations with the inflow gust, which is  $27d$ . The pitchwise periodicity of the flow domain without the inflow gust is  $d$ , leading to a substantially smaller domain size.

The Tyler-Sofrin rule [Tyler & Sofrin, 1962]

$$k_y^r = nk_y - k \frac{2\pi}{d/c}, \quad (3.32)$$

allows to analytically compute the dominant mode wavenumbers  $k_y^r$  of the acoustic response of the blade cascade. Let  $n$  be the harmonic component of the gust and  $k$  an integer (in this case only 0, 1 and 2 are of practical interest as the contribution to the radiated noise decreases with increasing  $n$ ). The mode orders  $m$  are proportional to  $k_y^r$  and can be computed through the constant  $R = m/k_y^r = B/k_y$ .

### 3. TEST CASE APPLICATIONS

---

Table 3.23 shows the dominant acoustic modes for each harmonic component. It is important to note that some modes are cut-off and decay exponentially when moving away from the stator blades while others are cut-on and propagate with constant amplitude in the two-dimensional field.

Table 3.23: Characteristics of the dominant acoustic modes of the linear cascade.

$n$	$k$	$k_y^r$	$m$	Type
1	0	+3.839	+11	cut-off
	1	-5.585	-16	cut-off
2	1	-1.745	-5	cut-on
3	1	+2.094	+6	cut-on
	2	-7.330	-21	cut-on

#### 3.4.2 Numerical methods

Given the nature of the problem, the solution of the non-linear version of the flow governing equations is required to deal with the large gradients that can be found for all the flow variables near the leading edge of the aerofoil. The equations are in conservative form and are non-dimensionalised for the computation following the rules given in Section 2.1.4 and using the set of reference values shown in Table 3.24. Note that this differs from the non-dimensional form of the results that are normalised following the problem definition [Dahl, 2004] as stated in Section 3.4.1.

Table 3.24: Reference values used for the non-dimensionalisation internal to the code.

Length ( $l_\infty$ ) [na]	Static speed of sound ( $c_\infty$ ) [m/s]	Static density ( $\rho_\infty$ ) [kg/m <sup>3</sup> ]	Static temperature ( $T_\infty$ ) [K]
blade chord $c$	340.4064531311	1.224528207246	288.3

### 3. TEST CASE APPLICATIONS

---

For the solution of this problem, two computational grids are used with the same number of nodes but different node distribution. Note that the two grids were used in two separate and independent series of simulations that include first the computation of the steady-flow state and then the solution of the unsteady problem that includes an incoming gust from the inflow plane. Both grids were designed using the software package GAMBIT version 2.3.16. Their characteristics are summarised in Table 3.25 where the first line refers to the models used for the steady-flow case, with the inflow wake amplitude set to zero, and the second line to the unsteady simulations. In Table 3.25, the characteristics of the grids adopted by other participants to the fourth NASA workshop are also reported for comparison purposes and labelled with the name of the main author of the paper. Escribano used different meshes for each incoming harmonic, whereas the same mesh is used for all harmonics in the current study. The meshes used in the current study are shown in Figures 3.53 and 3.54. As can be seen in the figures, the computational domain is split into three zones that correspond to the area near the inlet on the left-hand side, the channel between two blades, and the area near the outlet on the right-hand side. This gives a structured mesh of H-topology. This topology has a number of advantages. First, all the zones have the same number of nodes and they can be assigned to different parallel processes, therefore increasing the efficiency of the computation. Second, the general structure and the connections between the processes are simple, so the set-up of the problem is quicker and less error-prone. Finally, the boundary conditions are homogeneous on each face of each zone and this represents an advantage for the efficiency of the code. However, this type of topology has the disadvantage of leading to grids with a significant level of distortion near the leading edge of the blade where, due to the steep gradients in the flow field, the flow is rapidly changing in space and therefore requires a good level of spatial

### 3. TEST CASE APPLICATIONS

---

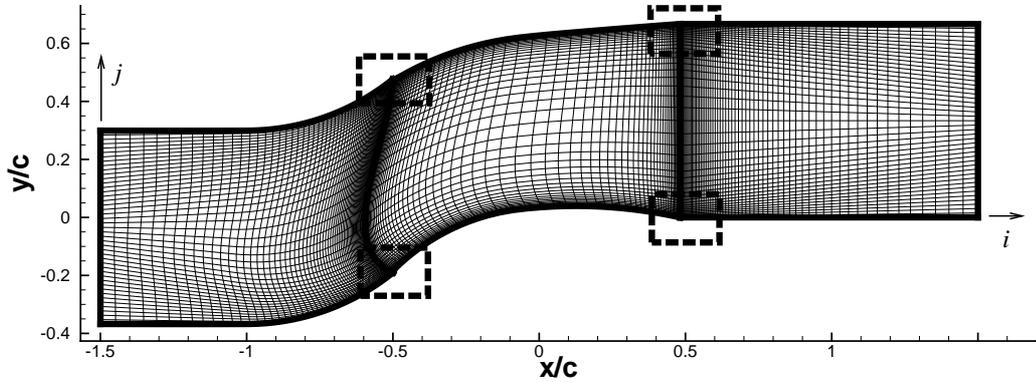
resolution. For this reason, the shape of the grid near the blade edges was carefully designed in order to keep the cells as regular as possible, as shown in Figures 3.53 (b-e) and 3.54 (b-e). The curved shape of the grid near the inlet in Figures 3.53 (b,d) and 3.54 (b,d) is chosen to follow as much as possible the streamlines in the area near the leading edge of the aerofoil and, at the same time, have the mesh lines normal to the inflow to satisfy the requirements of the Giles boundary condition. In the case of the simulations with an inflow gust, the number of vanes required is equal to 27 and the same mesh structure is vertically replicated without modifications.

Table 3.25: Characteristics of the computational meshes used in the current simulations and mesh characteristics of comparative simulations from Dahl [2004].

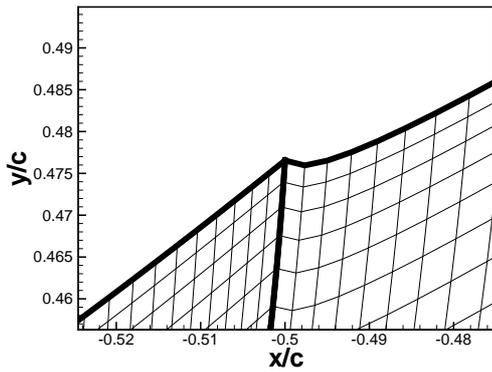
Simulation	Zones	Nodes for each zone			Total no. of nodes
		in $i$ dir.	in $j$ dir.	total	
<b>Ghillani (steady-state)</b>	3	65	45	2925	8775
<b>Ghillani (unsteady)</b>	81	65	45	2925	236925
<b>Envia</b>	na	na	na	na	170667
<b>Escribano</b>	na	na	na	na	9000/90000/360000
<b>Hixon</b>	8	na	na	na	313698

In the  $i$  direction, the zones are connected by the inter-block boundary conditions of Section 2.2.2 exchanging the data of five rinds of nodes with their neighbours. In the  $j$  direction, the models use either the periodic boundary (model cascade without inflow gust), or the inter-block boundary of Section 2.2.2 (model cascade with inflow gust). At the inlet and outlet, a modified version of Giles subsonic boundary formulation is used, which is detailed in Section 2.5.2. The modified Giles formulation includes a correction for the mean target value of the flow state similar to that proposed by Hixon *et al.* [2003, 2004]. On the aerofoil, in the direction normal to the surface, the inviscid wall boundary of Kim & Lee [2004] detailed in Section 2.5.1.2 is used. Similarly

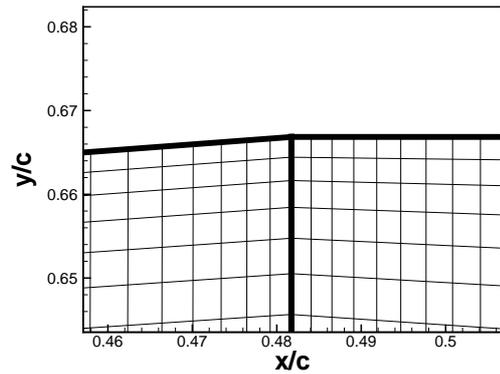
### 3. TEST CASE APPLICATIONS



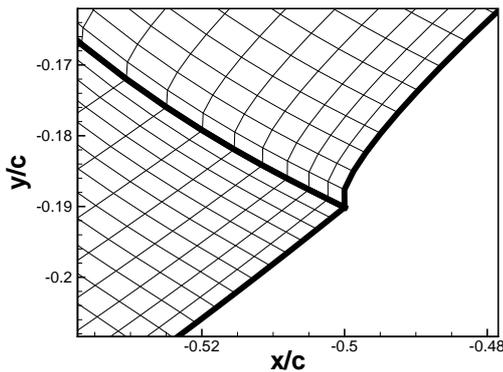
(a) General structure of the H grid number 1. The three main zones are delimited by a thick black line. The dashed boxes show the locations from which Figures 3.53 (b-e) are taken.



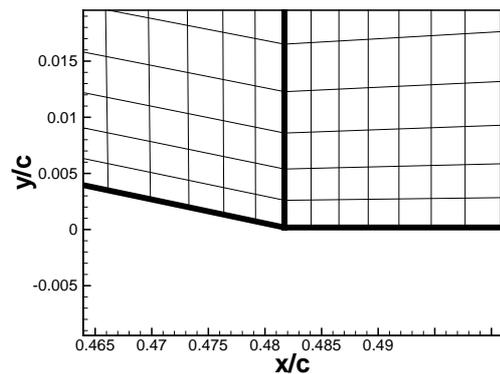
(b) Detail of the leading edge, pressure side.



(c) Detail of the trailing edge, pressure side.



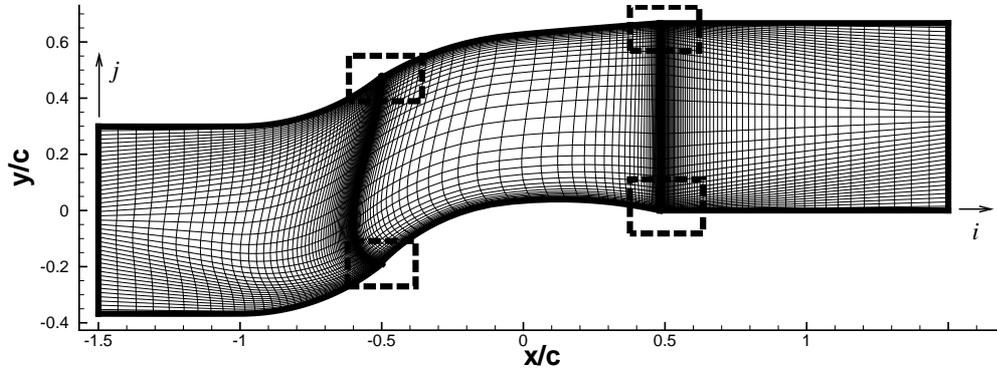
(d) Detail of the leading edge, suction side.



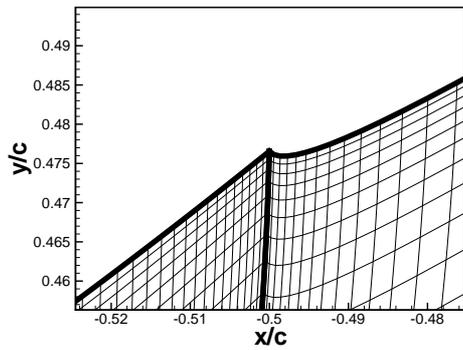
(e) Detail of the trailing edge, suction side.

Figure 3.53: Grid number 1 used for the solution of the fourth test case. The details regard the leading and the trailing edge of the blade.

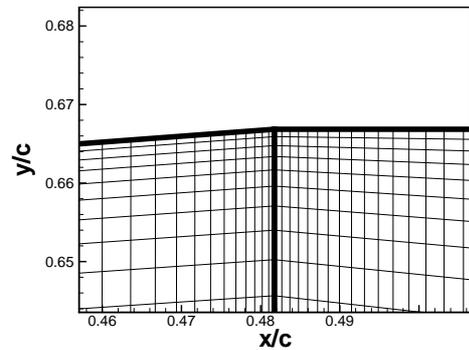
### 3. TEST CASE APPLICATIONS



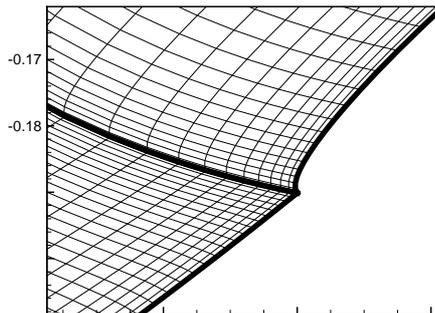
(a) General structure of the H grid number 2. The three main zones are delimited by a thick black line. The dashed boxes show the locations from which Figures 3.54 (b-e) are taken.



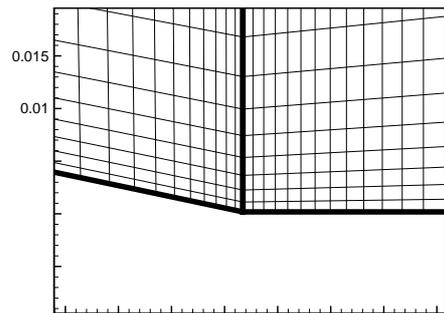
(b) Detail of the leading edge, pressure side.



(c) Detail of the trailing edge, pressure side.



(d) Detail of the leading edge, suction side.



(e) Detail of the trailing edge, suction side.

Figure 3.54: Grid number 2 used for the solution of the fourth test case. The details regard the leading and the trailing edge of the blade. The images are directly comparable to those of Figure 3.53. The higher node density in the proximity of the edges is shown. On the other hand, grid number 2 is coarser near the centre of the zone located within the blades.

### 3. TEST CASE APPLICATIONS

---

as to what is done for the second test case in Section 3.2, the high-order one-sided 7-point stencil half-derivatives are replaced by second-order 3-point stencils along the solid walls in the direction normal to them to enhance the computational stability of the simulation. This is the only modification with respect to the standard configuration of the solver and no exception was added to deal with the singularities associated to the leading and the trailing edge of the aerofoil.

The only difference in the boundary conditions applied between the steady and the unsteady runs is the introduction of the vortical gust, which is fed into the computational domain by the method detailed in Section 3.4.3.2.

The simulations of the linear cascade of aerofoils without an inflow gust starts from a uniform flow field that in non-dimensional form is

$$\mathbf{Q} = [\rho_0, u_0, v_0, p_0]^T = [1, 0.275, 0.2, 0.92]^T . \quad (3.33)$$

In Equation (3.33) the value of the non-dimensional pressure  $p_0$  is equal to that prescribed at the outflow by the problem definition and the other variables are set to levels compatible with the expected flow. The final flow state of this simulation is then replicated vertically for each of the 27 vanes and used as the starting flow state for the simulation of the cascade with the inflow gust.

The characteristics of the spatial filters used for the solution of this test case and their frequency of application are shown in Table 3.26.

The time-integration is performed using the Runge-Kutta scheme detailed in Sec-

### 3. TEST CASE APPLICATIONS

---

Table 3.26: Characteristics of the spatial filters used for the fourth test case.

<b>Filter type</b>	<b>NBT</b>	<b><math>\alpha_f</math> coeff.</b>	<b>Application frequency</b>
explicit	LOC	0	every time step
implicit	LOC	0.4-0.495	every time step

tion 2.3.2 proposed by Berland *et al.* [2006]. The time-integration uses the time steps shown in Table 3.27, which are essentially function of the smallest inter-nodal distance in the grid for both the steady and the unsteady simulations. The CFL condition is checked at the beginning of each simulation and the computed Courant numbers are below the values of 0.846 and 1.918, which are respectively the accuracy and the stability limits given by the combination of the spatial and the time-integration schemes.

Table 3.27: Runge-Kutta constant time steps used in the fourth test case.

<b>Grid no.</b>	<b>Min. inter-nodal distance [nd]</b>	<b>Time step [nd]</b>	<b>Courant no.</b>
1	1.69E-3	0.000694	0.58
2	2.43E-4	0.000115740	0.68

#### 3.4.3 Results

The proceedings of the fourth NASA workshop include the contributions from five different research groups that have presented a solution to the fourth test case. A sixth contributor (Shieh *et al.*) was able to submit only partial numerical results without added documentation [Dahl, 2004]. The presented solutions can be divided into two groups based on the numerical approach. The first group of solutions are by Bin *et al.*

### 3. TEST CASE APPLICATIONS

---

[2004]; Hixon [2004]; Nallasamy *et al.* [2004]; Wang *et al.* [2004] and are obtained with codes operating in the time-domain (both of finite-difference and finite-volume type). The second group of solutions are by Coupland [2004]; Escribano *et al.* [2004] and are from frequency-domain type of solvers. In the first group both high-order and second-order schemes are used. These solutions from both groups together with the reference solution in Envia [2004] are compared against the predictions from the current work in Sections 3.4.3.1 and 3.4.3.2. They are referred to with the name of the main author of the conference paper. Unfortunately, the published numerical values are incomplete.

It has to be noticed that not all of the authors adopted the value imposed by the problem definition for angle  $\beta = 44^\circ$ . In particular, Coupland [2004] used  $\beta = 45^\circ$  to make the gust purely 2D vortical, and Hixon [2004]  $\beta = 50^\circ$  without adding the reasons for the choice made. It is possible that a larger angle could be beneficial to avoid or reduce the discontinuity in the velocity component along the  $x$  direction that most of the authors noticed in correspondence of the trailing edge of the aerofoil. This aspect is discussed more in detail in Sections 3.4.3.1 and 3.4.3.2.

#### 3.4.3.1 Steady flow through an aerofoil cascade without inflow gust

A steady flow prediction of the linear cascade of aerofoils from Dahl [2004] was obtained by time-marching the time-accurate numerical scheme until each residual in non-dimensional form had reached a value either stable or lower than  $10^{-6}$ . The steady-state simulations were performed in time-accurate fashion. This simulation differs from the unsteady flow results of Section 3.4.3.2 in that (i) the computational domain pitchwise boundaries are one cascade pitch apart, and (ii) no gust is fed at the inflow. The difference between the initial field conditions and the converged solution generate

### 3. TEST CASE APPLICATIONS

---

a strong flow transient that is damped by the use of explicit filtering. Once obtained a partial convergence, the less dissipative implicit spatial filter of Section 2.2.3 with the coefficients as in Table 3.26 is used. The steady flow simulation is a precursor for the unsteady flow simulation reported in Section 3.4.3.2, which uses the steady flow prediction as its starting flow state. These runs are performed on multiple processors by dividing each of the three zones of Figures 3.53 and 3.54 into four or six parts to reduce the computation wall time. In Table 3.28, the main characteristics of each steady flow simulation in terms of duration, number of processes, and memory requirements are summarised. For grid number 2, the duration of the simulations in terms of non-dimensional time is halved with respect to grid 1 because the reached final accuracy is considered sufficient.

Table 3.28: Computational requirements and duration of the simulations performed for the steady flow solution of the fourth test case.

Grid	Simul.	Filter	Final Time [nd]	Processes	Walltime [hh:mm:ss]	CPU time [hh:mm:ss]	Total mem. [MB]
1	1	explicit	480	12	02:01:42	23:37:01	340
	2	implicit	480	12	02:04:10	24:09:59	348
2	1	explicit	240	12	52:42:35	548:08:42	536
	2	implicit	240	18	23:40:58	425:24:08	696

For simulation number 2 on grid number 1 of Table 3.28, Figures 3.55 and 3.56 illustrate the convergence history of the non-dimensional error between the target flow state from Dahl [2004] and the predicted flow state at two points located on the inflow and on the outflow. At the computational domain inflow, the problem definition in Dahl [2004] states the stagnation pressure  $p_{tot}$ , stagnation temperature  $T_{tot}$ , and flow angle  $\alpha$ . A fourth condition is given by setting the amplitude variation of the outgoing characteristic wave, in this case  $R_5$  in Equation (2.103), to zero. In a similar way, at the

### 3. TEST CASE APPLICATIONS

---

outflow, the static pressure  $p$  is imposed and the amplitude variation of three outgoing characteristic waves  $R_1$ ,  $R_3$ , and  $R_4$  in Equation (2.107a) is set to zero. As shown in Figure 3.55, the flow state at position  $(-1.5c, -0.307c)$  located on the inflow boundary reaches a non-dimensional difference in  $p_{tot}$ ,  $T_{tot}$ , and  $\alpha$  with respect to the target flow state of  $10^{-5}$  while the variation of  $R_5$  descends exponentially. At the outflow, as shown in Figure 3.56, the behaviour is similar with the non-dimensional error on the characteristic waves decreasing exponentially with increasing computational time and the difference between the outflow static pressure and the target static pressure that settles on a value slightly smaller than  $10^{-3}\rho_\infty c_\infty^2$ . This difference in outflow pressure is likely to be related to a shear layer that forms downstream of the cascade trailing edge, which is shown in Figures 3.58 and 3.59. The shear layer effect on the numerical prediction of the pressure field is further discussed in the context of Figures 3.58 and 3.59 later on. The convergence results for grid number 2 are similar to those in Figures 3.55 and 3.56.

Tables 3.29 and 3.30 show the flow state averaged over the inflow and the outflow planes at the end of the second simulation (with implicit filtering) on both grids of Table 3.25. With respect to the numerical reference solution supplied by Envia [2004], the Mach number at the inflow appears to be over-estimated and, as the non-dimensional target stagnation pressure is equal to 1, therefore the static temperature and pressure are slightly under-estimated. The predicted inflow angle  $\alpha$  does not show any significant difference with the reference prediction by Envia [2004].

Figure 3.57 shows the non-dimensional pressure distribution over both surfaces of the reference blade at the end of simulation number 2. The results obtained from the two grids in Table 3.25 are similar to one another and display the recognisable char-

### 3. TEST CASE APPLICATIONS

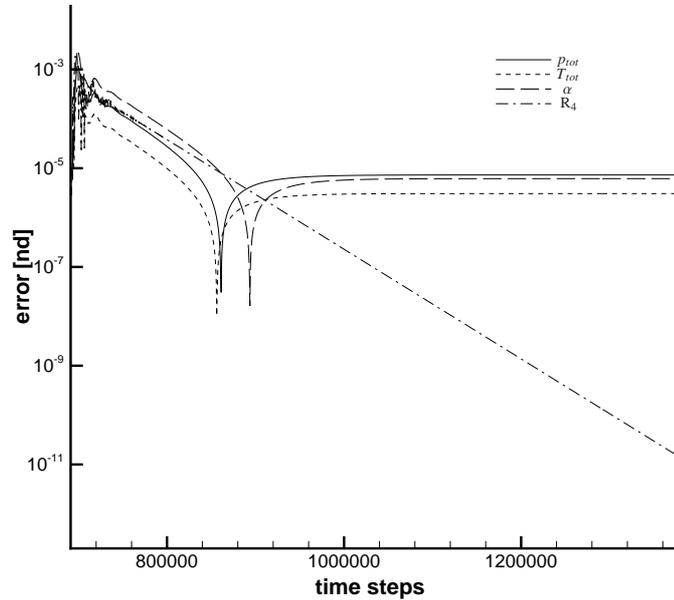


Figure 3.55: Cascade of aerofoils without inflow gust. Results from simulation 2 on grid 1 in Table 3.28. Non-dimensional difference between the target flow state and the predicted flow state on inflow at non-dimensional coordinates  $(-1.5, -0.307)$ . The number of time steps includes those from simulation number 1. 1 every 500 time steps is shown, for clarity.

Table 3.29: Fourth test case, results on the inflow plane at the end of simulation number 2 in Table 3.28. The values are averaged over the boundary and, except for the angle  $\alpha$ , are in non-dimensional form.

Author	Envia	Ghillani		Coupland	Bin
		grid 1	grid 2		
<b>Mach</b>	0.44958	0.48103	0.48224	0.448	0.4365
<b>Static <math>p</math></b>	0.87049	0.85483	0.85420	na	0.876
<b>Static <math>T</math></b>	0.96115	0.95373	0.95350	na	na
<b><math>\alpha</math> [°]</b>	36.00	35.99	35.99	na	36

### 3. TEST CASE APPLICATIONS

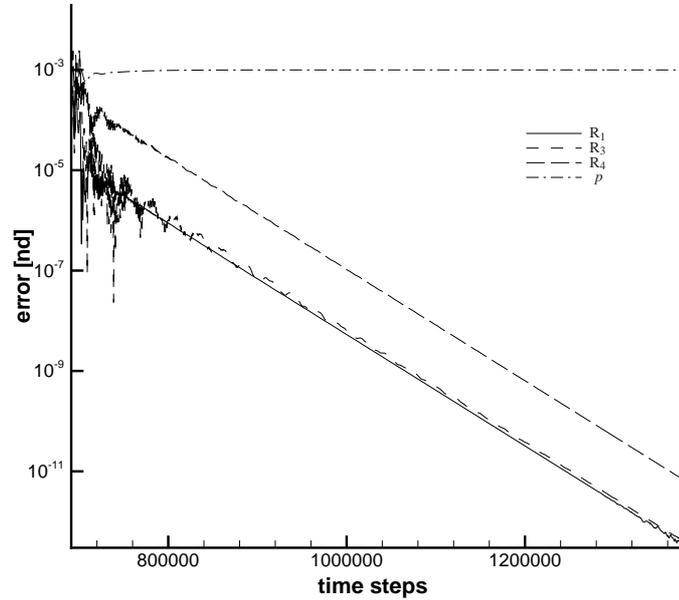


Figure 3.56: Cascade of aerofoils without inflow gust. Results from simulation 2 on grid 1 in Table 3.28. Non-dimensional difference between the target flow state and the predicted flow state on outflow at non-dimensional coordinates (1.5,0.394). The number of time steps includes those from simulation number 1. 1 every 500 time steps is shown, for clarity.

Table 3.30: Fourth test case, results on the outflow plane at the end of the simulation number 2 in Table 3.28. The values are averaged over the boundary and, except for the angle  $\alpha$ , are in non-dimensional form.

Author	Envia	Ghillani		Coupland	Bin
		grid 1	grid 2		
<b>Mach</b>	0.34704	0.37122	0.37337	na	0.3497
<b>Static <math>p</math></b>	0.92000	0.91878	0.91793	na	0.92
<b>Static <math>T</math></b>	0.97648	0.97465	0.97400	na	na
<b><math>\alpha</math> [°]</b>	-1.71	-2.65	-1.78	-1.6	-0.82

### 3. TEST CASE APPLICATIONS

---

acteristics of a compressor stator blade. At the leading edge, at  $x/c = -0.5$ , there is a sharp stagnation point followed by a rapid flow acceleration on both the suction side and the pressure side of the aerofoil. This rapid flow acceleration causes spurious numerical spatial oscillations in the predicted static pressure over the range  $-0.5 \leq x/c \leq -0.48$  on grid 1. These spatial oscillations are substantially suppressed in the results from grid 2, by local node clustering. Therefore, grid number 2 seems to give a better description of the zone near the edges due to its higher node density while, away from them, the results from grid number 1 appear to have less changes in curvature, which may indicate a better discrete representation of the aerofoil profile. The static pressure distribution reaches a minimum at  $x/c = -0.34$  on the suction side and then recovers towards the stagnation pressure value at the aerofoil trailing edge. This pressure recovery is well captured on both grids 1 and 2, with the grid 1 results displaying localised spatial oscillations on approach to the trailing edge at  $x/c = +0.5$ .

Figure 3.58 shows the distributions of the non-dimensional static pressure and of the Mach number at the end of simulation 2 on grid 1 in Table 3.28. The data is replicated over two cascade pitches to improve their readability. The results show a reduction in Mach number and an increment in the static pressure that is consistent with the diffuser role of a compressor cascade. A numerical problem seems to arise downstream of the trailing edge of the aerofoils where oscillations in the vertical direction can be detected. This numerical artifact is related to the inviscid high-order finite-difference nature of the scheme. The absence of the viscous forces leads to a duality, or pitchwise discontinuity in the field of the velocity component in the  $x$  direction. This causes problems to inviscid solvers, in which the Kutta condition is not imposed by other means. The flow state at the aerofoil trailing edge is defined by just one node and cannot describe the dual nature of the  $x$  velocity component in that point.

### 3. TEST CASE APPLICATIONS

---

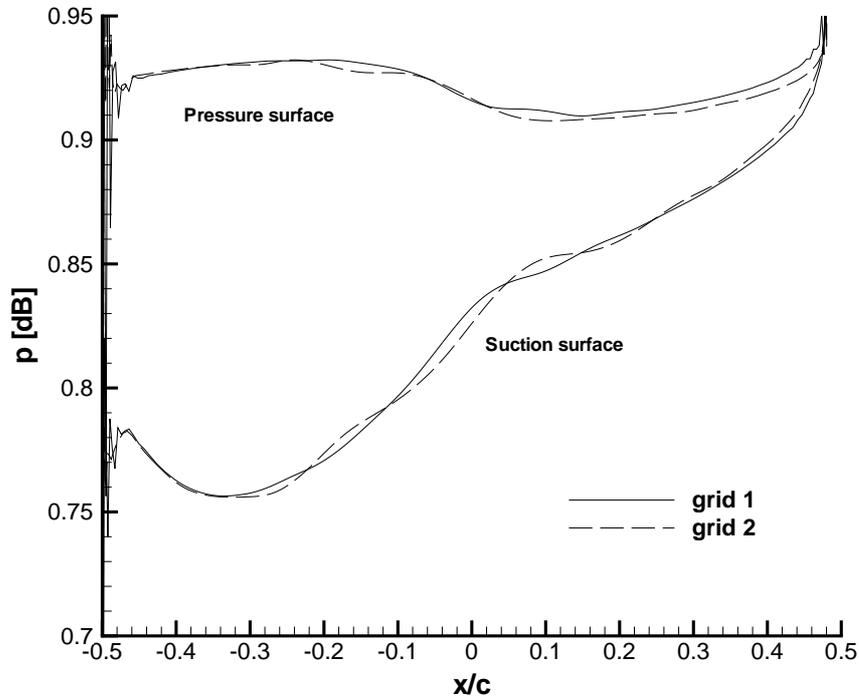


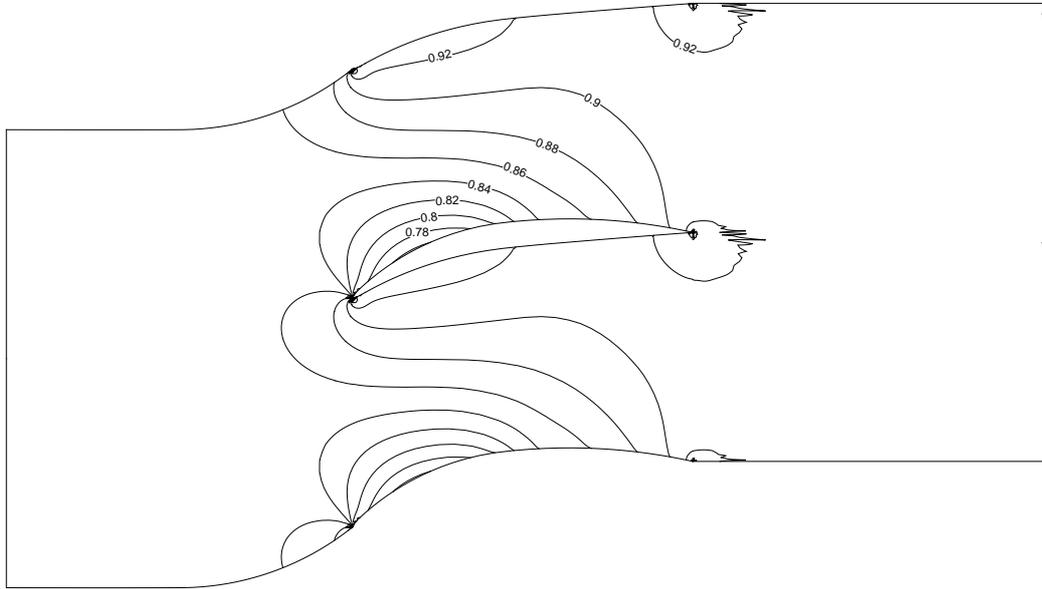
Figure 3.57: Fourth test case, non-dimensional static pressure distribution over the aerofoil surface. Results from simulations with grid 1 and 2.

The issue is not restricted to the edge only as the pitchwise discontinuity in the velocity affects a large part of the field in the wake of the aerofoil. Furthermore, the long stencils, typical of high-order solvers, applied across these discontinuities, generate oscillations that propagate in the pitchwise direction.

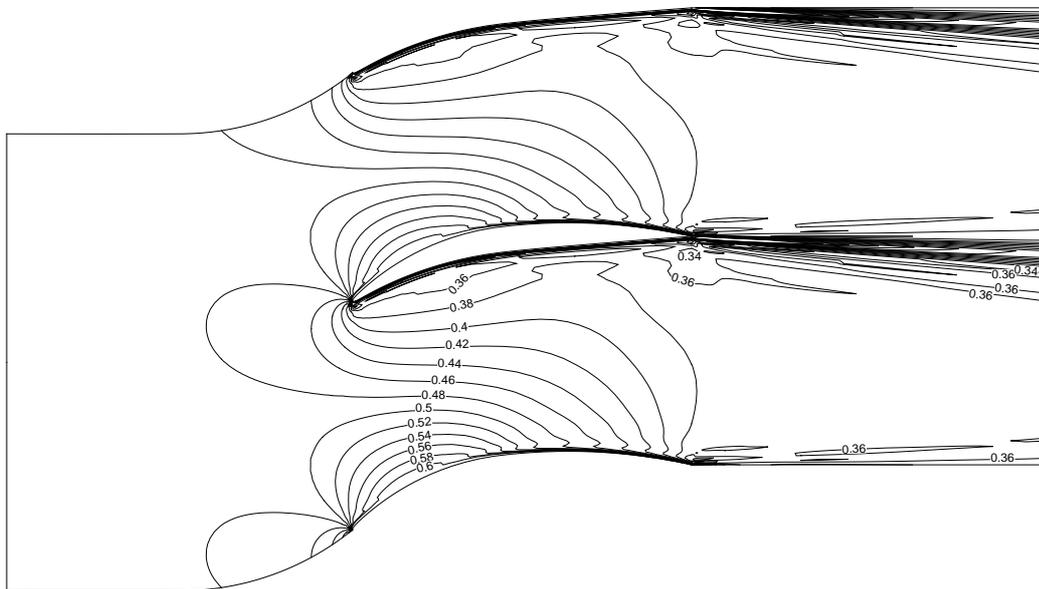
The problem mainly affects the velocity field but is also clearly visible in the pressure field. The results obtained from grid number 2 are shown in Figure 3.59. These display the same flow features as the predictions from grid 1 in Figure 3.58. The mesh clustering at the aerofoil trailing edge produces a slightly more localised static pressure peak. The shear layer downstream of the aerofoil trailing edge appears to be somewhat narrower.

### 3. TEST CASE APPLICATIONS

---



(a) Non-dimensional static pressure distribution. Constant contour spacing  $\Delta p = 0.02$ .

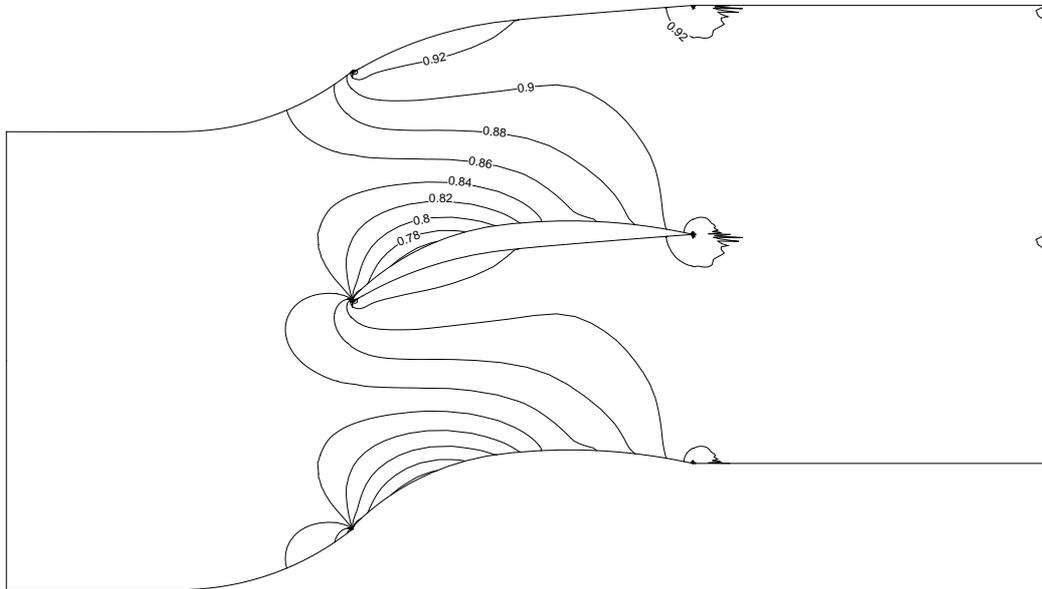


(b) Mach number distribution. Constant contour spacing  $M = 0.02$ .

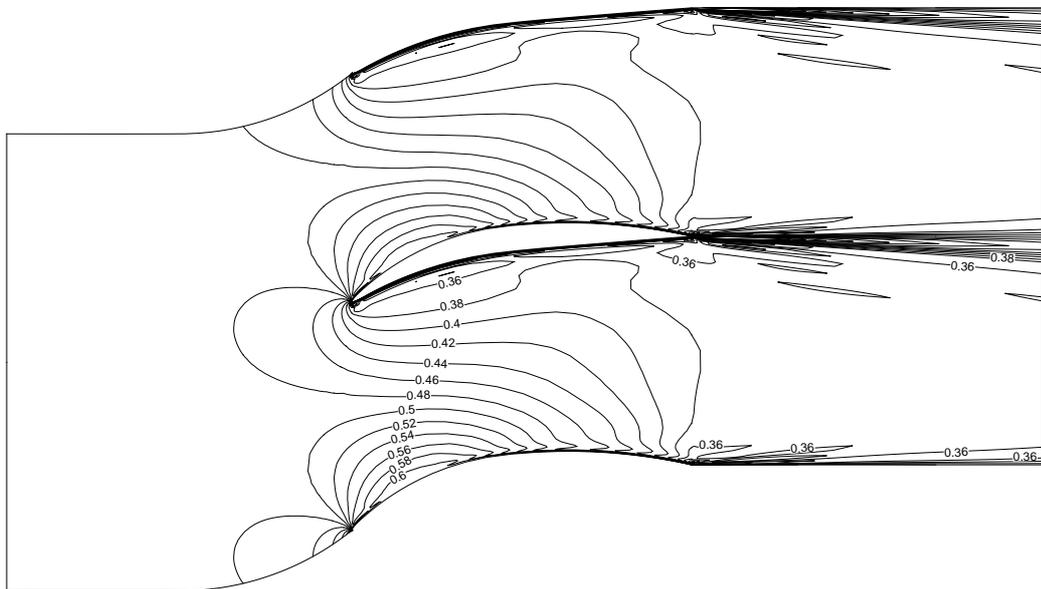
Figure 3.58: Fourth test case, steady-state simulation on grid 1. Pressure and Mach number distribution are shown over 2 vanes.

### 3. TEST CASE APPLICATIONS

---



(a) Non-dimensional static pressure distribution. Constant contour spacing  $\Delta p = 0.02$ .



(b) Mach number distribution. Constant contour spacing  $M = 0.02$ .

Figure 3.59: Fourth test case, steady-state simulation on grid 2. Pressure and Mach number distribution are shown over 2 vanes.

### 3. TEST CASE APPLICATIONS

---

#### 3.4.3.2 Unsteady flow through an aerofoil cascade with inflow gust

The time-dependent simulations with inflow gust are run with the same code as the corresponding steady flow simulations without inflow gust, using their final solution as initial condition and applying implicit filters with the characteristics shown in Table 3.26. The gust is introduced as a velocity perturbation from the inlet and the convergence of the time-dependent flow is determined as the statistical stationarity of the flow field, as required by the problem definition in Section 3.2.1. Similarly to the second test case in Section 3.2, the gust, defined by Equations (3.31), is introduced using an additional term in the definition of the Giles subsonic inflow boundary condition. The mathematical details regarding the implementation of this closure can be found in Section 2.5.2.

In the second test case, the gust is function of the two spatial coordinates and of time and is known at every point of the field, so, if required, the computation can start from a field already initialised with the velocity gust. In contrast, in this case, the gust is only defined analytically on the inlet boundary and, if the unsteady simulation is started without further preparation, the abrupt change in the velocity field near the border tends to generate oscillations that propagate within the field. As a result, the predicted flow field struggles to reach a statistically stationary state. This problem was overcome by initialising the field near the inflow with a velocity distribution compatible with the inflow condition. When the unsteady simulation is started, based on the mathematical definition of the gust and on the values of the steady-state results, an additional component having the amplitude shown in Figure 3.60 is superimposed on the steady-state flow field. This imposed velocity perturbation varies with a sinusoidal law that ensures the continuity of the velocity distribution and of its first derivative. At

### 3. TEST CASE APPLICATIONS

---

each node, based on the definition of the gust, the amplitude of the non-dimensional corrections  $u_0'$  and  $v_0'$  of the velocity components along the  $x$  and the  $y$  axes in the area of the computational field near the inflow boundary is then established with

$$A = \left[ 0.5 + 0.5 \cos \left( \pi \frac{x - x_b}{x_r - x_b} \right) \right] \sum_{n=1}^3 a_n \cos \left\{ n \left[ k_y \left( y - \frac{v}{u} (x - x_b) \right) - \frac{\omega}{u} (x - x_b) \right] \right\}, \quad (3.34a)$$

$$u_0'(x, y) = A \cos \beta, \quad (3.34b)$$

$$v_0'(x, y) = -A \sin \beta, \quad (3.34c)$$

where  $x_b$  is the  $x$  non-dimensional coordinate of the boundary inflow,  $x_r$  is the  $x$  non-dimensional coordinate of the first node where no correction is applied,  $k_y$ ,  $a_n$  and  $\omega$  are as by the problem definition in Section 3.4.1, and  $u$ , and  $v$  are the non-dimensional velocity components of the steady flow field at coordinates  $x, y$ . In case of conservative formulation, the momentum and energy equations are treated in a similar fashion.

For both grid number 1 and number 2, the unsteady runs are performed on a model including 27 vanes and divided into 81 zones. In turn, each zone is the divided vertically into two parts of the same extension, so that 162 processors are required and the total number of nodes is equal to 238464. Table 3.31 summarises the time-integration parameters of the unsteady simulations and their computational cost.

As required by the problem definition, the periodicity in pressure is considered achieved when the spectra of two successive periods differ by less than 1% for each of the input frequencies at the inflow, outflow and on the surface of the aerofoil. This

### 3. TEST CASE APPLICATIONS

---

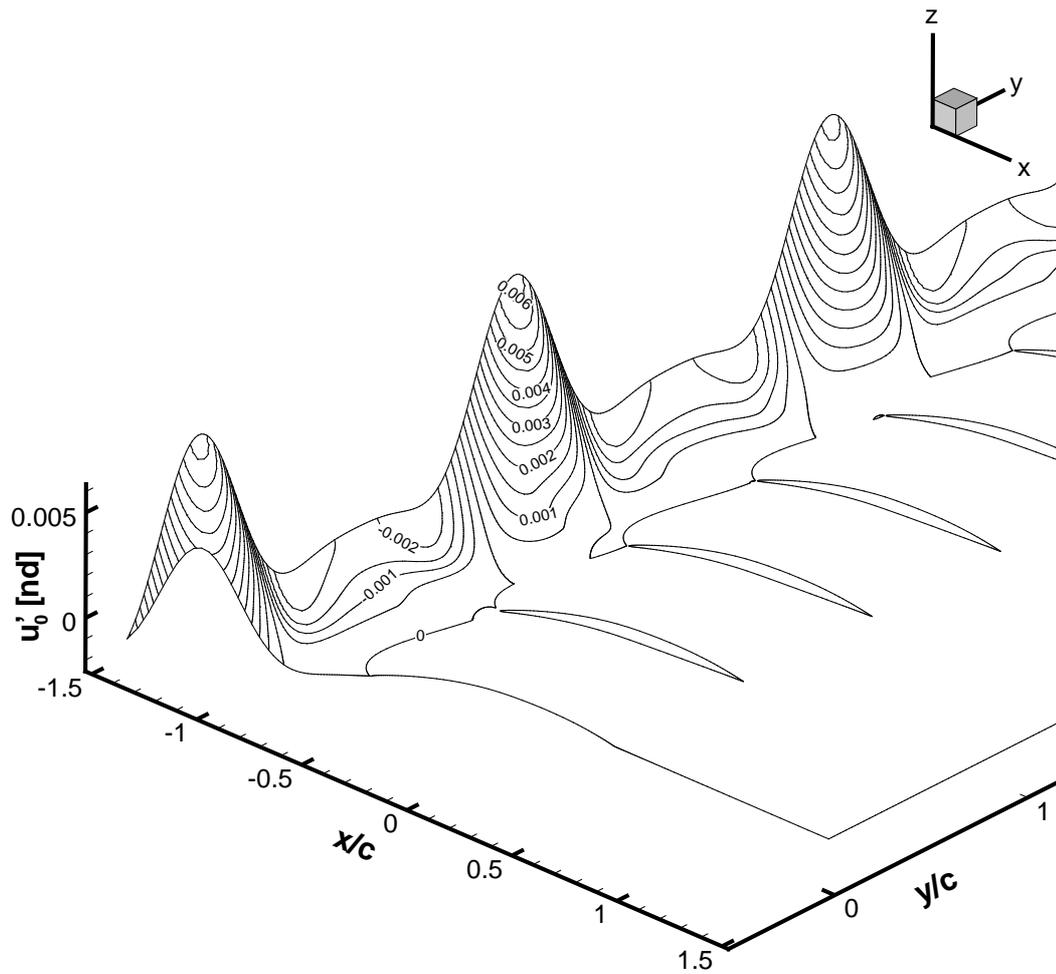


Figure 3.60: Cascade of aerofoils with inflow gust. Initialisation of the streamwise velocity field. A velocity perturbation in the form of an advected gust of increasing amplitude is used. Three harmonics  $n = 1, 2, 3$  are imposed. Contours of non-dimensional  $x$  velocity perturbation  $u_0'$  with constant contour spacing  $\Delta u_0' = 0.0005$ .

### 3. TEST CASE APPLICATIONS

---

Table 3.31: Time integration characteristics of the unsteady simulations performed for the fourth test case.

Grid	Time step [nd]	Steps	Final time [nd]	Walltime [hh:mm:ss]	CPU time [hh:mm:ss]	Total memory [MB]
1	0.000694	188160	130.6	01:31:29	241:47:10	4925
2	0.000115740	1658880	192.0	52:57:58	8515:29:18	7855

was implemented by sampling the pressure field on 12 nodes distributed as shown in Figure 3.52.

Figure 3.61 shows the convergence in the case of simulation 2 on grid number 2 in Table 3.28.

The predicted sound pressure level (in dB) from the unsteady simulations are shown in Figures 3.62 and 3.63. They are compared to the numerical reference solution given by Envia [2004] and by Escribano, Hixon and Bin as reported in Dahl [2004]. The same data are presented in numerical form in Table A.1 of Appendix A.

In Figure 3.64 the amplitude of the perturbations in dB for each dominant mode (see Table 3.23) extracted with a Fourier transform along the inflow and the outflow planes is compared with the results from other research groups. The same data are supplied in numerical form in Table A.2 of Appendix A.

In general, the comparison with the reference numerical solution supplied by Envia [2004] and with the work of the participants of the NASA workshop is satisfactory with the exclusion of the values on the outflow where the radiated noise is remarkably over-estimated. This is probably due to the the oscillations coming from the trailing edge already noticed in the steady-state solution. There could also be a secondary effect due to the reaction of the Giles subsonic outflow condition to the outgoing high-frequency transverse waves. On the surfaces of the reference blade, the results from grid 1 and

### 3. TEST CASE APPLICATIONS

---

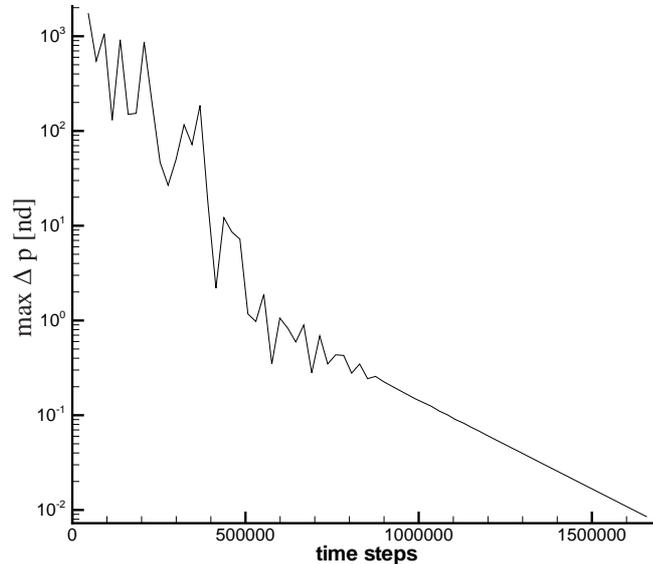


Figure 3.61: Simulation of the aerofoil cascade with inflow gust. Simulation 2 on grid number 2 in Table 3.28. Convergence in time of the non-dimensional pressure difference between two successive periods is shown. The value is the maximum among the 12 monitoring points of Figure 3.52.

2 do not show (apart from the third harmonic) relevant differences. In contrast, on both inflow and outflow planes, some results are not homogeneous and the difference can not easily be explained. They are surely related to the different node density of the various zones of the two grids. The rarefaction of the nodes near the inflow and outflow of grid 2 seems to adversely influence both the behaviour of the Giles condition and the accuracy of the vortical gust introduced.

The extraction of the three harmonics of the input gust from the pressure field confirms the theoretical predictions of the Tyler-Sofrin rule of Equation (3.32) summarised in Table 3.23. Although between the blades (see Figure 3.72) the largest pressure oscillations are those related to the fundamental angular frequency  $\omega$  as amplitude  $a_1$  dominates over  $a_2$  and  $a_3$ , the circumferential modes related to it, and in particular  $m = +11$  and  $m = -16$ , are all evanescent and their amplitudes, decreasing exponen-

### 3. TEST CASE APPLICATIONS

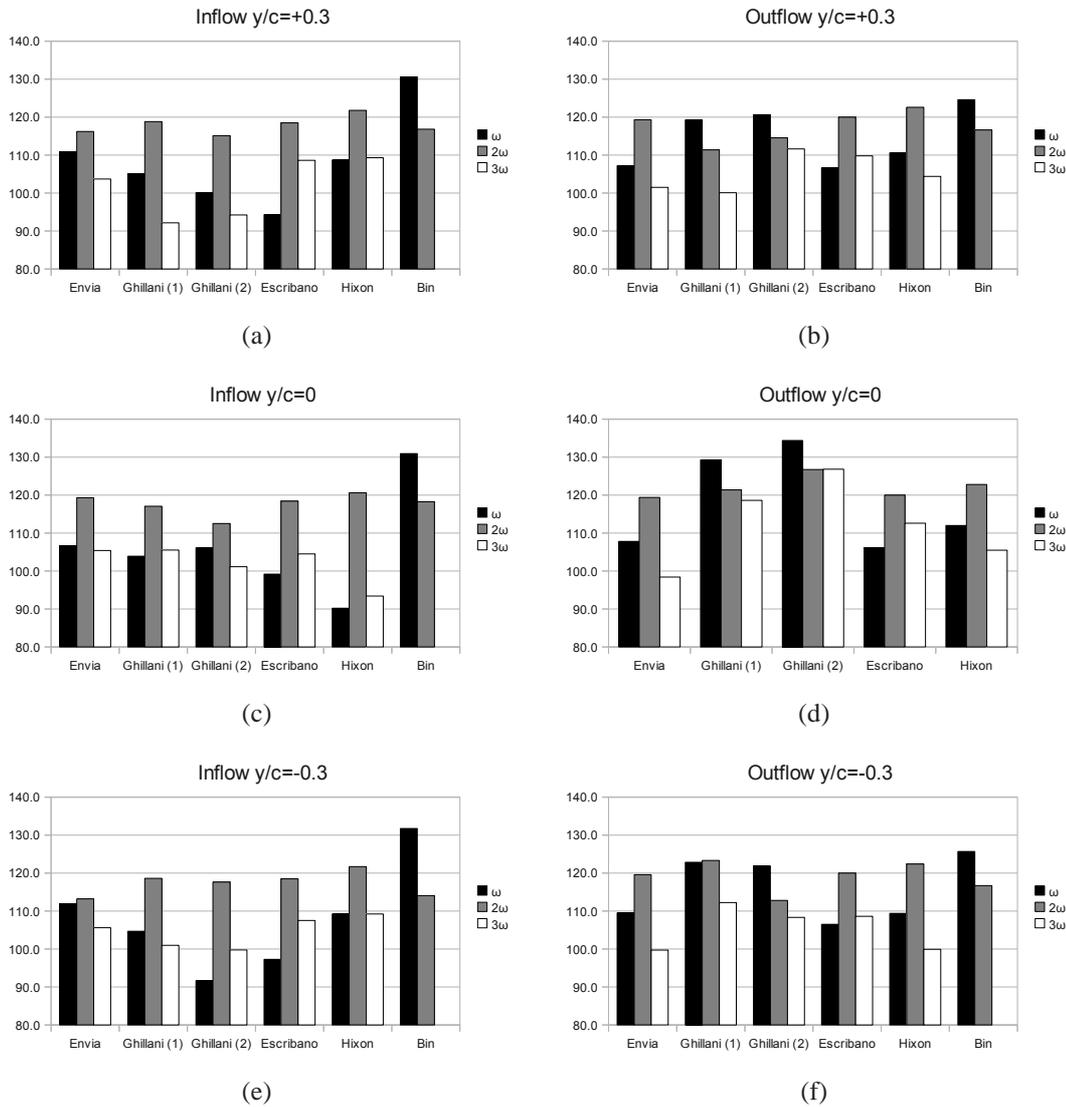


Figure 3.62: Sound pressure level (dB) predicted upstream and downstream of a cascade of aerofoils with inflow gust. Monitoring points (i1, i2, i3) and (o1, o2, o3) shown in Figure 3.52. Predictions are compared with reference values from Envia [2004], Escribano, Hixon and Bin reported in Dahl [2004].

### 3. TEST CASE APPLICATIONS

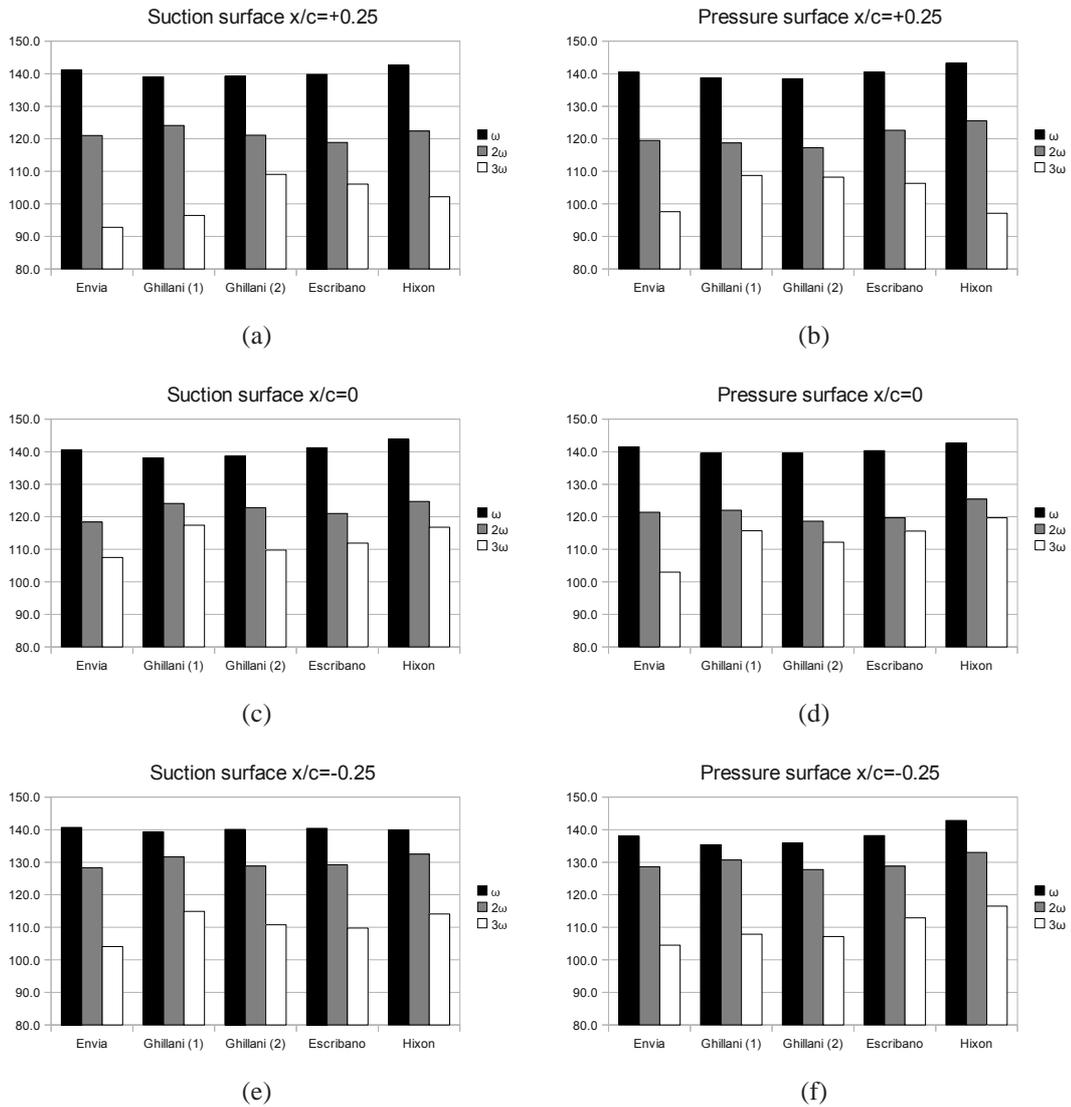


Figure 3.63: Sound pressure level (dB) predicted on the surface of an aerofoil cascade with an inflow gust. Surface monitoring points (a1, a2, a3) and (b1, b2, b3) shown in Figure 3.52. Predictions are compared with reference values from Envia [2004], Escribano, and Hixon reported in Dahl [2004].

### 3. TEST CASE APPLICATIONS

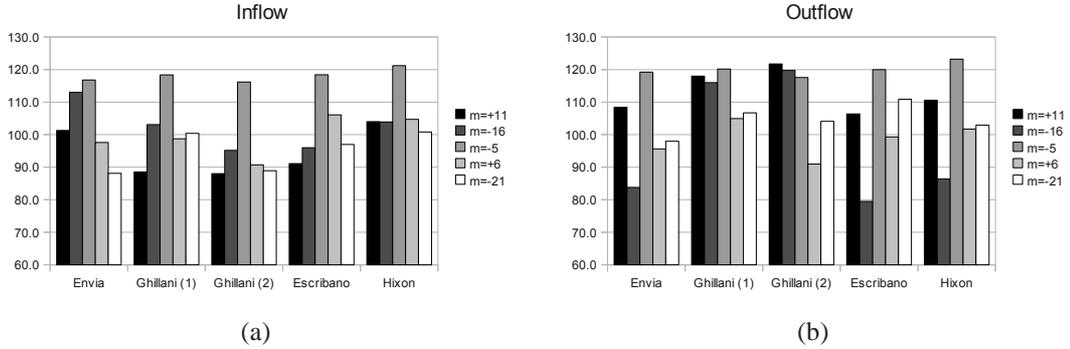


Figure 3.64: Fourth test case, unsteady simulations. The charts show the pressure amplitude of the relevant mode orders along the inflow and the outflow planes. Predictions are compared with reference values from Envia [2004], Escribano, and Hixon reported in Dahl [2004]. Data in dB.

tially from the edges of the blades, as shown in Figures 3.65 and 3.68 and details (a) and (b) of Figures 3.73 and 3.74, are exceeded by the oscillations generated by the second and the third harmonic. The second harmonic (reduced angular frequency  $2\omega$ ) has a clear propagating mode  $m = -5$  as shown in Figures 3.66 and 3.69 and details (c) and (d) of Figures 3.73 and 3.74, while all the others are evanescent. As expected, the amplitude of the wave propagating in the aft direction is slightly larger than that travelling upstream. Similarly, the third harmonic has two propagating modes  $m = -21$  and  $m = +6$  as shown in Figures 3.67 and 3.70 and details (e) and (f) of Figures 3.73 and 3.74. It has to be noted that in the  $x$  direction, the wavelength of the third harmonic is approximately  $0.5c$ . This means that in the areas where the grid is coarser, like in the proximity of the boundaries or near the centre of the zone between the blades, the number of points per wavelength is not far from the limit, conventionally fixed at 4.6 for spatial schemes of this order (see Section 2.2), below which the accuracy of the computation can be considered compromised.

### 3. TEST CASE APPLICATIONS

---

In the zone near the inflow, for all the harmonics the situation is clear: the waves are propagating from the blade keeping their amplitude or decreasing linearly in the logarithmic scale. The only exception to this are waves, fed into the field from the boundary, clearly visible for modes  $m = +11$ ,  $m = +22$  and  $m = +33$ . This can be explained remembering that the incoming gust has a  $\beta$  angle of  $44^\circ$  and, therefore, the gust is not purely 2D. So, along with the velocity variations, the gust is also composed of small incoming pressure and density waves of exactly this circumferential mode (remember that the problem simulates the presence upstream of an 11 blades rotor). From details (a), (c) and (e) of Figures 3.73 and 3.74 it is evident that these modes represent an exception and show an irregular behaviour on the inflow. The situation on the outflow is rather complicated due to the oscillations propagating from the trailing edge and the unsatisfactory performance by the Giles subsonic outflow boundary condition. In fact, the boundary seems to reflect waves (this is particularly evident for angular frequency  $2\omega$ ) into the field probably as a reaction to the outgoing high-frequency transverse oscillations. Comparing Figures 3.73 and 3.74, the higher quality of the results obtained from grid 1 is apparent and the additional computational power required to run the simulations on grid 2, due to the smaller applicable time step, is not justifiable.

An increase in the quality of the results could surely be obtained from a more refined mesh and from a special treatment of the singularities at the edges of the blades. Other types of non-reflecting boundary conditions for the inflow and the outflow could also be beneficial.

The shown results confirm that high-order solver can successfully perform analyses on large realistic problems of practical interest involving highly irregular grids.

### 3. TEST CASE APPLICATIONS

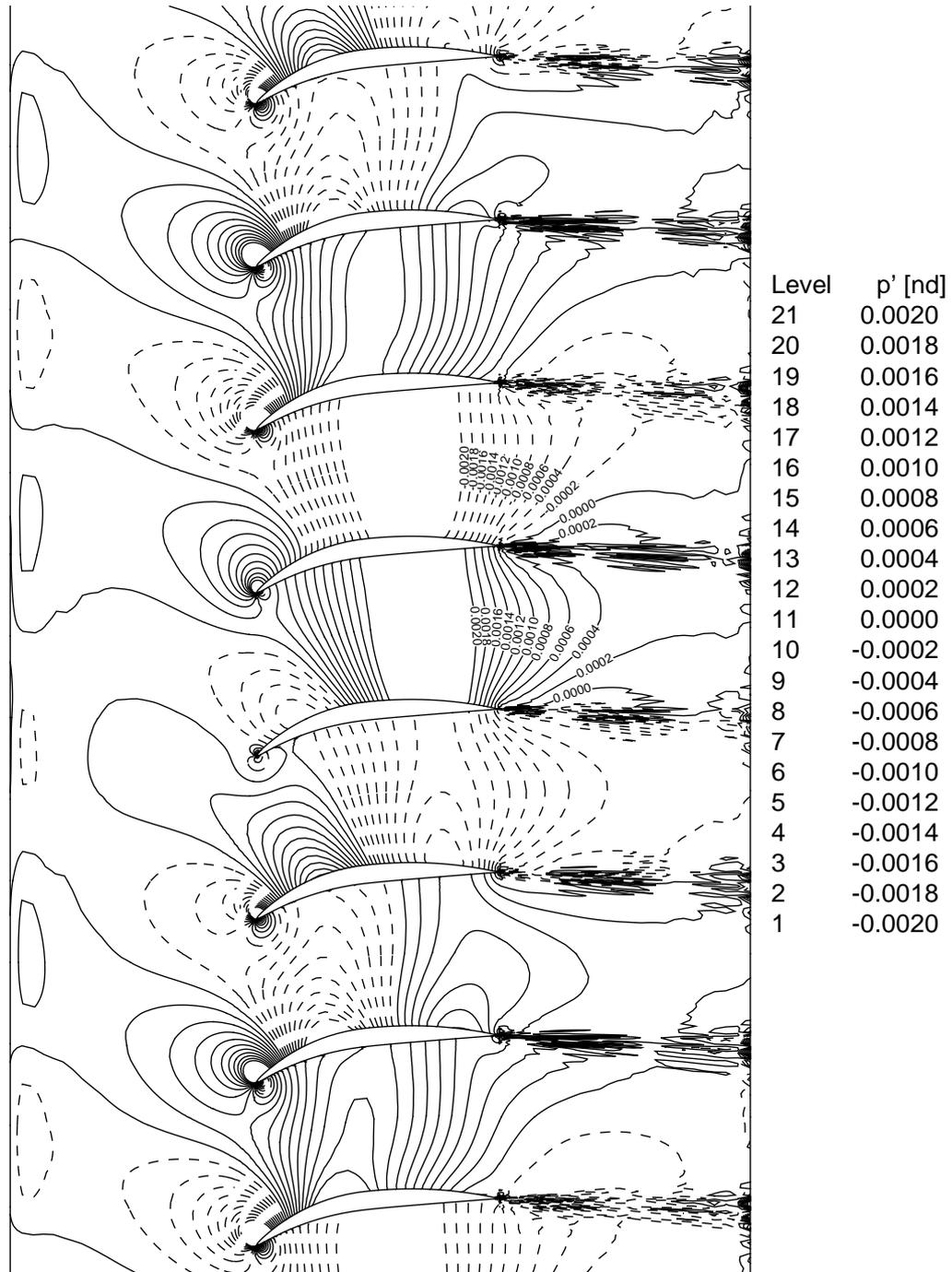


Figure 3.65: Fourth test case, detail of the unsteady simulation on grid 1. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

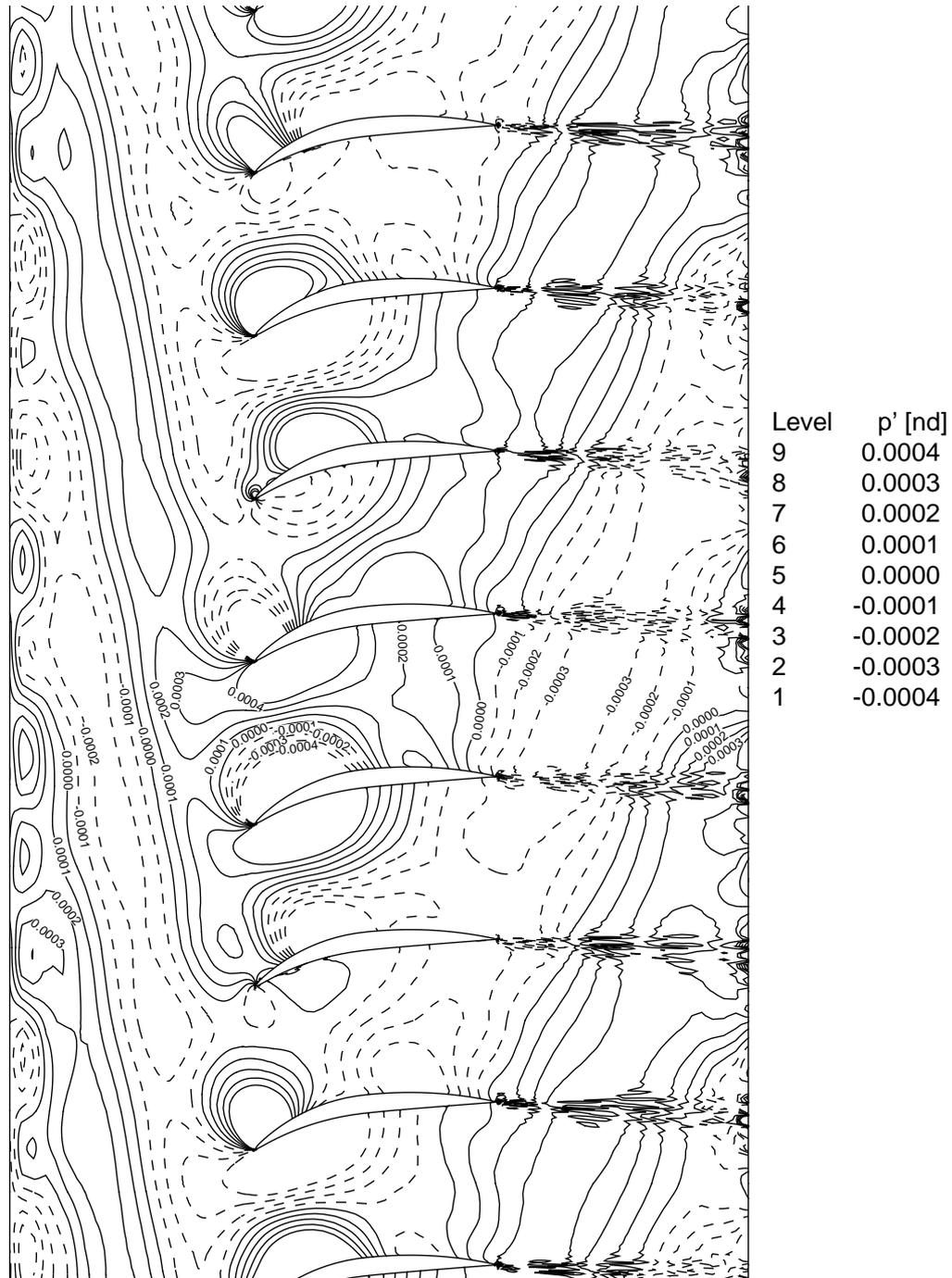


Figure 3.66: Fourth test case, detail of the unsteady simulation on grid 1. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $2\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

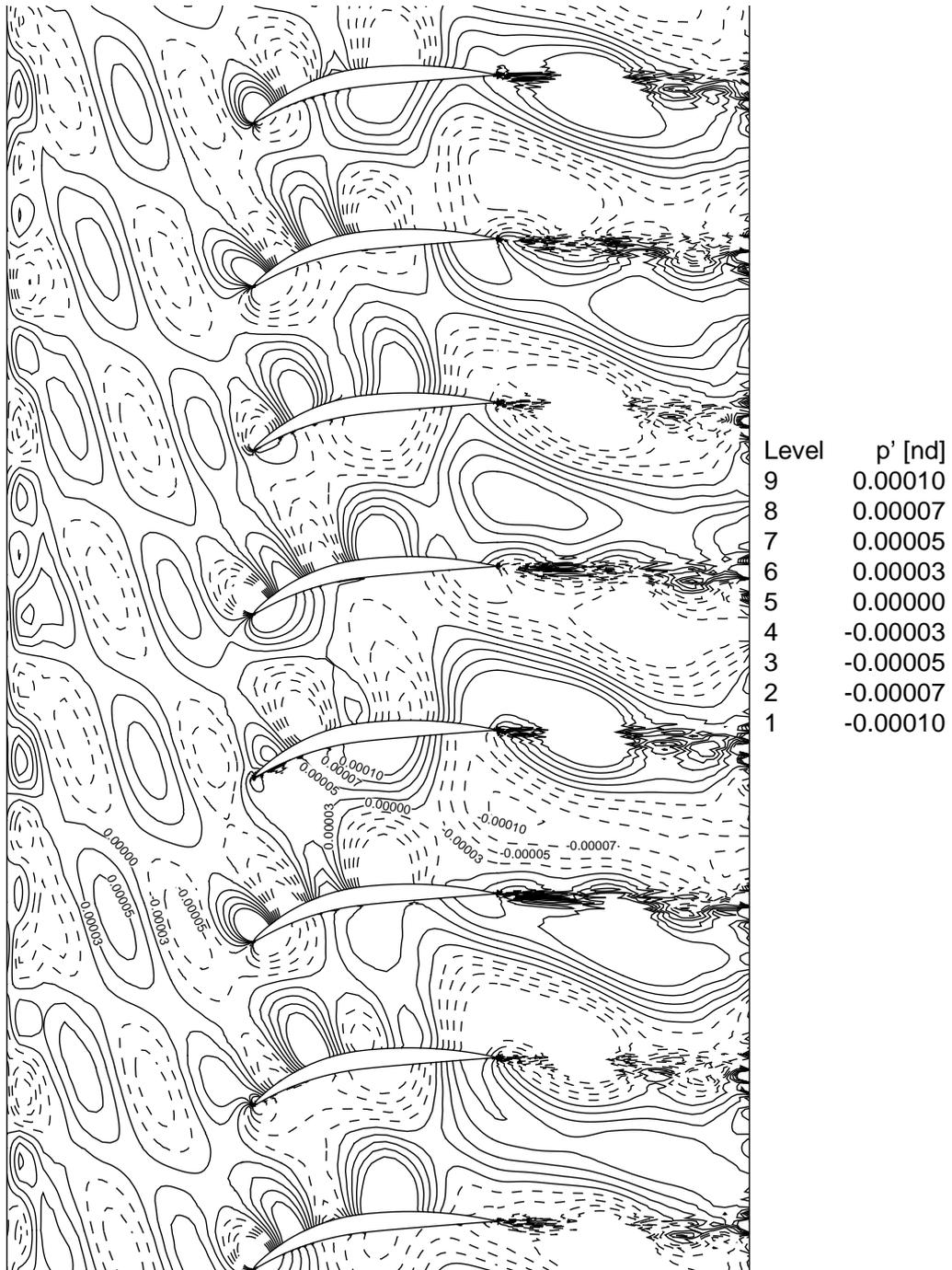


Figure 3.67: Fourth test case, detail of the unsteady simulation on grid 1. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $3\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

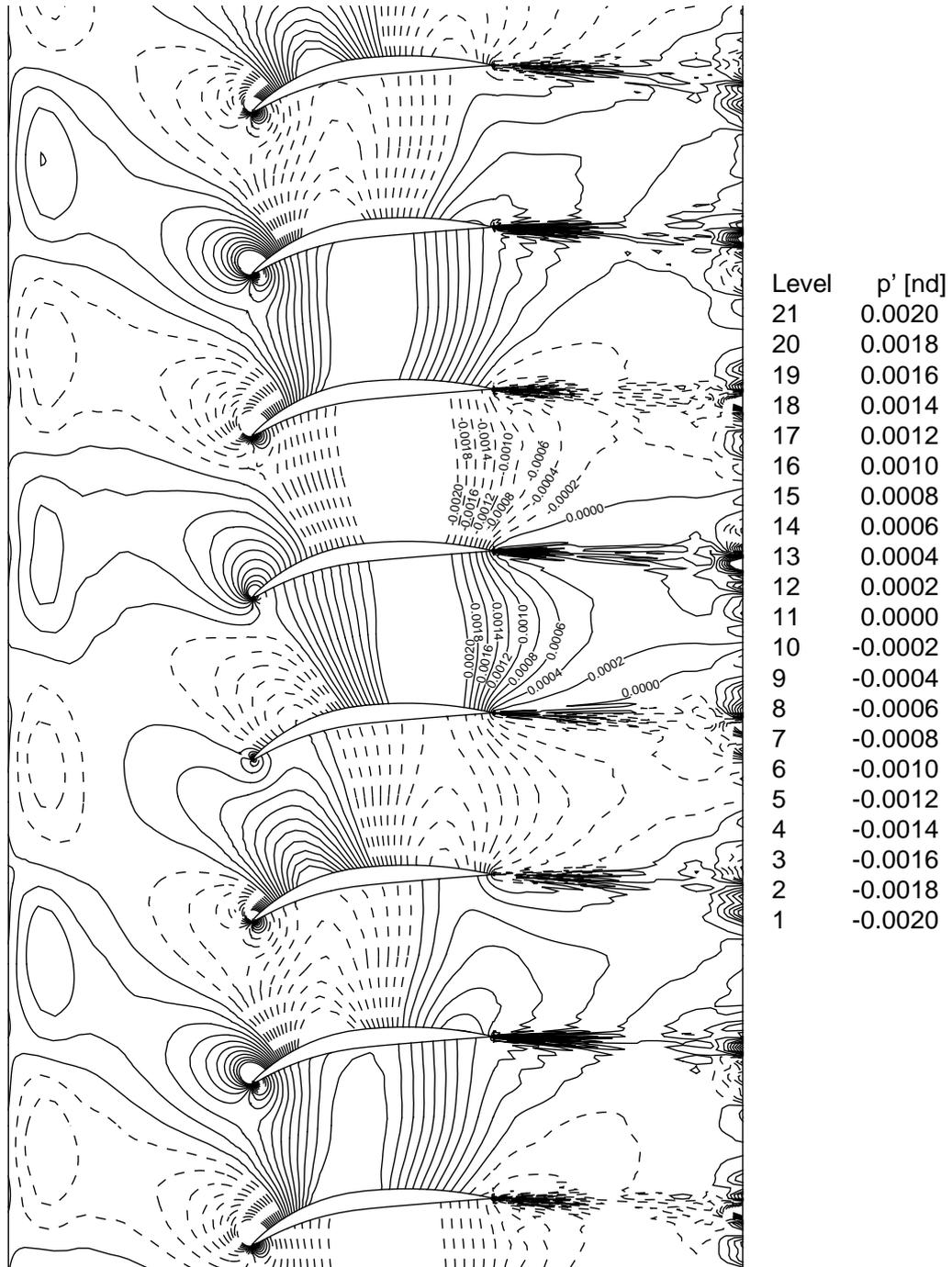


Figure 3.68: Fourth test case, detail of the unsteady simulation on grid 2. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

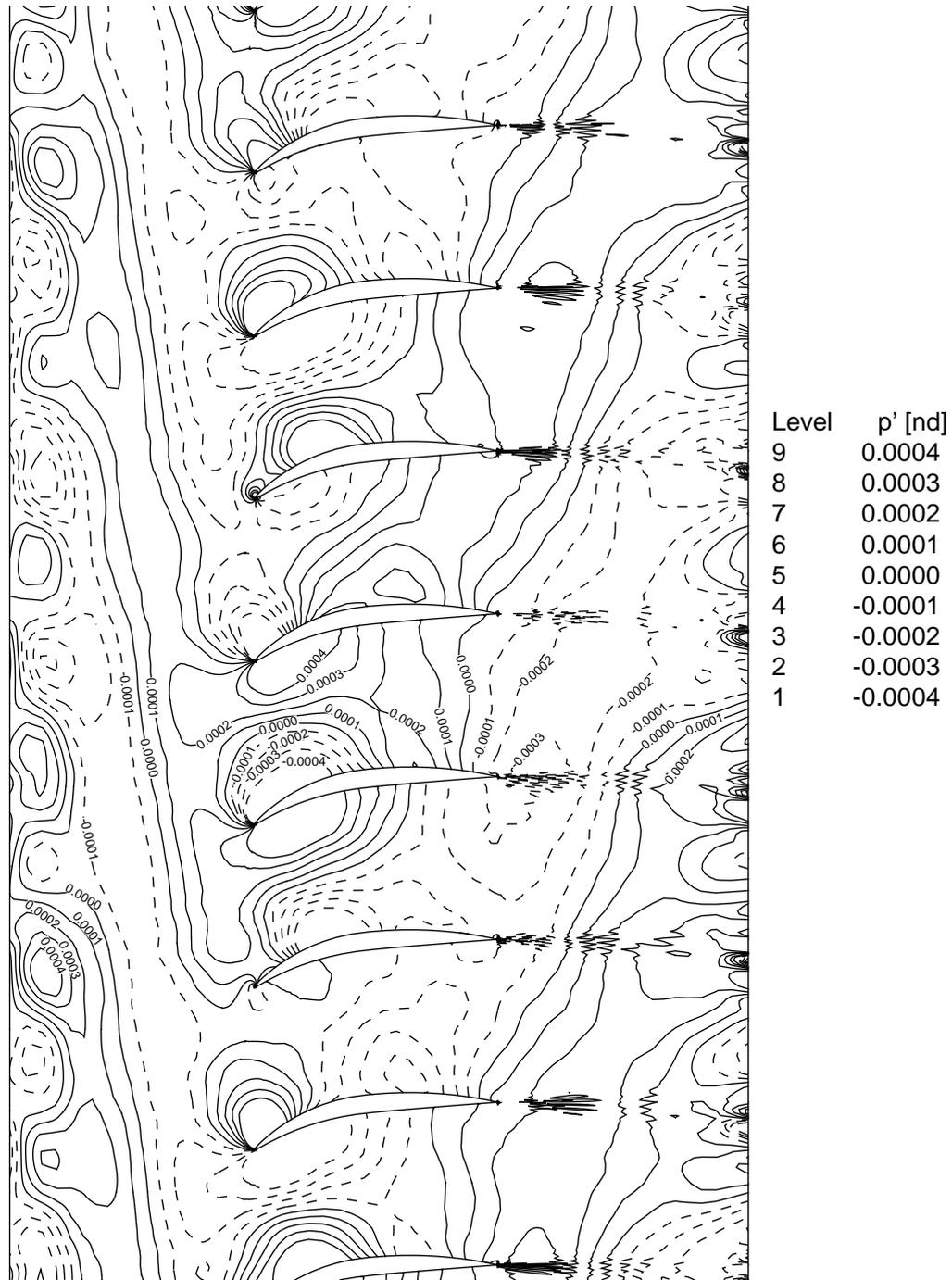


Figure 3.69: Fourth test case, detail of the unsteady simulation on grid 2. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $2\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---

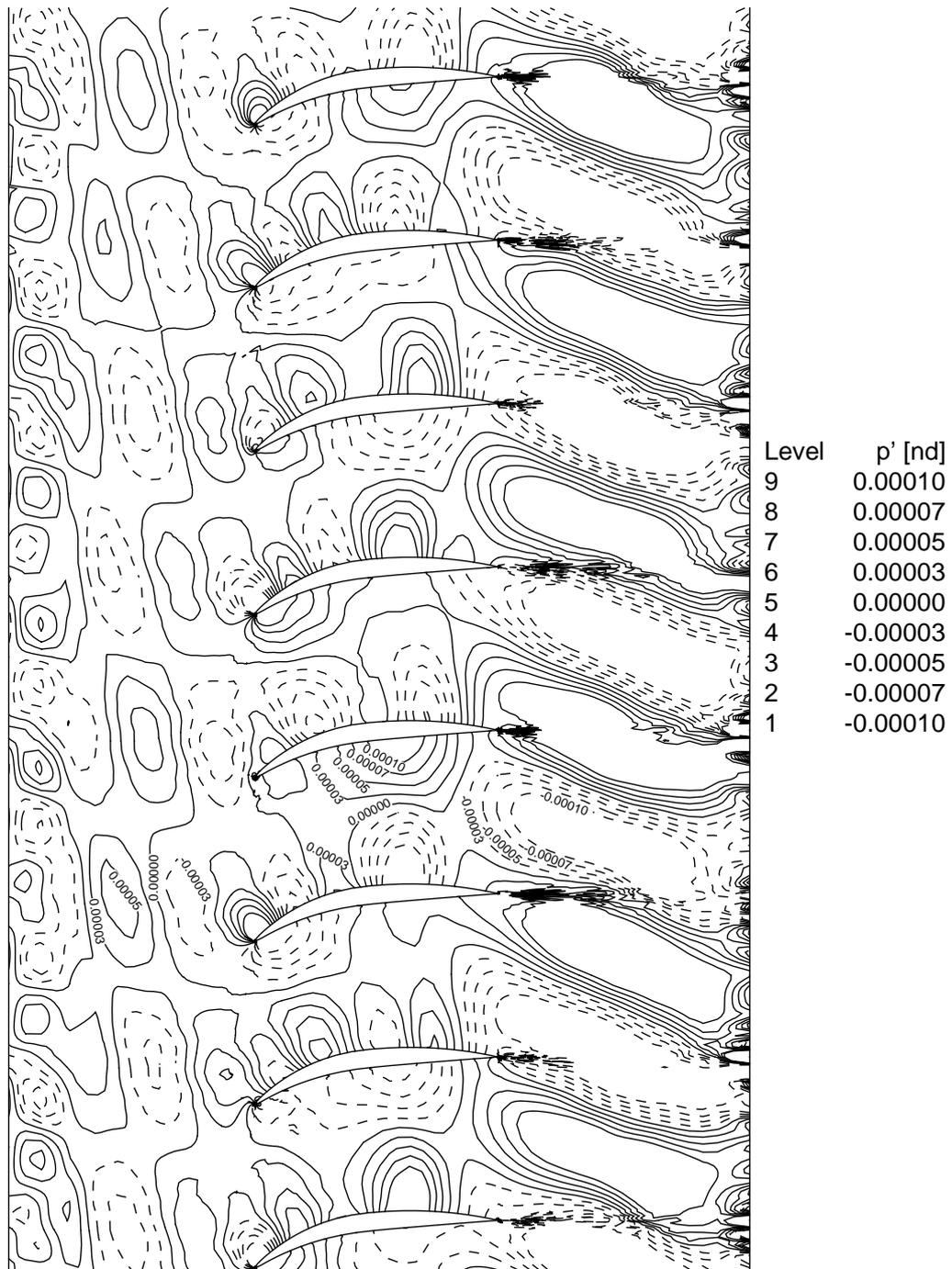
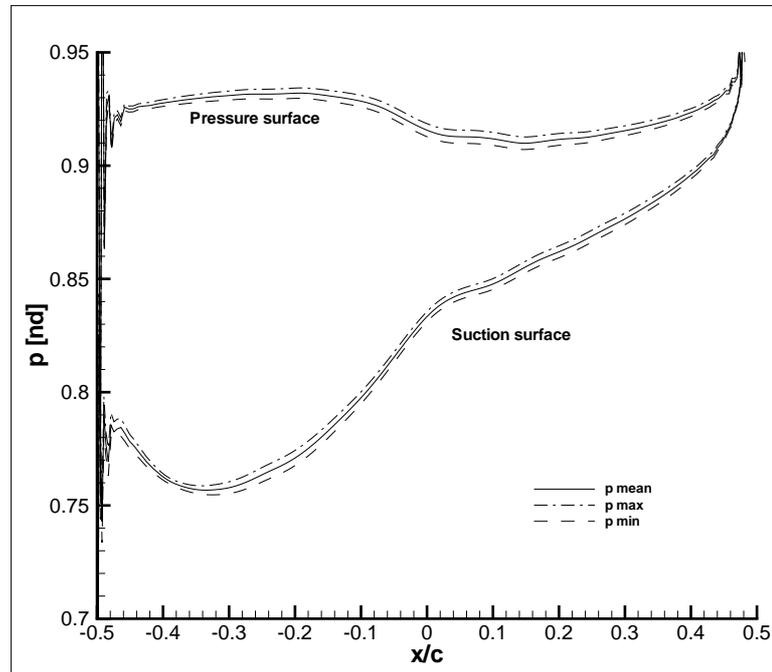


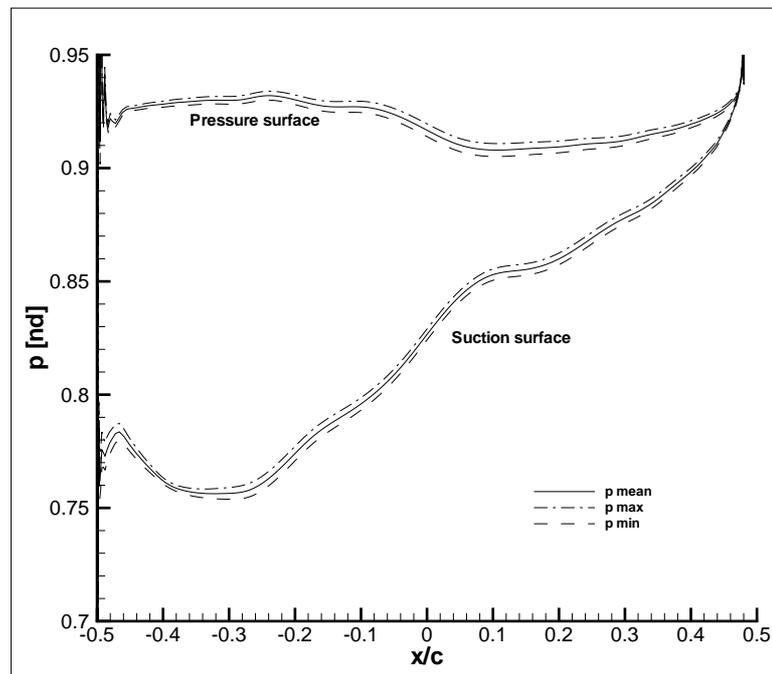
Figure 3.70: Fourth test case, detail of the unsteady simulation on grid 2. The non-dimensional pressure amplitude of the oscillations related to angular frequency  $3\omega$  is shown. Negative contours are shown by dashed lines.

### 3. TEST CASE APPLICATIONS

---



(a) Grid 1.

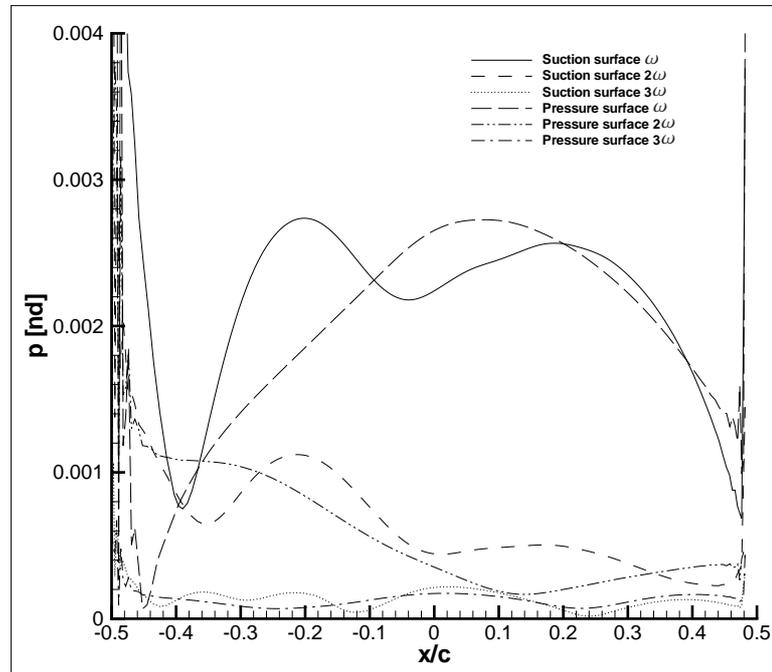


(b) Grid 2.

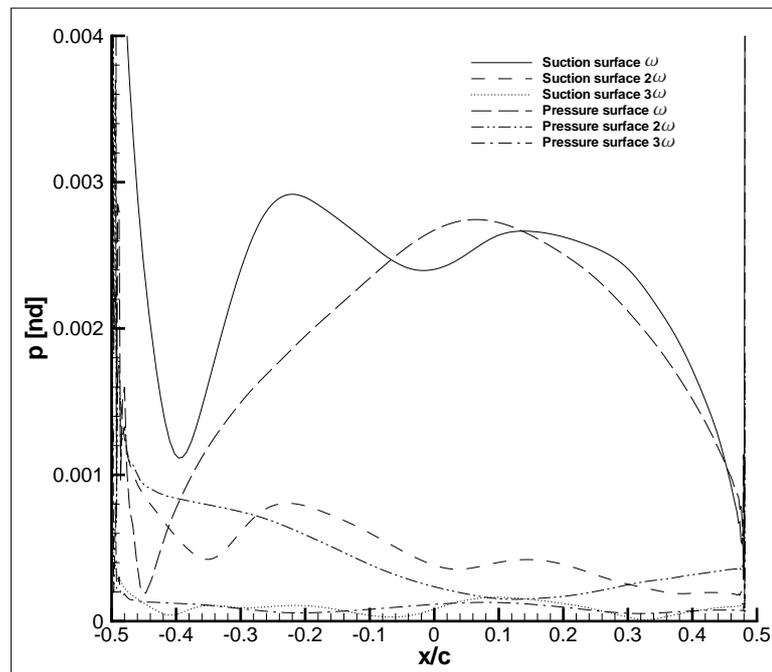
Figure 3.71: Fourth test case, unsteady simulations. The non-dimensional pressure along the aerofoil is shown.

### 3. TEST CASE APPLICATIONS

---



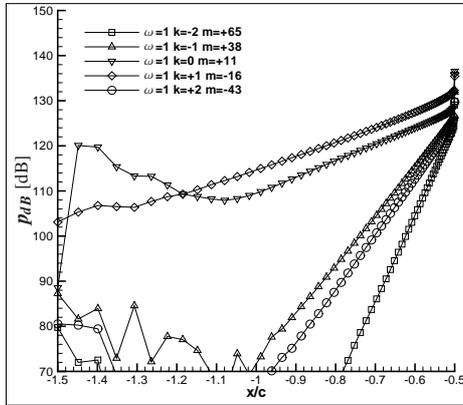
(a) Grid 1.



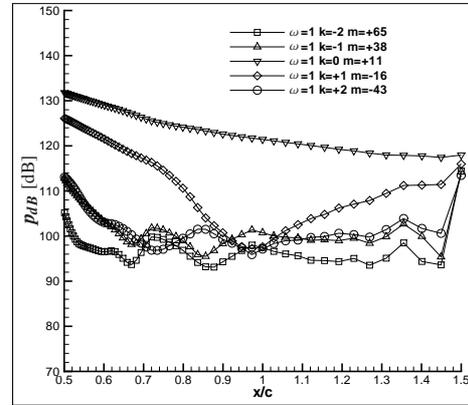
(b) Grid 2.

Figure 3.72: Fourth test case, unsteady simulations. The amplitude of non-dimensional pressure along the aerofoil for each of the three harmonics is shown.

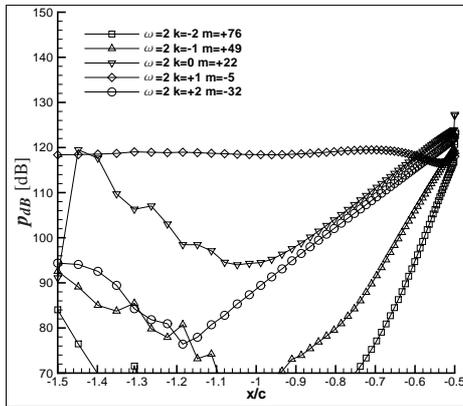
### 3. TEST CASE APPLICATIONS



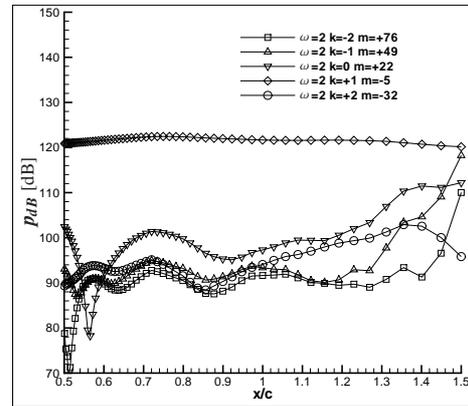
(a) Inflow,  $\omega$ .



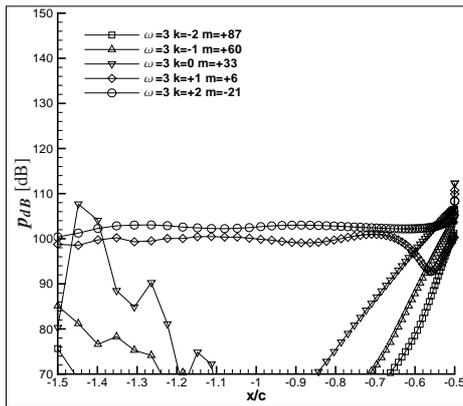
(b) Outflow,  $\omega$ .



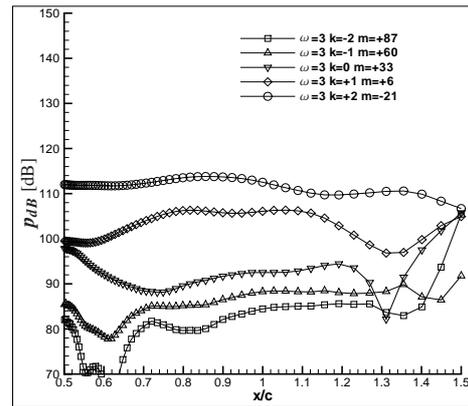
(c) Inflow,  $2\omega$ .



(d) Outflow,  $2\omega$ .



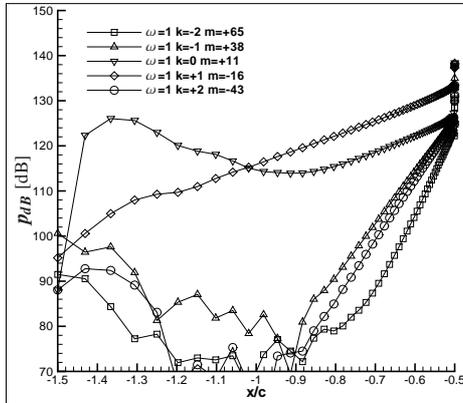
(e) Inflow,  $3\omega$ .



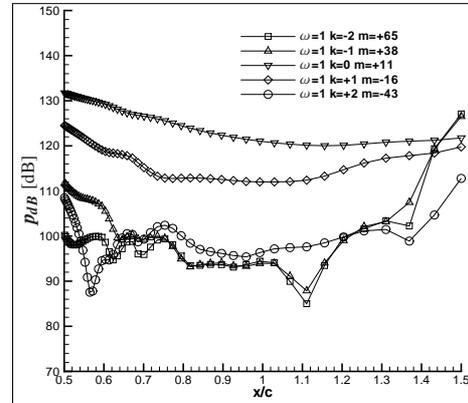
(f) Outflow,  $3\omega$ .

Figure 3.73: Fourth test case, unsteady simulation on grid 1. The acoustic mode amplitude upstream (left-hand side) and downstream (right-hand side) of the aerofoils is shown.

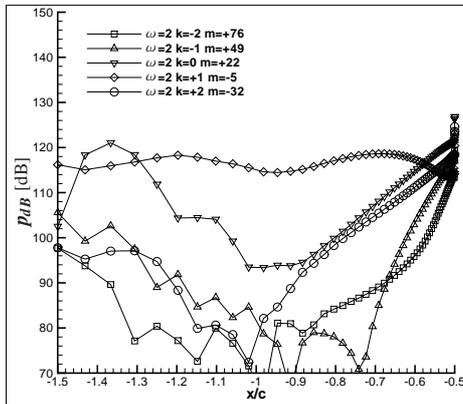
### 3. TEST CASE APPLICATIONS



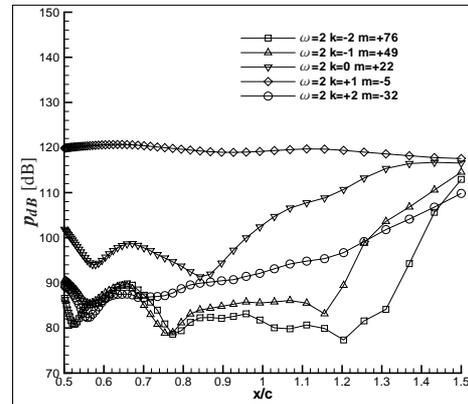
(a) Inflow,  $\omega$ .



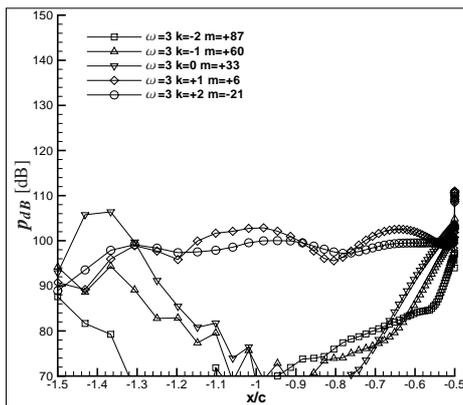
(b) Outflow,  $\omega$ .



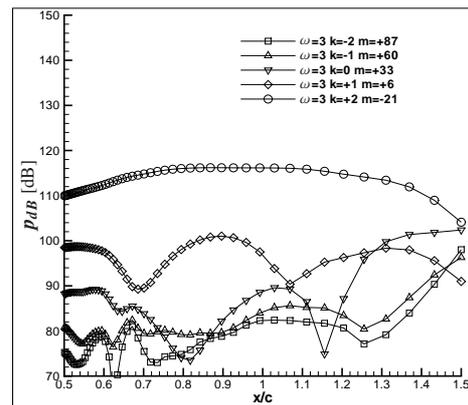
(c) Inflow,  $2\omega$ .



(d) Outflow,  $2\omega$ .



(e) Inflow,  $3\omega$ .



(f) Outflow,  $3\omega$ .

Figure 3.74: Fourth test case, unsteady simulation on grid 2. The acoustic mode amplitude upstream (left-hand side) and downstream (right-hand side) of the aerofoils is shown.

# Chapter 4

## Conclusions

This work developed and tested a three-dimensional high-order prefactored compact finite-difference inviscid flow solver for computational aeroacoustics. By explicit time-integration, time-resolved simulations were obtained of the noise produced by the interaction of unsteady aerodynamic flow with solid boundaries. The time-explicit scheme solves directly the sound generation and near-field propagation in the computational domain. The code developed is based on the two-dimensional in-house code by Spisso [unpublished] that was substantially upgraded by the implementation of numerical methods taken from the literature and by some novel extensions of these methods. The main upgrade was the extension of the scheme to solve the non-linear form of the Euler equations in both primitive and conservative form, in three-dimensions. A generalised characteristic-based framework for the boundary conditions allows to model problems involving curvilinear geometries of complex shape.

The range of non-reflecting boundary closures originally implemented by Spisso [unpublished] was extended through the addition of the Giles subsonic boundary conditions, their three-dimensional extension, and the development of a modified type of

## 4. CONCLUSIONS

---

buffer zone with a target flow state that is both space- and time-dependent. In addition, a symmetry boundary condition was devised for orthonormal Cartesian boundaries that retains the same accuracy of the internal spatial discretisation scheme.

The introduction of a high-order implicit compact filtering method locally controlled by a tunable coefficient and the use of different near boundary filters, both high- and low-order, one-sided and centred, allows to suppress the numerical instabilities typically associated to the use of high-order centred schemes in the presence of geometrical discontinuities or curvilinear grids.

The adoption of message passing interface (MPI) and domain-decomposition allows to overcome the limitations related to the amount of memory available for single-processor computations, thus extending the applicability of the solver to problems involving models of larger sizes. For this purpose, the data decomposition parameters allow to subdivide the zones of the structured grids in every spatial direction giving the user full control over the size and shape of the data assigned to each process.

In order to minimise the amount of data exchange among the processes and to make it independent from the stencil size used in the spatial differentiation scheme, a non-conventional inter-block boundary technique was used. The centred explicit stencils used for the computation of the flow derivatives on the borders were split into two parts. This allows to perform the communication between two processes by the exchange of only two values for each boundary node. This method is currently limited to problems that require prefactored filters with constant tunable coefficient.

The time-integration of the numerical scheme was upgraded by the addition of three multi-step explicit methods. These allow to tailor the trade-off between numerical stability and computational accuracy depending on the nature of the problem to be solved.

## 4. CONCLUSIONS

---

The new three-dimensional scheme was tested on four problems of increasing complexity. The solution of each test case aims to demonstrate the effectiveness of some of the implemented numerical techniques.

The first case, involving a still cylinder in an oscillating field, tests the implementation of some of the generalised characteristic-based boundary closures and, specifically, the curvilinear impermeable wall condition, which includes transverse velocity components. The second test case, which is a two-dimensional turbomachinery rotor-stator interaction problem, demonstrates the effectiveness of the non-reflecting inflow and outflow boundary closures on regular homogeneous grids, by giving results with an accuracy comparable to that of the best results available in the literature for this test case. The third and fourth test cases are higher fidelity models of the same rotor-stator interaction problem. The third test case is a three-dimensional geometry that tested the three-dimensional extension of the numerical scheme on a stretched mesh. The fourth test case modelled an aerofoil cascade by a highly distorted mesh and by solving the Euler equations in non-linear form.

The results from the test cases show that the numerical methods employed allow engineering accurate predictions of the rotor-stator turbomachinery interaction problem. In particular, Hixon's spatial compact prefactored scheme in association with the explicit time-integration provided by the classical Runge-Kutta or by its optimised version by Berland, have shown to be suitable for this class of aeroacoustical problems. For the two-dimensional models, the Giles non-reflecting boundary conditions, both at the inflow and at the outflow, as seen from the literature, confirm to be a good choice although the level of reflectivity of the outflow could affect the quality of the results in the area nearby and its adoption should be considered depending on the problem to be solved. For the three-dimensional problems, the modified buffer condition developed

#### 4. CONCLUSIONS

---

in this work has shown to be the only reliable choice as the tested alternatives have evidenced a long-time instability. The inviscid wall boundary by Kim and Lee has demonstrated to be both accurate and reliable in all the tests performed. In order to overcome the disturbances that originate from the geometrical discontinuities, typical of the high-order centred finite-difference schemes, the addition of a special treatment, as the one illustrated for test case two, has to be taken into consideration as it can have a beneficial impact on the quality of the numerical predictions.

The scalability of conventional high-order numerical schemes has been limited by the communication overhead between processes that increased with increasing order of the scheme. The present work indicates that by splitting the inter-block boundary spatial derivatives, this communication overhead can be reduced. This, in parallel with on-going development work in the USA, has opened the high-order finite-difference compact schemes for use on high-performance computers with  $> 10^3$  processors, for modelling flows of industrial interest.

The suite of governing equations, non-reflecting boundary conditions, variable-order filters, and explicit time-integration options that were added to the numerical method in this work have created a powerful tool for modelling a range of unsteady flows in which the aerodynamic sound generation by interaction with solid boundaries is a key performance parameter for specific engineering applications.

# Chapter 5

## Future work

The results of Chapter 3 highlighted opportunities for improving the numerical scheme, to enhance the accuracy of the solution as well as to extend the range of problems that can be addressed.

The results of the second test case reported in Section 3.2.3 indicate that special treatment of the geometrical singularities can increase the level of accuracy. More optimisation of the numerical method is required to further decrease the spurious oscillations generated at the plate edges and avoid the adverse influence on the error norm of the predictions. This could be aided by the development of an algorithm to automatically check the potential sources of instability from the geometrical shape of the mesh or based on the condition of the flow state.

While the Giles approximate 2D subsonic inflow boundary closure performs well in both the second and the fourth test case, its outflow counterpart, as also noticed in literature [Hixon *et al.*, 2000; Ragab & Salem-Said, 2007], displays a greater reflection of outgoing waves towards the computational domain interior, which is an important source of error in the computation. Alternative outflow boundary formulations could

## 5. FUTURE WORK

---

be tested, such as the one from Hagstrom & Goodrich [2003] that allows for arbitrarily high accuracy levels at the expense of the simplicity of implementation, or the perfectly matched layer (PML) in the implementations by Tam *et al.* [1998] or by Abarbanel *et al.* [1999].

The third test case in Section 3.3.3 shows the limitations of the non-reflecting boundary closures currently available in the code for the solution of 3D curvilinear problems. The Giles boundary extension to 3D regular meshes by Medida [2007], were here adapted to curvilinear meshes. This formulation while allows to model the inflow vortical gust, shows a long-time instability that inevitably leads to the failure of the computation. The implementation of the 3D extension given by Saxer & Giles [1993], although only valid for steady-state solutions, could provide a more effective non-reflecting 3D boundary closure.

Although often necessary to ensure the stability of the computation, the filtering in its explicit or implicit formulation is one of the main sources of reduction in the accuracy of the numerical predictions approaching the walls or the computational domain external boundaries. Improvements in this field through the introduction of new and more stable high-order one-sided stencils, or by the development of new techniques to reduce the adverse influence of the low-order centred filters while retaining the scheme stability, would be appropriate.

The filtering techniques in the compact scheme require the data exchange across inter-block boundaries of the flow state in rinds with thickness that increases with the filter order. The filters do not use the reduced form of communication described in Section 2.6.1. Implementing such reduced communication would aid the solution of 3D problems involving large amounts of data and additional memory which are currently related to the number of rind nodes.

## 5. FUTURE WORK

---

For what concerns the parallel code implementation, improvements are possible. In particular, it would be useful to generalise the connectivity between processes, which is currently limited to having only one neighbour in each spatial direction, in order to improve the flexibility of the solver and its use on more topologically complex grids. This limitation affected the fourth test case, where the complex geometry associated to the aerofoil cascade could have benefited from the use of a more complex connectivity tailored to this problem.

The current parallelisation strategy based on the definition of a hierarchy of processes that are in charge for the data input/output is very flexible and can run on every type of HPC cluster, but it is not computationally efficient. This is particularly evident when large problems are modelled and the reading and writing of data represents a non-negligible part of the computational time. The code was developed with this in mind and its structure remains open to the implementation of parallel input/output, which are supported by the MPI-2 standard level. For the same purpose it would be interesting to investigate the advantages that HDF5 libraries, now the default data storage format at the base of CGNS, could bring to the input/output process.

Other useful code developments would involve testing the viscous fluxes, which were added to the flow solver but have not been used for the computations shown in this work. This would allow to extend the applicability of the scheme to wall boundary layers or to flows where viscosity plays an important role. Furthermore, the introduction of a RANS- or a LES-based turbulence model, following Bogey & Bailly [2006], would enable the modelling of higher Reynolds number viscous flows.

## **Appendix A**

### **Numerical results of the fourth test case**

## A. NUMERICAL RESULTS OF THE FOURTH TEST CASE

Table A.1: Fourth test case, comparison of the sound pressure level  $p$  in dB at 12 monitoring points located at the inflow and outflow planes and on the blade surface shown in Figure 3.52.

Inflow plane  $x/c = -1.5$ , points i1 to i3

	$y/c = -0.3$ (i1)			$y/c = 0.0$ (i2)			$y/c = +0.3$ (i3)		
	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$
<b>Envia</b>	111.9	113.2	105.6	106.6	119.3	105.4	110.9	116.2	103.7
<b>Ghillani (1)</b>	104.7	118.6	101.0	103.9	117.1	105.5	105.1	118.8	92.2
<b>Ghillani (2)</b>	91.7	117.6	99.7	106.1	112.5	101.1	100.1	115.1	94.2
<b>Escribano</b>	97.3	118.5	107.5	99.2	118.4	104.5	94.3	118.5	108.6
<b>Hixon</b>	109.2	121.7	109.2	90.2	120.6	93.4	108.7	121.8	109.3
<b>Bin</b>	131.7	114.1		130.9	118.3		130.6	116.8	

Outflow plane  $x/c = 1.5$ , points o1 to o3

	$y/c = -0.3$ (o1)			$y/c = 0.0$ (o2)			$y/c = +0.3$ (o3)		
	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$
<b>Envia</b>	109.5	119.6	99.7	107.8	119.4	98.4	107.2	119.3	101.5
<b>Ghillani (1)</b>	122.8	123.3	112.3	129.2	121.4	118.6	119.3	111.4	100.1
<b>Ghillani (2)</b>	121.9	112.8	108.3	134.4	126.8	126.8	120.6	114.6	111.6
<b>Escribano</b>	106.5	120.0	108.6	106.1	120.0	112.6	106.6	120.0	109.8
<b>Hixon</b>	109.4	122.4	99.9	112.0	122.8	105.5	110.6	122.6	104.4
<b>Bin</b>	125.7	116.7					124.6	116.6	

Suction surface of the  $y = 0$  aerofoil, points a1 to a3

	$x/c = -0.25$ (a1)			$x/c = 0.0$ (a2)			$x/c = +0.25$ (a3)		
	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$
<b>Envia</b>	140.7	128.3	104.1	140.6	118.4	107.5	141.2	121.0	92.8
<b>Ghillani (1)</b>	139.4	131.6	114.9	138.1	124.1	117.4	139.0	124.1	96.5
<b>Ghillani (2)</b>	140.1	128.8	110.8	138.6	122.8	109.8	139.2	121.1	109.0
<b>Escribano</b>	140.4	129.2	109.8	141.2	121.0	111.9	139.8	118.9	106.1
<b>Hixon</b>	139.9	132.5	114.1	143.8	124.7	116.8	142.6	122.4	102.2

Pressure surface of the  $y = 0$  aerofoil, points b1 to b3

	$x/c = -0.25$ (b1)			$x/c = 0.0$ (b2)			$x/c = +0.25$ (b3)		
	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$	$\omega$	$2\omega$	$3\omega$
<b>Envia</b>	138.0	128.6	104.5	141.5	121.4	103.0	140.5	119.5	97.6
<b>Ghillani (1)</b>	135.4	130.7	107.9	139.5	122.0	115.7	138.7	118.7	108.8
<b>Ghillani (2)</b>	135.8	127.7	107.2	139.6	118.6	112.2	138.5	117.3	108.2
<b>Escribano</b>	138.2	128.8	112.9	140.3	119.7	115.6	140.5	122.6	106.3
<b>Hixon</b>	142.8	133.0	116.5	142.6	125.4	119.7	143.2	125.5	97.1

## A. NUMERICAL RESULTS OF THE FOURTH TEST CASE

---

Table A.2: Fourth test case, comparison of the pressure amplitude of the relevant mode orders along the inflow and the outflow planes from various workshop contributors (see Dahl [2004]). Data in dB.

	<b>Inflow</b>				
	<i>m</i> = +11	<i>m</i> = -16	<i>m</i> = -5	<i>m</i> = +6	<i>m</i> = -21
<b>Envia</b>	101.3	113.0	116.8	97.6	88.1
<b>Ghillani (1)</b>	88.6	103.1	118.4	98.8	100.4
<b>Ghillani (2)</b>	88.0	95.2	116.2	90.7	88.9
<b>Escribano</b>	91.1	96.0	118.4	106.1	97.0
<b>Hixon</b>	104.0	103.9	121.2	104.7	100.8

	<b>Outflow</b>				
	<i>m</i> = +11	<i>m</i> = -16	<i>m</i> = -5	<i>m</i> = +6	<i>m</i> = -21
<b>Envia</b>	108.4	83.8	119.2	95.6	98.0
<b>Ghillani (1)</b>	118.0	116.0	120.2	104.9	106.7
<b>Ghillani (2)</b>	121.8	119.8	117.6	91.0	104.1
<b>Escribano</b>	106.4	79.5	120.0	99.3	110.9
<b>Hixon</b>	110.6	86.4	123.2	101.7	102.9

# Bibliography

- ABARBANEL, S., GOTTLIEB, D. & HESTHAVEN, J.S. (1999). Well-posed Perfectly Matched Layers for advective acoustics. *Journal of Computational Physics*, **154**, 266–283.
- AHMADI, M. & GHALY, W.S. (1996). A finite volume method for the two-dimensional Euler equations with solution adaptation on unstructured meshes. In *6th ASME International Congress on Fluid Dynamics and Propulsion, Egypt*.
- ALLAMPALLI, V., HIXON, R., NALLASAMY, M. & SAWYER, S.D. (2009). High-accuracy large-step explicit Runge-Kutta (HALE-RK) schemes for computational aeroacoustics. *Journal of Computational Physics*, **228**, 3837–3850.
- ATASSI, H. & HAMAD, G. (1981). Sound generated in a cascade by three-dimensional disturbances convected in a subsonic flow. In *AIAA 7th Aeroacoustics Conference, Palo Alto, California, USA*, AIAA Paper 81-2046.
- BERLAND, J., BOGEY, C. & BAILLY, C. (2006). Low-dissipation and low-dispersion fourth-order Runge-Kutta algorithm. *Computers & Fluids*, **35**, 1459–1463.
- BIN, J., CHEONG, C. & LEE, S. (2004). Simulation of cascade - gust interaction by using grid-optimized dispersion-relation-preserving methods. In *Fourth computa-*

- tional aeroacoustics (CAA) workshop on benchmark problems*, 2004-212954, 495, National Aeronautics and Space Administration.
- BOGEY, C. & BAILLY, C. (2004). A family of low dispersive and low dissipative explicit schemes for flow and noise computations. *Journal of Computational Physics*, **194**, 194–214.
- BOGEY, C. & BAILLY, C. (2006). Computation of a high Reynolds number jet and its radiated noise using large eddy simulation based on explicit filtering. *Computers & Fluids*, **35**, 1344–1358.
- CALVO, M., FRANCO, J.M. & RANDEZ, L. (2003). Minimum storage Runge-Kutta schemes for computational acoustics. *Computers and Mathematics with Applications*, **45**, 535–545.
- CALVO, M., FRANCO, J.M. & RANDEZ, L. (2004). A new minimum storage Runge-Kutta scheme for computational acoustics. *Journal of Computational Physics*, **201**, 1–12.
- CARPENTER, M.H. & KENNEDY, C.A. (1994). *Fourth-order 2N-storage Runge-Kutta schemes*. NASA technical memorandum, 109112, National Aeronautics and Space Administration, Ames Research Center; National Technical Information Service, distributor, Langley Research Center, Hampton, Virginia.
- CHEN, X.X., ZHANG, X., MORFEY, C.L. & NELSON, P.A. (2004). A numerical method for computation of sound radiation from an unflanged duct. *Journal of Sound and Vibration*, **270**, 573–586.

- CHEONG, C. & LEE, S. (2001). Grid-optimized dispersion-relation-preserving schemes on general geometries for computational aeroacoustics. *Journal of Computational Physics*, **174**, 248–276.
- CHEONG, C., JOSEPH, P. & LEE, S. (2006). High frequency formulation for the acoustic power spectrum due to cascade-turbulence interaction. *Journal of the Acoustical Society of America*, **119**, 108–122.
- CHEONG, C., JURDIC, V. & JOSEPH, P. (2009). Decomposition of modal acoustic power due to cascade-turbulence interaction. *Journal of Sound and Vibration*, **324**, 57–73.
- COLONIUS, T. & LELE, S.K. (2004). Computational aeroacoustics: progress on nonlinear problems of sound generation. *Progress in Aerospace Sciences*, **40**, 345–416.
- COUPLAND, J. (2004). Linear unsteady CFD analysis of the cascade-gust interaction problem. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems, 2004-212954*, 495, National Aeronautics and Space Administration.
- DAHL, M.D., ed. (2000). *Third computational aeroacoustics (CAA) workshop on benchmark problems*, NASA Conference Publication, National Aeronautics and Space Administration.
- DAHL, M.D., ed. (2004). *Fourth computational aeroacoustics (CAA) workshop on benchmark problems*, NASA Conference Publication, National Aeronautics and Space Administration.
- DOWLING, A.P. & FLOWCS WILLIAMS, J.E. (1983). *Sound and sources of sound*. E. Horwood.

- ENGQUIST, B. & MAJDA, A. (1977). Absorbing boundary-conditions for numerical-simulation of waves. *Mathematics of Computation*, **31**, 629–651.
- ENVIA, E. (1988). *Influence of vane sweep on rotor-stator interaction noise*. Ph.D. thesis.
- ENVIA, E. (2000). Analytical solution of the category 3, benchmark problems 2 and 3. In *Third computational aeroacoustics (CAA) workshop on benchmark problems, 2000-209790*, 473, National Aeronautics and Space Administration, Langley Research Center.
- ENVIA, E. (2004). Benchmark solution for the category 3 problem 2: cascade-gust interaction. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems, 2004-212954*, 495, National Aeronautics and Space Administration.
- ENVIA, E. & KERSCHEN, E.J. (1986). Noise generated by convected gusts interacting with swept airfoil cascades. In *AIAA 10th Aeroacoustics Conference, Seattle, WA, USA*, AIAA Paper 86-1872.
- ESCRIBANO, A.G., SERRANO, A. & VASCO, C. (2004). Cascade-gust-interaction problem analysis based on linear CFD calculations. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems, 2004-212954*, 495, National Aeronautics and Space Administration.
- EVERS, I. & PEAKE, N. (2002). On sound generation by the interaction between turbulence and a cascade of airfoils with non-uniform mean flow. *Journal of Fluid Mechanics*, **463**, 25–52.

- FYFE, D. (1966). Economical evaluation of Runge-Kutta formulae. *Mathematics of Computation*, **20**, 392–398.
- GAITONDE, D.V. & VISBAL, M.R. (1998). High-order schemes for Navier-Stokes equations: algorithm and implementation into FDL3DI. Tech. rep., Air Vehicles Directorate, Air Force Research Laboratory.
- GAITONDE, D.V. & VISBAL, M.R. (1999). Further development of a Navier-Stokes solution procedure based on higher-order formulas. In *AIAA 37th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA*, AIAA Paper 99-0557.
- GILES, M.B. (1990). Nonreflecting boundary-conditions for Euler equation calculations. *AIAA Journal*, **28**, 2050–2058.
- GILL, S. (1951). A process for the step-by-step integration of differential equations in an automatic digital computing machine. *Mathematical Proceedings of the Cambridge Philosophical Society*, **47**, 96–108.
- GLEGG, S.A.L. (1999). The response of a swept blade row to a three-dimensional gust. *Journal of Sound and Vibration*, **227**, 29–64.
- GROENEWEG, J.F., SOFRIN, T., RICE, E.J. & GLIEBE, P.R. (1995). *Turbomachinery noise*, chap. 3, 151–210. Acoustical Society of America through the American Institute of Physics.
- HAGSTROM, T. & GOODRICH, J. (2003). Accurate radiation boundary conditions for the linearized euler equations in cartesian domains. *Siam Journal on Scientific Computing*, **24**, 770–795.

- HALL, K.C. (1997a). Exact solution to category 3 problems - turbomachinery noise. In *Second computational aeroacoustics (CAA) workshop on benchmark problems*, 3352, 348, National Aeronautics and Space Administration, Langley Research Center.
- HALL, K.C. (1997b). A variational finite element method for computational aeroacoustic calculations of turbomachinery noise. In *Second computational aeroacoustics (CAA) workshop on benchmark problems*, 3352, 348, National Aeronautics and Space Administration, Langley Research Center.
- HANSON, D.B. & HORAN, K.P. (1998). Turbulence/cascade interaction - spectra of inflow, cascade response, and noise. In *4th AIAA/CEAS Aeroacoustics Conference, Toulouse, France*, 688–700.
- HIRSCH, C. (1990). *Numerical computation of internal and external flows, volume 2, computational methods for inviscid and viscous flows*, vol. 2. John Wiley & sons, 1st edn.
- HIXON, R. (1999). Prefactored compact filters for computational aeroacoustics. In *37th Aerospace Sciences Meeting & Exhibit, Reno, Nevada, USA*, AIAA Paper 99-0358.
- HIXON, R. (2000a). Nonlinear comparison of high-order and optimized finite-difference schemes. *International Journal of Computational Fluid Dynamics*, **13**, 259–277.
- HIXON, R. (2000b). Prefactored small-stencil compact schemes. *Journal of Computational Physics*, **165**, 522–541.

- HIXON, R. (2003). Space-time mapping analysis for the accurate calculation of complex unsteady flows. In *AIAA/CEAS 9th Aeroacoustics Conference and Exhibit, Hilton Head, SC, USA*, AIAA Paper 2003-3205.
- HIXON, R. (2004). Solution of category 3, problem 2, using the space-time mapping analysis (STMA) method. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems*, 2004-212954, 495, National Aeronautics and Space Administration.
- HIXON, R., SHIH, S.H. & MANKBADI, R.R. (2000). Evaluation of boundary conditions for the gust-cascade problem. *Journal of Propulsion and Power*, **16**, 72–78.
- HIXON, R., NALLASAMY, M. & SAWYER, S.D. (2002). Parallelization strategy for an explicit computational aeroacoustics code. In *AIAA/CEAS 8th Aeroacoustics Conference and Exhibit, Breckenridge, CO, USA*, AIAA Paper 2002-2583.
- HIXON, R., NALLASAMY, M., SAWYER, S. & DYSON, R. (2003). Mean flow boundary conditions for computational aeroacoustics. In *9th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2003-3299.
- HIXON, R., NALLASAMY, M., SAWYER, S. & DYSON, R. (2004). Unsteady validation of a mean flow boundary condition for computational aeroacoustics. In *42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA*, no. AIAA Paper 2003-3299 in 42nd AIAA Aerospace Sciences Meeting and Exhibit, 5052–5061.
- HIXON, R., SESCU, A. & ALLAMPALLI, V. (2010). *Towards the prediction of noise from realistic rotor wake/stator interaction using CAA*, vol. 6 of *Procedia Engineering*, 203–213.

- HIXON, R., SESCO, A. & SAWYER, S. (2011). Vortical gust boundary condition for realistic rotor wake/stator interaction noise prediction using computational aeroacoustics. *Journal of Sound and Vibration*, **330**, 3801–3817.
- HU, F.Q. & MANTHEY, J.L. (1997). Application of PML absorbing boundary conditions to the benchmark problems of computational aeroacoustics. In *Second computational aeroacoustics (CAA) workshop on benchmark problems*, 3352, 348, National Aeronautics and Space Administration, Langley Research Center.
- HU, F.Q., HUSSAINI, M.Y. & MANTHEY, J.L. (1996). Low-dissipation and low-dispersion Runge-Kutta schemes for computational acoustics. *Journal of Computational Physics*, **124**, 177–191.
- HUNG, C.M. (2002). Definition of contravariant velocity components. In *3rd AIAA Theoretical Fluid Mechanics Meeting, Saint Louis, Missouri, USA*, NASA 20020059532.
- KENNEDY, C.A. & CARPENTER, M.H. (1994). Several new numerical methods for compressible shear-layer simulations. *Applied Numerical Mathematics*, **14**, 397–433.
- KENNEDY, C.A., CARPENTER, M.H. & LEWIS, R.M. (2000). Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations. *Applied Numerical Mathematics*, **35**, 177–219.
- KETCHESON, D.I. (2010). Runge-Kutta methods with minimum storage implementations. *Journal of Computational Physics*, **229**, 1763–1773.
- KIM, J.W. & LEE, D.J. (2000). Generalized characteristic boundary conditions for computational aeroacoustics. *AIAA Journal*, **38**, 2040–2049.

- KIM, J.W. & LEE, D.J. (2003). Characteristic interface conditions for multiblock high-order computation on singular structured grid. *AIAA Journal*, **41**, 2341–2348.
- KIM, J.W. & LEE, D.J. (2004). Generalized characteristic boundary conditions for computational aeroacoustics, part 2. *AIAA Journal*, **42**, 47–55.
- KOCH, W. (1971). On the transmission of sound waves through a blade row. *Journal of Sound and Vibration*, **18**, 111–128.
- KREISS, H.O. (1970). Initial boundary value problems for hyperbolic systems. *Communications on Pure and Applied Mathematics*, **23**, 277–298.
- LELE, S.K. (1992). Compact finite-difference schemes with spectral-like resolution. *Journal of Computational Physics*, **103**, 16–42.
- LOCKARD, D.A. & MORRIS, P.J. (1997). A parallel simulation of gust/cascade interaction noise. In *Second computational aeroacoustics (CAA) workshop on benchmark problems*, 3352, 348, National Aeronautics and Space Administration, Langley Research Center.
- LOCKARD, D.P. & MORRIS, P.J. (1998). Radiated noise from airfoils in realistic mean flows. *AIAA Journal*, **36**, 907–914.
- LOCKARD, D.P., BRENTNER, K.S. & ATKINS, H.L. (1995). High-accuracy algorithms for computational aeroacoustics. *AIAA Journal*, **33**, 246–251.
- MAJUMDAR, S.J. & PEAKE, N. (1996). Three-dimensional effects in cascade - gust interaction. *Wave Motion*, **23**, 321–337.
- MANI, R. & HORVAY, G. (1970). Sound transmission through blade. *Journal of Sound and Vibration*, **12**, 59–83.

- MEDIDA, S. (2007). *Curvilinear extension to the Giles non-reflecting boundary conditions for wall-bounded flows*. Master's thesis, University of Toledo.
- NALLASAMY, M., HIXON, R., SAWYER, S.D. & DYSON, R.W. (2004). Category 3: sound generation by interacting with a gust problem 2: cascade-gust interaction. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems, 2004-212954*, 495, National Aeronautics and Space Administration.
- NALLASAMY, M., HIXON, R. & SAWYER, S. (2007). Solution of unsteady Euler equations: gust-cascade interaction tones. *Computers & Fluids*, **36**, 724–741.
- NASA (2012). <http://www.grc.nasa.gov/WWW/cgns/>. Data-Array Structure Definitions.
- PACHECO, P.S. (1997). *Parallel programming with MPI*. Morgan Kaufmann Publishers, San Francisco, California.
- PEAKE, N. (1993). The scattering of vorticity waves by an infinite cascade of flat plates in subsonic flow. *Wave Motion*, **18**, 255–271.
- POINSOT, T.J. & LELE, S.K. (1992). Boundary-conditions for direct simulations of compressible viscous flows. *Journal of Computational Physics*, **101**, 104–129.
- QUARTERONI, A. & VALLI, A. (1999). *Domain decomposition methods for partial differential equations*. Oxford University Press.
- RAGAB, S.A. & SALEM-SAID, A.H. (2007). Response of a flat-plate cascade to incident vortical waves. *AIAA Journal*, **45**, 2140–2148.
- RONA, A. & SPISSO, I. (2007). Implementation of a high-order finite difference scheme to model wave propagation. In *13th AIAA/CEAS Aeroacoustics Conference, Rome*.

- RUDY, D.H. & STRIKWERDA, J.C. (1980). A nonreflecting outflow boundary-condition for subsonic Navier-Stokes calculations. *Journal of Computational Physics*, **36**, 55–70.
- SALEM-SAID, A. & RAGAB, S.A. (2008). Effects of internal vortex structure and mach number on the parallel vortex-plate interaction. In *38th AIAA Fluid Dynamics Conference and Exhibit*.
- SAXER, A.P. & GILES, M.B. (1993). Quasi-3-dimensional nonreflecting boundary-conditions for Euler equations calculations. *Journal of Propulsion and Power*, **9**, 263–271.
- SMITH, B., BJORSTAD, P. & GROPP, W. (1996). *Domain decomposition: parallel multi-level methods for elliptic partial differential equations*. Cambridge University Press.
- SMITH, S.N. (1972). Discrete frequency sound generation in axial flow turbomachines. Tech. Rep. 3709, Aeronautical Research Council.
- SPISSO, I. (unpublished). Development of a prefactored high-order compact scheme for low-speed aeroacoustics, Ph.D. thesis, University of Leicester, Leicester, UK.
- STANESCU, D. & HABASHI, W.G. (1998). 2N-storage low dissipation and dispersion Runge-Kutta schemes for computational acoustics. *Journal of Computational Physics*, **143**, 674–681.
- TAM, C.K.W. (1995). Computational aeroacoustics - issues and methods. *AIAA Journal*, **33**, 1788–1796.

- TAM, C.K.W. (2004). Computational aeroacoustics: an overview of computational challenges and applications. *International Journal of Computational Fluid Dynamics*, **18**, 547–567.
- TAM, C.K.W. & DONG, Z. (1994). Wall boundary-conditions for high-order finite-difference schemes in computational aeroacoustics. *Theoretical and Computational Fluid Dynamics*, **6**, 303–322.
- TAM, C.K.W. & DONG, Z. (1996). Radiation and outflow boundary conditions for direct computation of acoustic and flow disturbances in a nonuniform mean flow. *Journal of Computational Acoustics*, **4**, 175–201.
- TAM, C.K.W. & HARDIN, J.C., eds. (1997). *Second computational aeroacoustics (CAA) workshop on benchmark problems*, NASA conference publication, National Aeronautics and Space Administration, Langley Research Center.
- TAM, C.K.W. & WEBB, J.C. (1993). Dispersion-relation-preserving finite-difference schemes for computational acoustics. *Journal of Computational Physics*, **107**, 262–281.
- TAM, C.K.W., KURBATSKII, K.A. & FANG, J. (1997). Numerical boundary conditions for computational aeroacoustics benchmark problems. In *Second computational aeroacoustics (CAA) workshop on benchmark problems*, 3352, 348, National Aeronautics and Space Administration, Langley Research Center.
- TAM, C.K.W., AURIAULT, L. & CAMBULI, F. (1998). Perfectly matched layer as an absorbing boundary condition for the linearized Euler equations in open and ducted domains. *Journal of Computational Physics*, **144**, 213–234.

- THOMPSON, K.W. (1987). Time-dependent boundary-conditions for hyperbolic systems. *Journal of Computational Physics*, **68**, 1–24.
- THOMPSON, K.W. (1990). Time-dependent boundary-conditions for hyperbolic systems, II. *Journal of Computational Physics*, **89**, 439–461.
- TYLER, J.M. & SOFRIN, T.G. (1962). Axial flow compressor noise studies. *SAE Transactions*, **70**, 309–332.
- UK DEPARTMENT OF TRANSPORT (2003). *The future of air transport - white paper and the civil aviation bill*. UK Government.
- VAN DER HOUWEN, P.J. (1972). Explicit runge-kutta formulas with increased stability boundaries. *Numerische Mathematik*, **20**, 149–164.
- VARIOUS (2005). *Annex 16 - Environmental protection - volume 1 - aircraft noise (ed.5)*. International Civil Aviation Organization.
- VISBAL, M.R. & GAITONDE, D. (1998). High-order accurate methods for unsteady vortical flows on curvilinear meshes. In *36th Aerospace Sciences Meeting & Exhibit, Reno, Nevada, USA*, AIAA 98-0131.
- VISBAL, M.R. & GAITONDE, D.V. (2002). On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. *Journal of Computational Physics*, **181**, 155–185.
- WANG, X.Y., HIMANSU, A., CHANG, S.C. & JORGENSON, P.C.E. (2000). Applications of the space-time conservation element and solution element (CE/SE) method to computational aeroacoustic benchmark problems. In *Third computational aeroacoustics*

- (CAA) workshop on benchmark problems, 2000-209790, 473, National Aeronautics and Space Administration, Langley Research Center.
- WANG, X.Y., HIMANSU, A., CHANG, S.C. & JORGENSEN, P.C.E. (2004). Computation of a single airfoil gust response and gust-cascade interaction using the CE/SE method. In *Fourth computational aeroacoustics (CAA) workshop on benchmark problems*, 2004-212954, 495, National Aeronautics and Space Administration.
- WASISTHO, B., GEURTS, B.J. & KUERTEN, J.G.M. (1997). Simulation techniques for spatially evolving instabilities in compressible flow over a flat plate. *Computers & Fluids*, **26**, 713–739.
- WEI, D. & CHEONG, C. (2010). Frequency-domain computation of inflow broadband noise due to interaction of a rectilinear cascade of flat plates with incident turbulence. *Journal of mechanical science and technology*, **24**, 2431–2440.
- WHITEHEAD, D.S. (1987). *Classical two-dimensional methods*, vol. 1, chap. 3, 59–88. Advisory Group for Aerospace Research and Development.
- WILLIAMSON, J.H. (1980). Low-storage Runge-Kutta schemes. *Journal of Computational Physics*, **35**, 48–56.