ISSN 2176-5480

22nd International Congress of Mechanical Engineering (COBEM 2013)
November 3-7, 2013, Ribeirão Preto, SP, Brazil

# METHOD FOR AEROACOUSTIC ANALYSIS IN ASYMMETRIC WAKE FLOW

**Marlon Sproesser Mathias**
**Marcello Augusto Faraco de Medeiros**
Departamento de Engenharia Aeronáutica - Escola de Engenharia de São Carlos - USP - Av. João Dagnone, 1100, São Carlos, SP
marlon.mathias@usp.br
marcello@sc.usp.br

***Abstract.*** *Bluff bodies immersed in flows generate wakes, such wakes may be asymmetric if the body generates lift or is immerse in a shearing flow. Vortex are generated and pairings happen in the wake, generating sound. The asymmetrical characteristics influence the pairings and hence the intensity and directivity of the sound generated. This work involves numerical solution of the compressible Navier-Stokes equations and post-processing of the results in order to compute noise intensity in different regions of the flow. The sound level is measured by the flow dilatation in each region, calculated from the output of the solver. The aim is to develop an appropriate direct numerical solver and understand how each parameter of the wake affects intensity and directivity of the sound. Such knowledge, in the design of an aircraft, might be used in directing the sound from wings away from observers in the ground, projecting quieter high lift devices and designing quieter turbofan engines in order to meet stricter environmental standards.*

***Keywords:*** *Aeroacoustics, Vortex Pairing, Asymmetric Wake, Direct Numerical Simulation, Navier-Stokes*

## 1. INTRODUCTION

Aircrafts in take off or landing procedures generate a high level of sound, with engines getting more and more silent, high-lift devices are becoming responsible for a greater part of the noise. Those devices, just like other bluff bodies that generate lift, create an asymmetrical wake downstream.

Wakes at certain Reynolds numbers start forming vortices (Gennaro, 2008) and their pairings are among the biggest sound sources in a wake. Figure 1 shows their forming at a Reynolds number of 300 (Willianson, 1996).

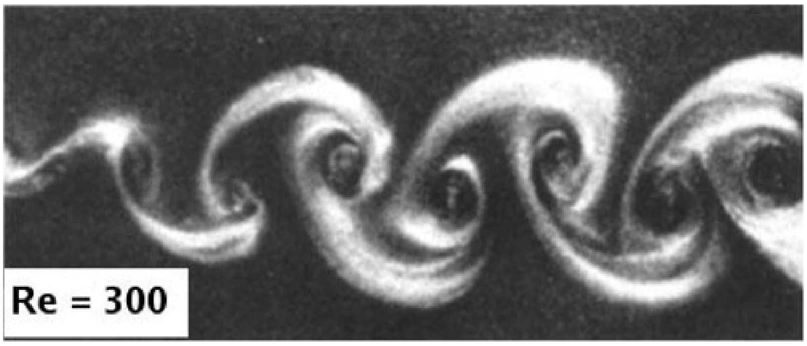

Figure 1. Von Karman vortex formation at $Re = 300$

The asymmetry causes vortices on both side to be different from each other, which causes a directivity in the sound. Understanding how this sound can be attenuated or directed away from observers in the ground is an important step towards meeting stricter environmental requirements in the future.

Figure 1 illustrates the wake's spatial development, in this paper, another approach is taken. Similarly to (Colaciti, 2009), who studied instability and aeroacoustics on a mixing layer, in this paper periodical domain will be used to simulate the temporal development of a set of vortices, from their forming to their pairing. By doing this, computational costs are greatly reduced and boundary conditions in the inflow and outflow are much simpler, while still allowing the directivity and intensity variations to be measured.

In a periodical domain, the vortex frequency must be known so that the domain size can be determined before the simulation is run. (Gennaro, 2008) describes the most unstable wave numbers and how they relate to the asymmetry. Simulations will be run for that range of values to determine their acoustical effect.

## 2. METHODS

An aeroacoustical analysis requires a precise modeling of turbulence in order correctly predict sound generation. Direct Numeric Simulation was chosen as it solves flow equations at all scales, only limited by the mesh, and uses no turbulence modeling. The DNS solves the Compressible Navier-Stokes Equations. The computational domain is two dimensional, periodical in the flow direction and open in the other, two vortices are generated and paired during the simulation.

Initially, the DNS described by (Colaciti, 2009) was modified and used, the output was post-processed in order to obtain the dilatation values, and hence, sound intensity. Later a new DNS was written.

The new DNS is written in Matlab® and has a comparable performance to others written in lower level languages like Fortran.It Achieves times of one microsecond or less per mesh node per time-step in common modern processors. The code takes advantage of Matlab®'s very high efficiency when operating with matrices and vectors, also allowing easy use of parallelization and GPU processing.

The DNS receives the flow initial condition and calculates it through time. The initial condition is given by the base flow plus a small disturbance in order to excite its most unstable mode.

Dilatation values on both sides of the wake are measured and stored.

All variables are adimensional and related to Reynolds number ($Re$), Mach number ($Ma$) and Prandtl number ($Pr$).

### 2.1 Governing equations

Compressible bi-dimensional Navier-Stokes equations are used in this DNS. It solves for products of the primitive variables, as described by (Anderson, 1995). Equation (1) contains the variables for which the solutions will be calculated, Eqs. (2) and (3) are the flux terms and Eq. (4) relates them in a form that can be numerically integrated.

$$\mathbf{U} = \left\{ \begin{array}{l} \rho \\ \rho u \\ \rho v \\ \rho\left(e + \frac{V^2}{2}\right) \end{array} \right\} \tag{1}$$

$$\mathbf{F} = \left\{ \begin{array}{l} \rho u \\ \rho u^2 + p - \tau_{xx} \\ \rho v u - \tau_{xy} \\ \rho\left(e + \frac{V^2}{2}\right) u + pu - k\frac{\partial T}{\partial x} - u\tau_{xx} - v\tau_{xy} \end{array} \right\} \tag{2}$$

$$\mathbf{G} = \left\{ \begin{array}{l} \rho v \\ \rho u v - \tau_{xy} \\ \rho v^2 + p - \tau_{yy} \\ \rho\left(e + \frac{V^2}{2}\right) v + pv - k\frac{\partial T}{\partial y} - u\tau_{xy} - v\tau_{yy} \end{array} \right\} \tag{3}$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\frac{\partial \mathbf{F}}{\partial x} - \frac{\partial \mathbf{G}}{\partial y} \tag{4}$$

After the solution is calculated, the primitive variables are obtained from the $U$ vector.

### 2.2 Adimensionalization

All calculations are made with adimensional variables, therefore, results will be suitable for various scales. The unitary length is defined as the wake's width, velocity is normalized to the mean free-stream flow. Fluid's initial density ($\rho_0$), molar mass ($Mmol$) and heat capacity with constant volume ($C_v$) are also unitary.

The fluids viscosity ($\mu$), base temperature ($T_0$) and thermal conductivity ($k$) are calculated in order to achieve the required $Re$, $Ma$ and $Pr$, respectively, by using Eqs. (5), (6) and (7), where $R$ is the universal gas constant and $Re$ is normalized to the wake width ($L_w$).

$$\mu = \frac{\rho_0}{Re} L_w \tag{5}$$

$$T_0 = \frac{1}{R\gamma Ma^2} \tag{6}$$

$$k = \frac{C_v\gamma\mu}{Pr} \tag{7}$$

## 2.3 Initial condition

The asymmetric wake base flow used is given by Eq. (8), where $u$ is the velocity in flow direction and $y$ is the position normal to the wake.

$$u(y) = A\left(1 - e^{-Bx^2}\right) + C\tanh(Dy) \tag{8}$$

$A$ is related to the base flow velocity relative the the center of the wake and, in this work, always normalized to 1, a value of zero would result in a purely shearing flow. $C$ is the asymmetry, a value of zero would produce a symmetrical wake. $B$ and $D$ control the wake width and chosen so that the width is always unitary.

The width is measured as the distance between the values of $y$ where $u$ is half the free-stream velocity for both sides of the wake.

$v$ for the base flow is zero, but a small disturbance is added to it so that vortex are formed and paired in the desired frequency and in less time steps, reducing the computational cost.

Sines of two frequencies are used, according to Eq. (9), where $L_x$ is the domain length.

$$v(x,y) = e^{-E_1y^2}U_1 \sin\left(\frac{2\pi}{L_x}x\right) + e^{-E_2y^2}U_2 \sin\left(\frac{4\pi}{L_x}x\right) \tag{9}$$

$U_1$ and $U_2$ are directly related to the disturbance intensity, large values reduce time steps needed, but they should still be small enough so that results are unaffected. $E_1$ and $E_2$ relate to the disturbance width, values should be small enough so that derivatives close to the center won't become too high and large enough so that disturbance values close to the boundary are very close to zero.

Figure 2 shows an example of initial $u$ and $v$ velocities as well as a velocity field, note that, in the field, $v$ was greatly amplified for better visualization.

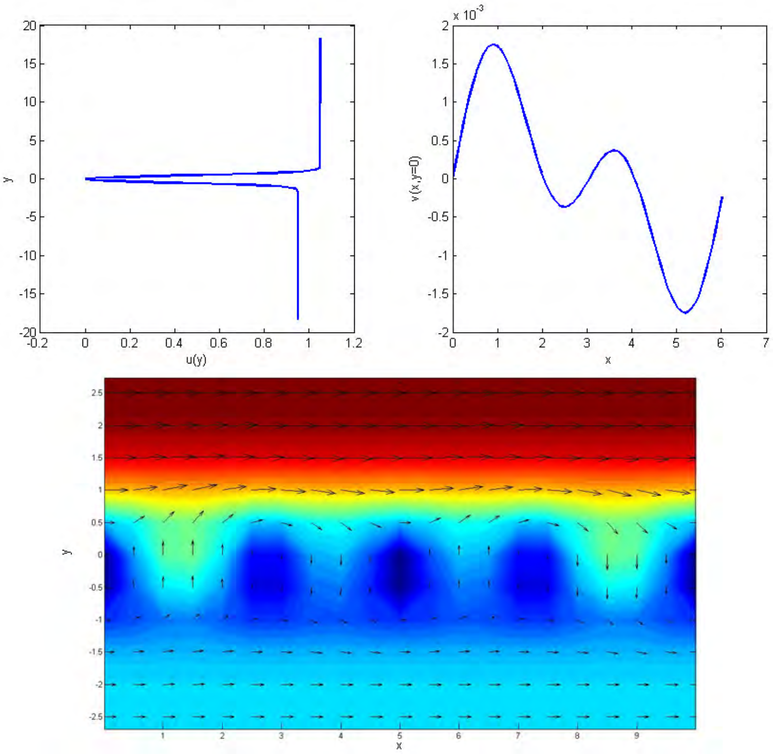

Figure 2. Example of initial velocities

Initial fluid density is uniform across the domain and equals to $\rho_0$. The initial temperature is given by the Crocco-Busemann relation, as used by (Colaciti, 2009), Eq. (10) shows the initial temperature field.

$$T(x,y) = T_0 \left(1 + r\frac{\gamma - 1}{2} Ma^2 \left(U(x,y)^2 + V(x,y)^2\right)\right) \tag{10}$$

### 2.4 Mesh

This DNS uses a bi-dimensional structured mesh, uniform in $x$ direction and stretched in $y$ direction.

The vortices's wave number, $\alpha$, defines domain length in $x$, $L_x$ according to Eq. (11), size in $y$ direction, $L_y$ , is chosen so that domain's border are far enough from the vortices and sound waves to be formed and aligned before being measured.

$$L_x = \frac{2}{\alpha\, Ma} \tag{11}$$

$n_x$ columns of mesh nodes are uniformly distributed in $x$ direction and $n_y$ rows of nodes are distributed in $y$ according to Eq. (12), $A$ is a stretching parameter. This distribution was chosen because the distance from one row to the next increases almost uniformly as $|y|$ increases.

$$y_i = L_y \frac{\text{sinh}\left(A\left(\frac{2i}{n_y - 1} - 1\right)\right)}{2\text{sinh}(A)} \quad i = 0 \ldots (n_y - 1) \tag{12}$$

Buffer zones are created close to both boundaries in $y$ direction. They greatly reduce wave reflection and numerical instabilities in those regions. Their working will be further explained in the Numerical time integration subsection.

### 2.5 Spatial derivatives

For spatial derivatives, 4th order finite difference scheme is used in both directions.As $x$ direction in periodical and uniform, the centered scheme is used on the whole domain. For $y$ derivatives, because of the non-uniform mesh, a $\xi$ transformed space must be used in which $y_i$ corresponds to $\xi_i$, the derivative is calculated by using Eq. (13), where both terms on the right-hand side can be calculated by using finite differences. Derivatives on nodes close to both boundaries are approximated by derivation of 5-point Lagrange polynomials.

$$\frac{\partial u}{\partial y} = \frac{\partial u}{\partial \xi}\frac{\partial \xi}{\partial y} \tag{13}$$

Matrices are created with all coefficients so, during DNS execution, derivatives for each direction can be calculated by a single matrix multiplication. They are shown in Eqs. (14), (15) and (16).

$$\mathbf{D_x} = \frac{1}{\Delta x}\begin{bmatrix} 0 & -a_1 & -a_2 & & & a_2 & a_1 \\ a_1 & 0 & -a_1 & -a_2 & & & a_2 \\ \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & a_2 & a_1 & 0 & -a_1 & -a_2 & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ -a_2 & & & a_2 & a_1 & 0 & -a_1 \\ -a_1 & -a_2 & & & a_2 & -a_1 & 0 \end{bmatrix}_{[n_x \times n_x]} \tag{14}$$

$$\mathbf{D_\xi} = \begin{bmatrix} 0 & a_1 & a_2 & & & -a_2 & -a_1 \\ -a_1 & 0 & a_1 & a_2 & & & -a_2 \\ \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & -a_2 & -a_1 & 0 & a_1 & a_2 & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ a_2 & & & -a_2 & -a_1 & 0 & a_1 \\ a_1 & a_2 & & & -a_2 & -a_1 & 0 \end{bmatrix}_{[n_y \times n_y]} \tag{15}$$

$$\mathbf{D_y^*} = (\mathbf{I_{n_y}}(\mathbf{D_\xi y}))^{-1}\mathbf{D_\xi} \tag{16}$$

For 4th order approximations, $a_1 = 2/3$ and $a_2 = -1/12$. The final $D_y$ matrix is obtained by changing rows 1, 2, $n_y - 1$ and $n_y$ of $D_y^*$ by the coefficients of the Lagrange polynomial derivate. The scheme close to the boundaries is changed so boundary conditions are observed.

Higher order schemes may be obtained by changing $a_1$ and $a_2$, adding enough $a_i$ coefficients to the matrices and using Lagrange polynomials for rows in which the centered stencil would go beyond the boundaries.

The boundary conditions used are a fixed density and zero normal velocity at both exits in the $y$ direction, while $x$ is periodical.

The DNS stores all variables in 4 matrices, $Y_i$ one for each variable of the $U$ vector from Eq. (1). Cell $(i,j)$ of each matrix stores the value for position $(x_i, y_j)$. This way, derivatives for $x$ and $y$ directions are approximated by Eqs. (17) and (18), respectively.

$$\frac{\partial \mathbf{Y_i}}{\partial x} \approx \mathbf{Y_i D_x} \tag{17}$$

$$\frac{\partial \mathbf{Y_i}}{\partial y} \approx \mathbf{D_y Y_i} \tag{18}$$

The matrices order for obtaining the derivatives is changed so that for $x$ derivatives, the rows of $Y_i$ are multiplied by he right coefficients and for $y$ derivatives, the columns of $Y_i$ are multiplied by the coefficients.

### 2.6 Numerical time integration

The numerical integration through time is done by a 4th order Runge-Kutta method, shown in Eqs. (19) to (23).

$$\mathbf{K_1} = f(\mathbf{Y^k}) \tag{19}$$

$$\mathbf{K_2} = f\left(\mathbf{Y^k} + \frac{dt}{2}\mathbf{K_1}\right) \tag{20}$$

$$\mathbf{K_3} = f\left(\mathbf{Y^k} + \frac{dt}{2}\mathbf{K_2}\right) \tag{21}$$

$$\mathbf{K_4} = f\left(\mathbf{Y^k} + dt\mathbf{K_3}\right) \tag{22}$$

$$\mathbf{Y^{n+1}} = \mathbf{Y^n} + \frac{dt}{6}\left(\mathbf{K_1} + 2\mathbf{K_2} + 2\mathbf{K_3} + \mathbf{K_4}\right) - dt[\sigma(Y^n - Y^0)]_{(i,j)} \tag{23}$$

$Y^n$ represents all 4 matrices with variable values for all nodes in time $t^n$ and $f(Y)$ is the derivative of $Y$ through time, calculated in Eq. (4). $\sigma$ is a damping function, equals to zero in all domain except for the buffer zone, it prevents echoes and instabilities in the boundaries. $\sigma$ must be chosen so that it's big enough to damp waves before they echo back into the domain but should also be small enough so that it won't cause reflections itself. Equation (24) is used on both sides of the domain, $\xi$ is linearly transformed into $y$ in both buffer zones. $I$ and $E$ are coefficients for $\sigma$.

$$\sigma(\xi) = I(1 - \cos(\xi))^E \quad 0 \leq \xi \leq \frac{\pi}{2} \tag{24}$$

The maximum time step is given by the CLF condition, as used by (Germanos, 2009). Equation (25) relates to convection and Eq. (26), to diffusion. The time step must not exceed the value given by any of the equations and is also chosen to be a divisor of the post-processing interval used.

$$dt \leq \frac{CFL}{\left(\frac{\frac{1}{M}+u_{max}}{dx_{min}} + \frac{\frac{1}{M}+v_{max}}{dy_{min}}\right)} \tag{25}$$

$$dt \leq \frac{Re}{\left(\frac{1}{dx_{min}^2} + \frac{1}{dy_{min}^2}\right)} \tag{26}$$

### 2.7 Noise filtering

During execution, results were found to contain high frequency noise originated from various sources, ranging from rounding and truncation errors to high frequency physical phenomena that are not in the scope of this work. This noise caused numerical derivatives to produce erroneously high values, ultimately causing results to diverge. A 8th order implicit filter described by (Gaitonde, 1998) was used. It involves solving the linear system in Eq. (27) for the whole domain. The bar denotes filtered variables.

$$\alpha\overline{u_{i-1}} + \overline{u_i} + \alpha\overline{u_{i+1}} = a_0 ui + \sum_{j=1}^{4}\left(a_j \frac{u_{i-j} + u_{i+j}}{2}\right) \tag{27}$$

Close to boundaries, points outside the domain are excluded from the equation and the coefficients are changed. All coefficients $a_i$ are a function of $\alpha$, the value of $\alpha$ relates to the filtering intensity. The whole system takes the matrix form shown in Eq. (28). $F_1$ is a tridiagonal matrix and $F_2$, eneadiagonal.

$$\mathbf{F_1}\overline{\mathbf{U}} = \mathbf{F_2}\mathbf{U} \tag{28}$$

Noise generation is much more intense in $y$ direction, the filter is applied after each time step in this direction and after a fixed number of time steps in $x$, this reduces the final computational cost.

### 2.8 Post-processing

At the beginning and at multiples of a defined interval, simulation results are processed in order to obtain values relevant to the aeroacoustical problem.

Acoustic intensity is calculated through the flow dilatation, it is, the derivative of $\rho$ with respect to time, it is obtained directly from Eq. (23) when applied to $\rho$. The dilatation's root-mean-square is calculated on both sides of the domain, far from the shearing flow and outside the buffer zone. Values obtained can be compared at various instants and show the asymmetry of the sound at every stage of the vortex forming and pairing.

$u$ and $v$ components of velocity are calculated and used to compute the vorticity.

Matlab's built-in plot tools are used to plot dilatation on both sides and the vorticity, images and values are, then, stored.

## 3. RESULTS

Simulation was run for values of $Re = 1000$, $Ma = 0.4$, $Pr = 0.8$, $\alpha = 0.82$ and an asymmetry $C = 0.1$. For this wave number, the domain's $x$ dimension is $L_x = 6.1$ and a $y$ dimension of $L_y = 48.8$ was chosen. The mesh used has 100 nodes in $x$ and 600 in $y$, the stretching parameter is $A = 1.3$.

For the initial perturbation, values chosen are $U_1 = 5 \times 10^{-4}$, $U_2 = 1 \times 10^{-3}$, $E_1 = 3$ and $E_2 = 3$ for Equation 9. The filter's $\alpha$ parameter was set to 0.2 and filtering in $x$ was done every 10 steps.

The buffer zone is 100 nodes wide and has parameters $I = 8$ and $E = 2$ for Equation 24.

### 3.1 Performance

The code was run on a desktop computer equipped with an Intel® Core™ i5 - 3470 processor and 4 Gb of DDR3 - 1333 RAM, on Matlab® R2013a.

Table 1 contains performance results and maximum memory usage by the Matlab® process. The number of time steps changes with the grid because of the CFL condition. While idle, this install of Matlab® uses around 353 Mb of memory.

During execution the work load was mostly distributed among 2 of the 4 cores available. Matlab® has the built-in capability of parallelizing matrix operations, in this case, all 4 cores weren't fully used most likely because of a memory speed bottleneck.

Execution time per node per step increased at around the fourth root of the number of nodes.

Table 1. DNS execution time and memory used for different mesh sizes. From adimensional time 0 to 10

| Mesh ($n_x \times n_y$) | Time steps | Total time (s) | Time per node per step ($\mu$s) | Memory use (Mb) |
|---|---|---|---|---|
| 100 × 100 | 940 | 7.66 | 0.81 | 362 |
| 100 × 200 | 1035 | 18.55 | 0.89 | 366 |
| 100 × 400 | 1227 | 52.4 | 1.06 | 382 |
| 100 × 600 | 1418 | 101.4 | 1.19 | 396 |
| 150 × 600 | 1840 | 220.6 | 1.33 | 406 |
| 200 × 600 | 2262 | 380.9 | 1.40 | 430 |

### 3.2 Post processed results

The post-processor plots the vorticity close to the vortices in the middle, top and bottom plots show the dilatation above and below the vortices. Note that in all plots, $x$ and $y$ scales are not the same as the domain is much larger in $y$ direction.

Figure 3's left side shows the initial condition obtained. The dilatation obtained is caused by numerical noise from the initial condition, it's order of magnitude is $10^{-16}$. A different value, although still very small, can be observed close to the periodical inflow and outflow boundaries, it is likely caused by rounding errors when computing the sines used on the initial disturbance.

The initial condition may not represent a physically possible flow although it is very close. When iteration through time begins, the small discrepancies can be seen as waves going from the center to the boundaries, Fig. 3's right side shows these waves at the adimensional time $T = 3$, as they enter the buffer zone. These wave's order of magnitude is $10^{-4}$.

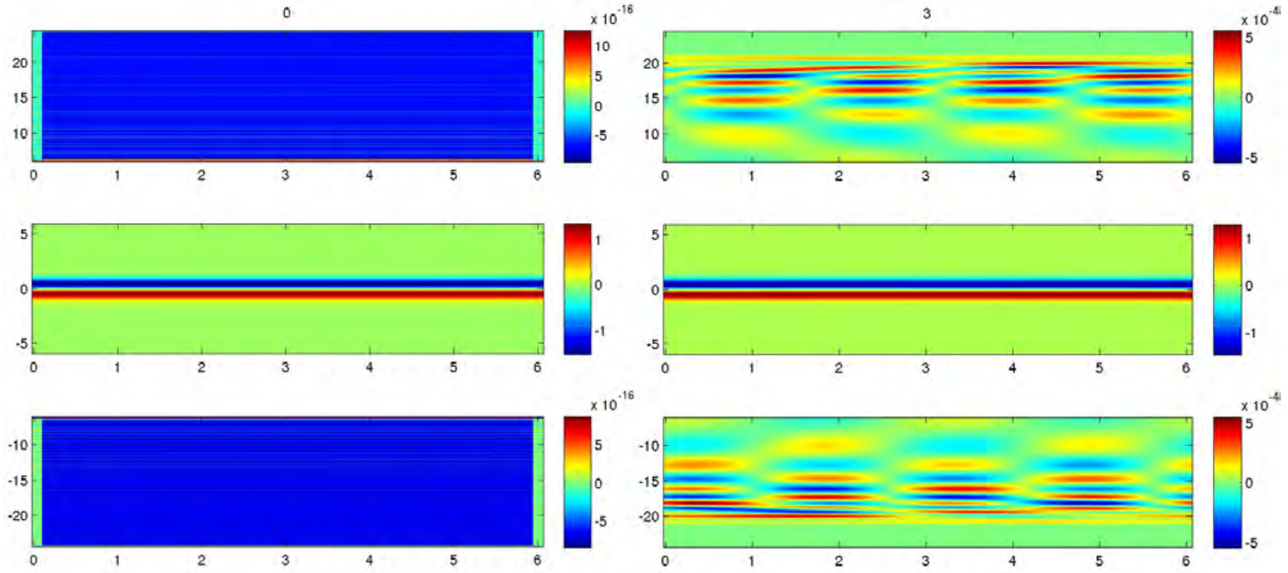

Figure 3. Initial condition (left) and dilatation waves caused by it (right).

For 20 adimensional times, the vorticity plot remains visually the same, after that, oscillations can be observed, from time 30 to 40, vortices are quickly formed and paired. This proves the wake's instability, after a slight disturbance, the flow quickly moves away from the initial condition instead of going back to it.

Figure 4 shows vorticity for various times, illustrating this evolution. At $T = 30$ and $T = 35$ four vortices can be seen, two positive and two negative. At $T = 40$ pairing is happening.

Figure 5 shows vorticity and dilatation for times 20 and 50. At the first, there is no visible high-frequency oscillation, which means that the initial disturbance was successfully damped. At the later, some high-frequency noise can be observed, but it is soon dissipated. It does not cause further numerical instability but is an indication that the filtering scheme or the buffer zone might have to be improved.

### 3.3 Sound Asymmetry

Dilatation values were stored during simulations, the root mean square of dilatation on a 10 node wide section just before the buffer zone is used to compare sound intensity on both sides of the wake. Fig. 6 is a plot of the dilatation on

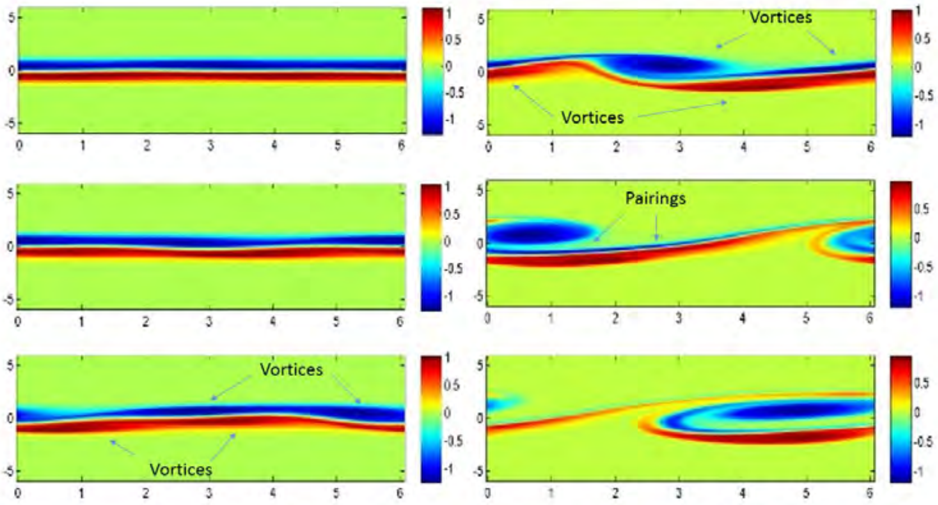

Figure 4. Vorticity for adimensional times 20, 25, 30, 35, 40 and 45.

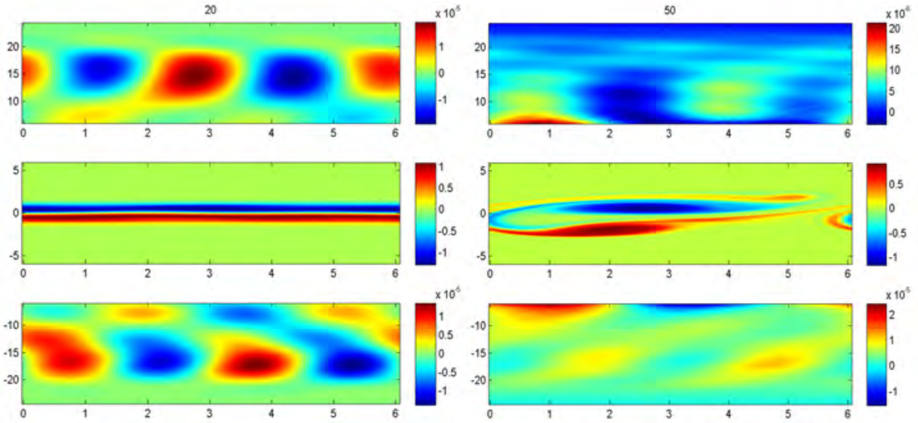

Figure 5. Vorticity and dilatation for times 20 and 50.

both sides through time. The upper side is where velocity is higher.

The first peak was cut off as it is related to the initial condition and has no practical meaning. The peak at time 40 is caused by the vortex pairing. The sound is intenser on the upper side of the domain through almost the whole simulation. The initial condition caused a high frequency oscillation that was later damped to very small values before the pairing happens.

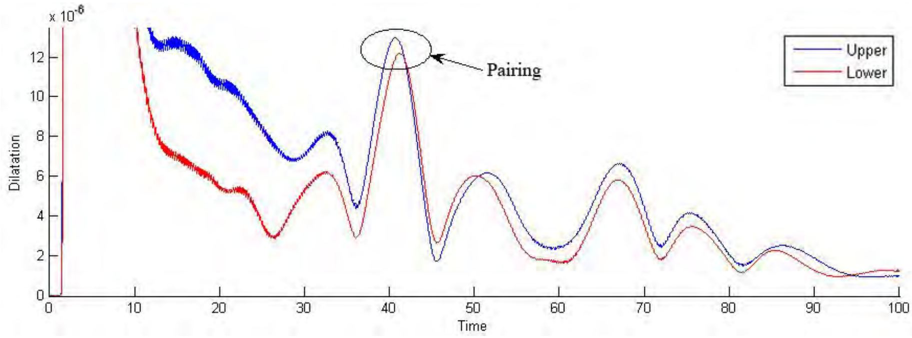

Figure 6. Sound intensity on both sides of the domain and the peak caused by pairing.

Various runs were made with different values for Reynolds and Mach numbers, the results are shown in Fig. 7. The asymmetry is measured as the difference from upper to lower mean dilatation divided by the mean dilatation.

Figure 8 shows mean intensity for those runs.

Asymmetry and intensity were found to increase with Reynolds and Mach at the points tested.

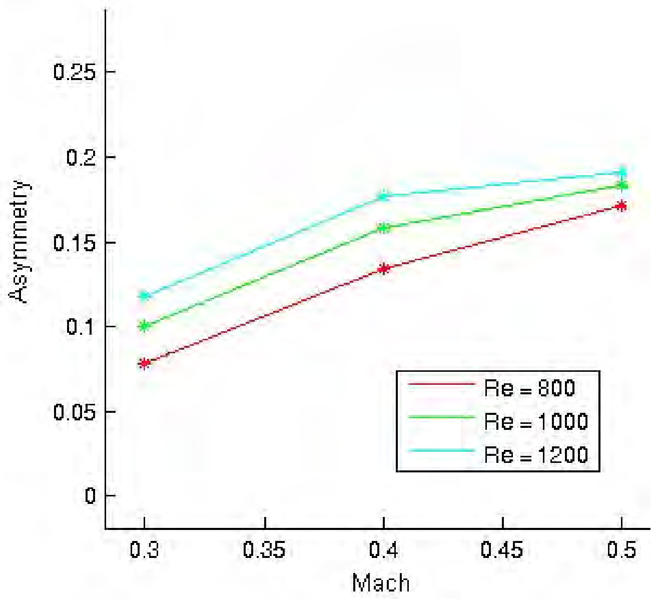

Figure 7. Asymmetry change for various values of $Re$ and $Ma$.

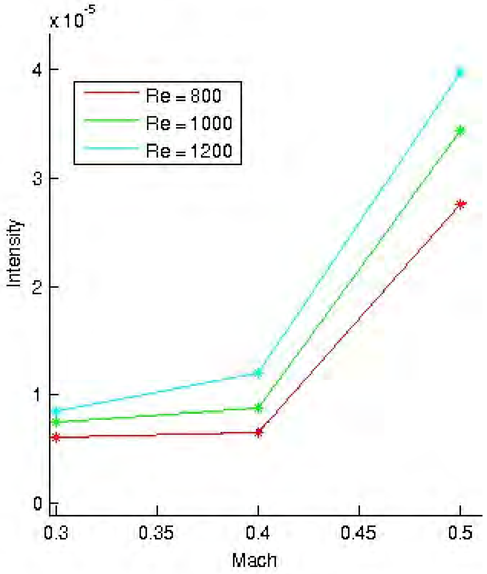

Figure 8. Mean sound intensity for various values of $Re$ and $Ma$.

## 4. DISCUSSION

During this study a DNS was successfully developed and run despite some minor filtering problems. Matlab®, despite being a high-level language, proved itself as a capable CFD tool, as long as the code is written with matrix operations instead of loops.

The influence of Reynolds and Mach numbers over the sound intensity was measured and tools for further analysis of asymmetry, developed. More points will be needed in order to obtain a more precise relation. The wave number and wake asymmetry effect may also be studied.

As results are adimensional, they can be scaled to various problems as long as adimensional values are in an acceptable range for using a DNS. For higher values of $Re$, other kinds of simulation might be more suitable, for example an Large Eddy Solver, that uses mathematical models for small scale phenomena.

This DNS also allows for other kinds of asymmetrical wakes to be analyzed, data from various sources may be used as wake input.

For future works, other filtering schemes may be used, for example the implicit difference schemes with filtering described by (Lele, 1992). A better initial condition could also be used to reduce initial high frequency oscillation.

A non-periodical domain could be run and compared against these results to know how the periodical approximation affects the solution. The non-periodical DNS would also allow measuring upstream-downstream asymmetry.

## 5. ACKNOWLEDGEMENTS

The authors would like to thank Fapesp, Fundação de Amparo à Pesquisa do Estado de São Paulo, for the financial support. And also all the other students and professors who helped and supported this project.

## 6. REFERENCES

Anderson Jr., J.D., 1995. *“Computational Fluid Dynamics”*, McGraw-Hill Education, 1st Edition.

Bogey, C.; Bailly, C., 2004. *“A Family of Low Dispersive and Low Dissipative Explicit Schemes for Flow and Noise Computations"* Laboratoire de Mécanique des Fluides et d’Acoustique, École Centrale de Lyon. Jounal of Computational Physics 194: 194-214.

Choi, H.W.; Moon, Y.J.; Lee, K.J., 2003. *“Numerical Simulation of the Aeroacoustic Noise in the Separated Laminar Boundary Layer”* Departament of Mechanical Engineering, Korea University. KSME International Journal Vol. 17, No. 2: 280-287.

Chung, T.J., 2002. *“Computational Fluid Dynamics”* Cambridge University Press, 1st Edition,.

Colaciti, A.K., 2009. *“Aeroacústica e Instabilidade de uma Camada de Mistura Compressível”* Master thesis, Escola de Engenharia de São Carlos, Universidade de São Paulo.

Gaitonde, D.V.; Visbal, M.R., 1998. *“High-Order Schemes for Navier-Stokes Equations: Algorithm and Implementation into FDL3DI”* Air Vehicles Directorate. Air Force Research Laboratory. Wright-Patterson Air Force Base, Ohio.

Gennaro, E.M., 2008. *“Análise da Instabilidade Hidrodinâmica de uma Esteira Assimétrica”* Master thesis, Escola de Engenharia de São Carlos, Universidade de São Paulo.

Germanos, R.A.C., 2009. *“Simulação Numérica da Evolução Linear e Não-Linear em uma Camada de Mistura Compressível Tridiensional”* Doctoral thesis, Escola de Engenharia de São Carlos, Universidade de São Paulo.

Koch, W., 1985. *“Local Instability Characteristics and Frequency Determination of Self-Excited Wake Flows”* DFVLR/AVA Institut für Theoretische Strömungsmechanik. Journal of Sound and Vibration 99(1): 53-83.

Lele, S.K., 1992. *“Compact Finite Difference Schemes with Spectral-like Resolution”* Center for Turbulence Research, NASA-Ames Research Center. Journal of Computational Physics 103: 16-42.

Meitz, H.L.; Fasel, H.F., 2000. *“A Compact-Difference Scheme for the Navier–Stokes Equations in Vorticity–Velocity Formulation”* Department of Aerospace and Mechanical Engineering, University of Arizona. Journal of Computational Physics 157: 371–403.

White, F.M., 1974. *“Viscous Fluid Flow”*, McGraw-Hill, 1st Edition.

Williamson, C.H.K, *“Vortex Dynamics in the Cylinder Wake”* Mechanical and Aerospace Engineering, Upson Hall, Cornell Uiversity. Annual Review of Fluid Mechanics 28: 477-539.

## 7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.