DIRECT NUMERICAL SIMULATION OF A FLOW AND MATRIX-FREE ANALYSIS OF ITS INSTABILITIES OVER AN OPEN CAVITY

Marlon Sproesser Mathias, marlon.mathias@usp.br
Andres Gaviria M., andres.gaviria@sc.usp.br
Marcello A. F. Medeiros, marcello@sc.usp.br
São Carlos School of Engineering, University of São Paulo, Av. João Dagnone, 1100, CEP 13563-120, São Carlos, SP, Brazil

Abstract. *A method to perform hydrodynamic instability analysis based on Direct Numerical Simulations is presented, it takes a Jacobian-free approach. The method employs high order spatial and temporal discretizations and has very low requirements of computational time and memory compared to conventional matrix-forming methods. The method was validated with calculations of the two-dimensional flow over an open cavity in a flat plate, then, some of the most unstable modes were calculated. Details of the numerical treatment applied to guarantee consistency in the numerical solution are discussed, such as mesh and domain dependencies, and buffer zones for the open boundary conditions. The implementation presented here is applicable to any flow with Cartesian geometry, however the method can be extend to complex geometries and three dimensional flows.*
Keywords: Open cavity flow, Direct Numerical Simulation, Hydrodynamic instability, Matrix-free instability analysis, Shift-invert.

1. INTRODUCTION

An open cavity can be used to represent several different parts of an aircraft in flight, for example the gap between the slats or flaps and the main element of the wing, both at deployed and retracted positions. Better understanding of the flow mechanics over this simplified geometry can aid aircraft designers to predict the transition from laminar to turbulent flow over the wing and the slat's influence over the flow instability, as well as the related acoustic emissions (Colonius, 2001).

In this paper, the development of a DNS for this flow is shown. Special attention is given to the high order spatial and temporal discretizations applied to instability and acoustic analysis. For spatial discretizations various numerical schemes were employed in the DNS, including compact finite differences with spectral-like resolution (Lele, 1992). These schemes allow to resolve a wider range of scales, than traditional finite difference schemes. The time integration was performed with a standard fourth-order Runge Kutta method.

Depending on the Reynolds and Mach numbers, and cavity aspect ratio, the flow can reach a stationary or a periodic state, (Brès and Colonius, 2008) bring a diagram showing this fact for a fixed cavity aspect ratio. By increasing the Reynolds number, three-dimensional instabilities are observed at lower Mach numbers, while two-dimensional instabilities tend to be triggered first if the Mach number is higher. For a length to depth ratio of 2, flows above a Re_D of roughly 1300 are 3D unstable. The minimum Re for a 2D unstable flow is reduced as the Mach number increases, becoming lower than the 3D instability threshold around Mach 0.4. These instabilities eventually saturate to become periodic oscillations in the flow.

This DNS is also intended to be used in conjunction with matrix-free instability analysis methods to compute the eigenvalues of the flow and their respective eigenfunctions. Regular matrix-forming methods would demand an enormous amount of memory to compute the flow modes, in the order of the squared total number of nodes in the domain multiplied by the number of variables, which would easily scale into gigabytes or even terabytes of memory solely to store the required matrices (Theofilis, 2011). This matrix-free method is based on the work of Eriksson and Rizzi (1985), who have implemented a method based on Arnoldi (1951) to find the most unstable or least stable modes of an inviscid flow around an airfoil.
Later works have applied the same methods on the full Navier-Stokes equations (Chiba, 1998; Tezuka and Suzuki, 2006) and also included shift-invert strategies to focus on eigenvalues in a specific region of the complex plane (Gómez, 2015).

2. METHODS

2.1 Direct Numerical Simulation

The DNS code was developed by our group, for several kinds of problems on hydrodynamic instability and aeroacoustics. The code was adapted for the open cavity flow. It is written in Fortran 90 and uses a domain decomposition technique by using the library 2decomp&fft and MPI for parallelization. The code uses a structured mesh in a rectangular domain. The grid is much finer in the cavity and close to the flat plate in the wall-normal direction. There is also a lower level of stretching the stream-wise direction, concentrating nodes on the cavity, and making sure there are nodes positioned exactly on the cavity edges. Closer to the open boundaries, in both directions, the spacing is greatly increased, as part of the buffer zone. No multidomain technique is used in this case, grid points inside the wall, before and after the cavity, are

ignored by the solver. Figure 1 illustrates the domain, the buffer zone is shown in blue, while points inside the wall are orange, darker gray levels represent a finner grid.

Both the flat plate and the cavity walls have no-slip and no-penetration boundary conditions, as well as zero pressure gradient in the normal direction and constant temperature. Just upstream of the flat plate, there is a free-slip region to represent free flow before the leading edge of the no-slip wall. The inflow boundary is defined as an uniform flow at constant temperature and zero stream-wise pressure gradient. At the outflow pressure is kept constant, velocity is extrapolated by imposing second derivative equal to zero.

The two top corners in the cavity were points of special attention for the boundary conditions. The Neumann condition for pressure in other points is observed by computing the density in each node of the boundary so that the derivative of the pressure is null in the normal direction at the wall. Both corner nodes present a problem as walls in distinct directions meet and the density that observes the Neumann condition in one direction might violate it for the other. It was decided that the pressure in both nodes would be the mean of the pressures that meet the boundary condition for each direction.

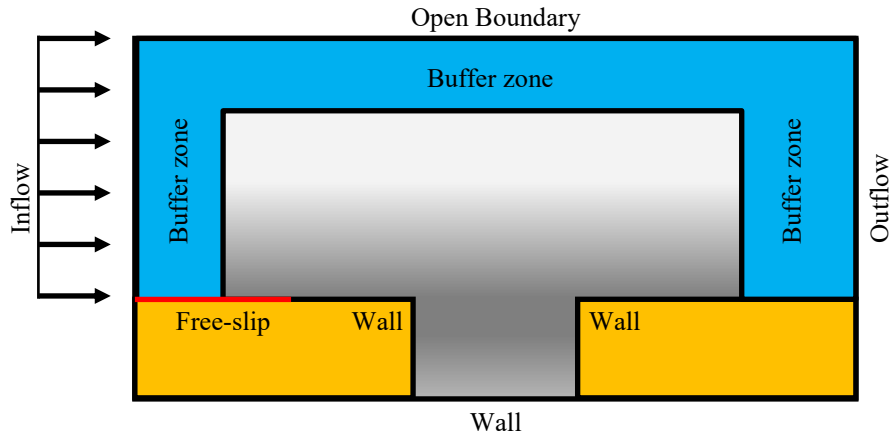

Figure 1. Illustration of the domain and its boundaries. Buffer zone is shown in blue, points inside the wall are orange, darker grays indicate a finer grid.

Compressible Navier-Stokes equations are solved for each point in the domain for density, velocity components and internal energy. Spatial derivatives are computed by a finite differences method chosen before runtime. Explicit second and fourth order methods were implemented, as well as compact fourth order spectral-like and sixth order schemes described by Lele (1992). It is possible to use a distinct differentiation method for the buffer zone, usually of lower order, the transition from one method to the other is smooth. Time integration is performed at a fixed time step by either an explicit Euler, a second order or a fourth order Runge-Kutta scheme or the Crank-Nicholson method.

Once more, both the corner nodes presented a challenge for these methods when computing the derivative. Fig. 2 illustrates the Regions considered. In a inner node, i.e. one away from the boundaries, finite differences method uses the nodes around it to approximate the derivative. If no other changes were made to the code, corners nodes 1 and 2 would be considered as inner nodes and, thus, derivatives computed for them would account for both points on the wall and points immersed in the flow, which would not be correct and cause the solution to fail. Because of this, both these nodes only consider values in Region 2 of Fig. 2.

This has solved the problem for both corner nodes and for the Region between them. But nodes in Regions 1 and 3 of the figure would still have its derivatives miscalculated, causing discontinuities in the domain close to the corner, which would later cause the simulation to fail. This was solved by considering both the last node in Region 1 and the first node in Region 3 as inner nodes for the finite differences method, which makes the derivative a continuous function again.

In sum, derivatives in Region 2 depend only on values in it. Derivatives in Regions 1 and 3 depend on values both in their respective region and on values in Region 2. The approach for wall-normal derivatives is analogous.

The initial condition is defined as free-flow velocity above the flat plate and zero velocity in the cavity, at constant pressure and temperatures. Results from previous runs can also be used as initial condition as long as they share the same mesh. Reynolds and Mach numbers and the numerical methods may be changed.

All values are non-dimensional, being normalized by the free flow speed, cavity depth and initial density. A base temperature is defined for each run.

In this type of simulation, high-frequency numerical noise may quickly build up, causing the simulation to diverge from the physical solution. To attenuate this noise, numerical low-pass filters are implemented (Gaitonde, 1998). The filter strength can be adjusted before runtime and, ideally, should be as low as possible, minimizing its effect on results. Filtering was turned off a few nodes away from the boundary due to the large derivatives present and to avoid using uncentered stencils.

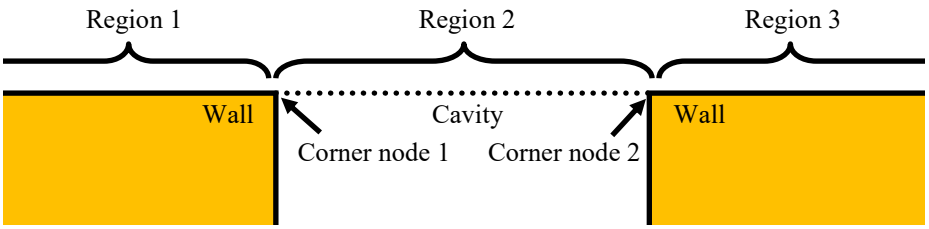

Figure 2. Close-up of the corner nodes in the domain and the Regions considered for derivatives.

So far, the implementation was done only for a two-dimensional domain, which assumes all variables as constant in the span-wise direction. This is a valid assumption for stability analysis as long as the flow conditions are such that two-dimensional modes become unstable before three-dimensional ones.

2.2 Matrix-Free Instability Analysis

In order to find the oscillation modes of a certain system, one has to compute the eigenvalues and eigenvectors of the flow Jacobian matrix. It is, given that the Navier-Stokes equations and all boundary conditions are described by

$$\frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{f}(\boldsymbol{u}) \tag{1}$$

Where $\boldsymbol{u}$ is a vector that contains all flow variables for each node in the mesh, the jacobian is given by

$$\boldsymbol{A} = \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial \boldsymbol{u}} \tag{2}$$

The size of this matrix is $4N \times 4N$ for two-dimensional flows and $5N \times 5N$ for three-dimensional. 4 and 5 are the numbers of variables for each node, respectively. N being the number of nodes in the domain. Given the domain's size is usually in the order of hundreds of points in the stream-wise and wall-normal directions and a few dozens of points in the span-wise direction, the total number of variables to solve for is in the order of 10^5 to 10^6 in two-dimensional cases and 10^6 to 10^7 when accounting for the third dimension.

Due to the implicit differentiation methods used, in which each node in the domain influences every other at all steps, the Jacobian of this system would be a full matrix. At the double precision used by this code, which takes 8 bytes of memory per value, simply storing the Jacobian for a system with 10^6 variables, for example, would take 8×10^{12} bytes, or 8 terabytes of memory, which is way out of reach for the vast majority of modern computers.

This method was proposed by Eriksson and Rizzi (1985) for the Euler equations and first implemented by Chiba (1998) to the incompressible Navier-Stokes equations. It is based on Arnoldi's method for solving the eigenproblem. The most important feature of the method is to be able to approximate the most unstable or least stable modes without storing the jacobian matrix for the flow. The DNS itself is embedded into the method instead of being linearized to the jacobian matrix. In summary, the method adds small disturbances to the steady-state flow and uses the DNS to observe how the flow reacts to them. After a series of orthogonal disturbances, the Hessenberg matrix in Arnoldi's method is obtained, which is several orders of magnitude smaller than the flow's Jacobian matrix would be. The eigenvalues and eigenvectors of this matrix are computed and, finally, used to approximate the flow's modes and their respective amplification or attenuation rates. This is shown in the flowchart in Fig. 3.

Eriksson and Rizzi use a random vector for the initial disturbance. Chiba computes it as a random vorticity field, for which the Poisson equation is solved numerically, resulting in the disturbance's stream function.

In this work, a random vector is also used, it is multiplied by a weighting function, making sure the disturbance magnitude is larger closer to the cavity and almost zero close to the boundaries, as that is the region of interest for instability modes. In addiction to this random vector, a unitary perturbation, also multiplied by the weighting function, can also be used.

In step (3) of Fig. 3, ϵ controls the norm of the perturbation, it is worth noting that the number of mesh node influences the module of the perturbation in each node. This variable should be adjusted so that the perturbation is small enough so that non-linearities remain small but should be large enough so numerical and roundoff errors influence on results are also small. The T time span in step (4) should be shorter than the half-periods of the flow's modes to avoid aliasing problems. In step (5), a second-order central finite differences scheme is used, Eriksson and Rizzi note that higher order methods may be used, this reduces the effect of non-linearities in the results but requires more DNS runs.

It should also be noted that in step (12), the equation for eigenvalues is not fully reversible for the imaginary part, as the exponential function of a complex number is not injective. This means that the method brings a correct approximation only for the real part of the eigenvalues, related to the amplification rate. The imaginary part, related to the frequency, must

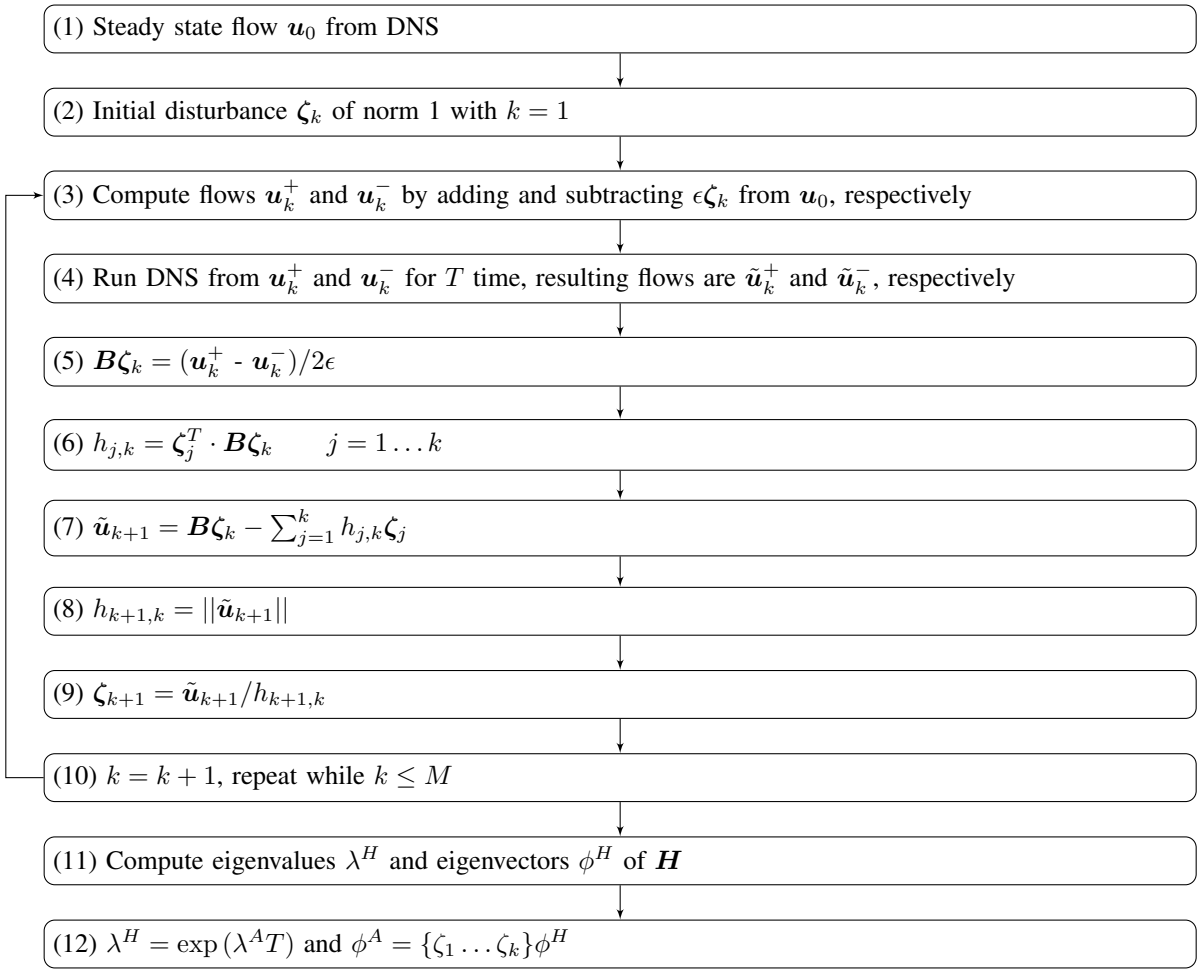

Figure 3. Flowchart of matrix-free method for computing the flow modes.

be checked by other means, for example by using the DNS to simulate each one of the modes identified and observing the oscillation frequencies.

Shift-invert strategies may be also used in conjunction with this method, as done by Gómez (2015). This allows Arnoldi's method to retrieve modes in a specific Region of the complex plane, for example, it could obtain modes related to acoustic of the flow, which may be most stable than other modes. A code for this routine was implemented in Matlab, it calls the Fortran DNS as necessary. An important perk of this method is that it can be programmed in a very modular way. The DNS could be easily replaced or updated with only minor changes to the Matlab code. All parallel processing capabilities of the DNS are retained.

3. RESULTS

3.1 Direct Numerical Simulation

The case used for code validation was extracted from the work of Colonius, Basu and Rowley (1999). The cavity's aspect ratio is $L/D = 4$, the boundary's layer thickness related to the cavity length is $L/\theta = 102$, Reynolds and Mach numbers are, respectively, $Re_\theta = 60$ and $Ma = 0.6$. The flow characteristic length is the depth of the cavity, free-flow velocity is also unitary. Contrary to the reference case, the inflow condition in the study is a uniform flow, which develops into the boundary layer downstream. Note that for this base length, the actual Reynolds number considered by the code is 1530.

The cavity position in the flat place is such that the boundary layer will have the desired thickness at the cavity's backward facing step. Linear growth theory was used to find the wall length upwind of the cavity, which begins at $x = 5.33672$ and ends at $x = 9.33672$.

These parameters result in a periodic flow. A case with a lower $Re = 300$ with the same geometry was also run, as it results in a steady state flow, suitable for instability analysis. In this case, $Re_\theta = 26.6$, approximately.

All meshes used have had the stretching parameters carefully chosen so that there are always nodes exactly over the

walls in all directions. Three meshes were used, as shown in Tab. 1.

Table 1. Meshes used for simulations.

Mesh	**1**	**2**	**3**
Nodes in regular domain	600 × 300	800 × 400	800 × 400
Domain dimension (x)	-2 to 15	-2 to 15	-4 to 20
Domain dimension (y)	-1 to 4	-1 to 4	-1 to 6
Nodes in cavity	216 × 105	288 × 140	211 × 120
Nodes in buffer zone (x/y)	50 / 50	50 / 50	50 / 50
Buffer zone stretching (x/y)	100 / 100	100 / 100	100 / 100
Differentiation scheme	4th order explicit	4th order compact	4th order compact

Both domains 1 and 2 are the same size, meshes 2 and 3 are larger than mesh 1. It is worth noting that buffer zone nodes were not counted in the mesh size shown in Tab. 1. Buffer nodes in the stream-wise direction are added twice to the domain, once in the inflow and once in the outflow boundaries. The time step should usually be chosen to be close to the largest numerically stable step, according to the CFL condition. In this case, for mesh comparison purposes, all cases have used the same value of $dt = 1.2 \times 10^{-3}$.

It was noticed that the compact, spectral-like, 4th order differentiation scheme would not converge for the coarser mesh 1, larger meshes are needed for it to work. In the buffer zones, the differentiation method had its order decreased to 2 in mesh 1 and was changed for the explicit 4th order for meshes 2 and 3. For the time stepping, the fourth order Runge-Kutta scheme was used.

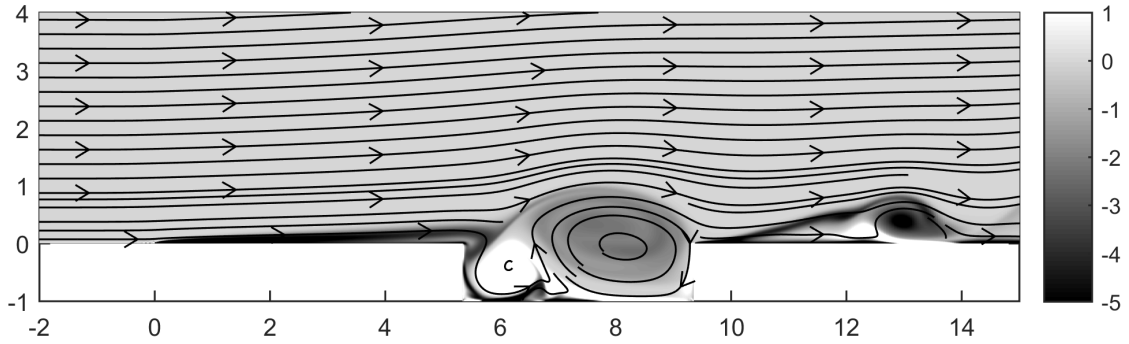

Figure 4. Streamlines and vorticity for $Re_\theta = 60$ at arbitrary time after reaching periodic flow. Mesh 2, buffer zone not shown.

Figure 5 shows the evolution of free-flow normal velocity at cavity corner height at three quarters from one edge to the other in the stream-wise direction for all three meshes, normalized by the free-flow, the reference was extracted from Colonius, Basu and Rowley (1999), the phase was manually adjusted. Data was taken from time steps 100000 to 150000.

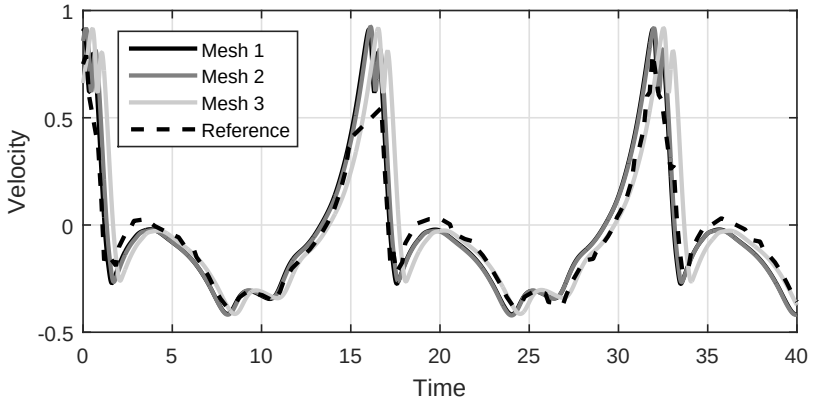

Figure 5. Freeflow-normal velocity for $Re_\theta = 60$ at a given point for all meshes after achieving a periodic flow.

All three meshes have converged to a similar solution, very close to the reference results. Even the coarser mesh 1 was able to reach satisfactory results, which means that the more expensive compact methods, which require finer grids, may not be necessary in this case.

3.2 Matrix-Free Instability Analysis

For the lower $Re = 300$ cases, both meshes 1 and 2 from Tab. 1 were used. The only change made from the previous case was the Reynolds number.

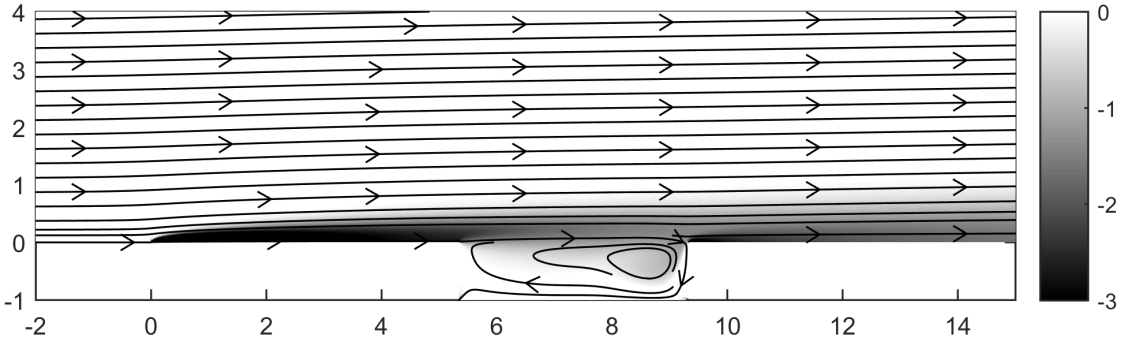

Figure 6. Streamlines and vorticity for $Re = 300$ after reaching steady state flow. Buffer zone not shown.

Figure 7 shows modes 1 and 3 found for this flow. The algorithm was run on mesh 2, for 150 Krilov iterations, each using 100 time steps in te DNS, the initial perturbation was in the order of 10^{-5}. Note that mode 2 is the complex conjugate of mode 1. Modes are numbered from most unstable to most stable.

In this case, the perturbation was a vector of ones multiplied by the weighting function, to concentrate disturbance energy close to the cavity. It was noted that without this weighting function, the algorithm would find modes closer to the boundaries, inside the buffer zones, which are most likely numeric and outside the scope of this work.

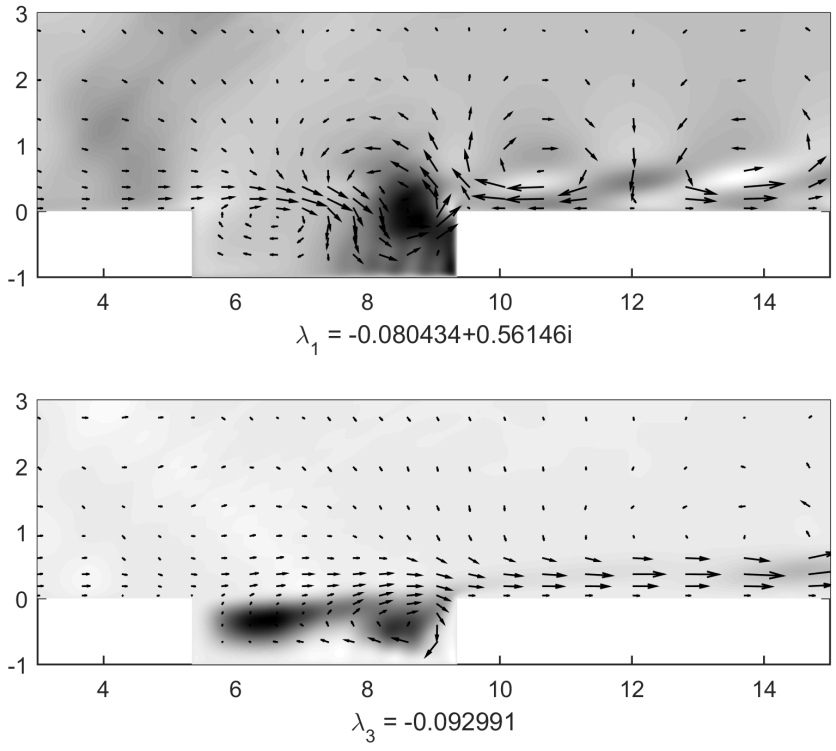

Figure 7. Modes 1 and 3 found with mesh 2. Shades of gray indicate pressure fluctuations (brighter is positive) and vector field shows velocity fluctuations.

Both modes are stable, as can be verified by the negative real part of the eigenvalue. Note that, as discussed in the methods section, the imaginary part shown may not be correct and would require further checking.

Figure 8 compares results for a different number of Krilov iterations in the Arnoldi method. The upper plot is for 50 iterations and the bottom plot, for 100. Both show mode 1 with mesh 2. Once more, 100 time steps were used for each Krilov iteration.

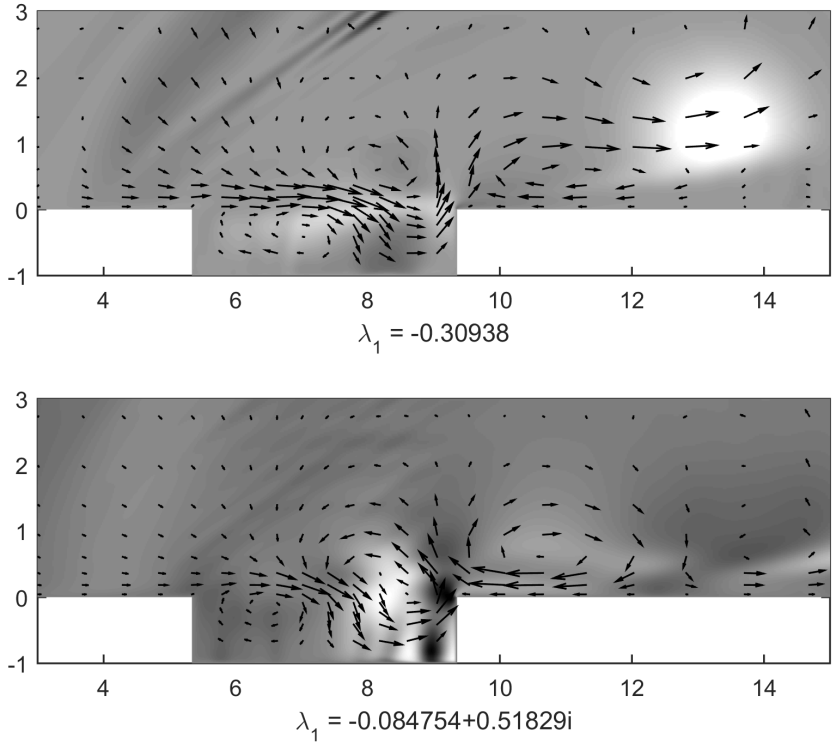

Figure 8. Mode 1 for mesh 2 with 50 (top) and 100 (bottom) Krilov iterations. Shades of gray indicate pressure fluctuations (brighter is positive) and vector field shows velocity fluctuations.

It can be seen that the results for 100 and 150 iterations are visually close for the velocity field but not for the pressure fluctuations, the most unstable eigenvalue has changed about 5%. For 50 iterations, the result is considerably different and, thus, can be considered not converged.

Figure 9 is the same as Fig 7 but with mesh 1. Once more, 150 Krilov iterations were used, each running the DNS for 100 time steps.

Comparing both results, it can be seen that mode 1 has switched with modes 2 and 3 when the mesh changed. The modes became less stable, but kept the qualitative characteristics. Further study is needed to determine how much finer the mesh needs to be and what are the Arnoldi method parameters to numerically converge the solutions and produce reliable results.

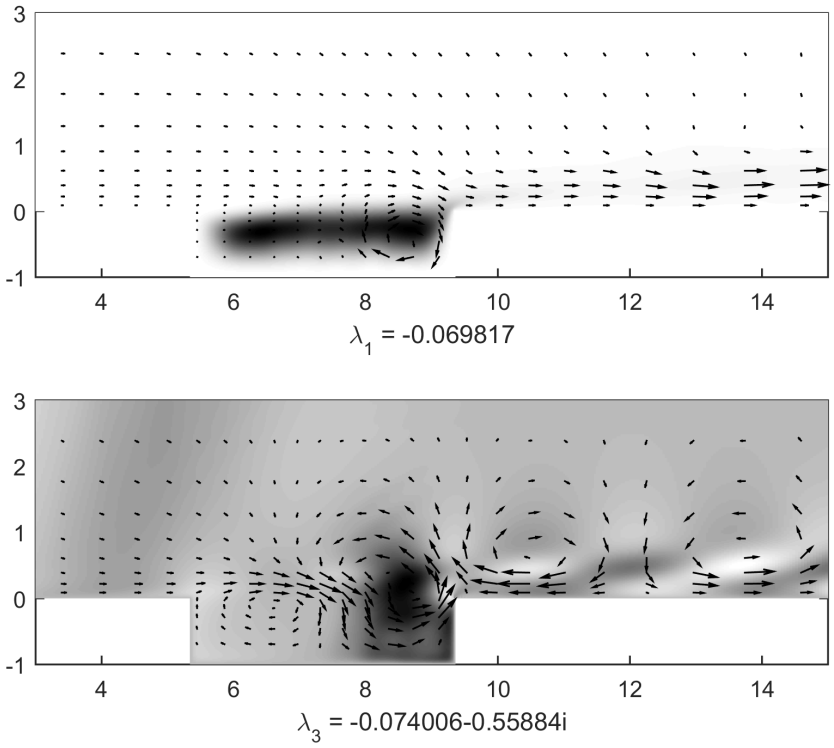

Figure 9. Modes 1 and 3 for mesh 1. Shades of gray indicate density fluctuations (brighter is positive) and vector field shows velocity fluctuations.

4. DISCUSSION AND NEXT STEPS

This paper has presented methods to simulate a two-dimensional flow over an open cavity and to identify its most unstable modes. Both algorithms take from some minutes to a few hours to run in modern computers, allowing multiple cases to be run with a range of different parameters.

Results and interesting points obtained from the two-dimensional analysis can be used on larger, three-dimensional runs, for which the code can be adapted without major changes. For example, cases with stable two-dimensional modes but unstable three-dimensional modes can be solved in two-dimensions to obtain a base flow and passed to the three-dimensional solver to observe the unstable modes.

High order differentiation methods can be used while checking for flow modes, but care has to be taken because these methods are very sensitive to disturbances and their numerical instabilities can easily be found by the matrix-free analysis instead of the actual flow instabilities if the mesh is not fine enough.

The buffer zone and the filters will be adjusted to remove all non-physical oscillations in the domain, allowing for a steady flow to be found at a much higher precision than the current results have reached. This will also allow much smaller perturbations to be used by the matrix-free method, reducing the risk of non-linear effects contamination in the results.

Finally, shift-invert strategies may be implemented in the instability analysis so specific regions of the eigenvalue plane can be obtained, this could be used, for example, if acoustic modes are to be found.

The matrix-free instability analysis method presented is very capable of finding the modes in a flow, but it depends on a very reliable DNS code, low in dispersion and diffusion as these characteristics would directly influence the results. It also introduces new parameters that need to be checked for numerical convergence, such as number of Krilov iterations and magnitude and type of initial perturbation.

5. ACKNOWLEDGEMENTS

The authors would like to thank the National Council for Scientific and Technological Development (CNPq), the São Paulo Research Foundation (FAPESP) and the Air Force Office of Scientific Research (AFOSR) under Grant No.FA9550-11-1-0354-P00002 for their support.

6. REFERENCES

Arnoldi, W. E., 1951 "The principle of minimized iterations in the solution of the matrix eigenvalue problem", Quarterly of Applied Mathematics 9, pp. 17-29.

Brès, Guillaume A.; Colonius, Tim, 2008 "Three-Dimensional Instabilities in Compressible Flow over Open Cavities", Journal of Fluid Mechanics 599, http://www.journals.cambridge.org/abstract_S0022112007009925

Chiba, S., 1998 "Global Stability Analysis of Incompressible Viscous Flow", Journal of Japan Society of Computational Fluid Dynamics 7, pp. 20-48.

Colonius, Tim; Basu, Amit J.; Rowley, Clarence W., 1999 "Computation of Sound Generation and Flow-Acoustic Instabilities in the Flow Past an Open Cavity" Proceedings of the Joint Fluids Engineering Conference, San Francisco, USA

Colonius, Tim, 2001 "An Overview of Simulation, Modeling, and Active Control of Flow/acoustic Resonance in Open Cavities", 39th Aerospace Sciences Meeting and Exhibit, pp. 1-12, http://arc.aiaa.org/doi/abs/10.2514/6.2001-76

Eriksson, Lars E.; Arthur Rizzi, 1985 "Computer-Aided Analysis of the Convergence to Steady State of Discrete Approximations to the Euler Equations", Journal of Computational Physics 57, pp. 90-128.

Gaitonde, Datta V.; Miguel R. Visbal, 1998 "High-Order Schemes for Navier-Stokes Equations: Algorithm and Implementation Into FDL3DI", Wright-Patterson Air Force Base

Gómez, Francisco; Pérez, José Miguel; Blackburn, Hugh M.; Theofilis, Vassilios, 2015 "On the Use of Matrix-Free Shift-Invert Strategies for Global Flow Instability Analysis" Aerospace Science and Technology 44, pp. 69-76, http://linkinghub.elsevier.com/retrieve/pii/S1270963814002284

Lele, Sanjiva K., 1992 "Compact Finite Difference Schemes with Spectral-like Resolution", Journal of Computational Physics 103, pp. 16-42.

Tezuka, Asei; Suzuki, Kojiro, 2006 "Three-Dimensional Global Linear Stability Analysis of Flow around a Spheroid", AIAA journal 44.8, pp. 1697-1708, http://arc.aiaa.org/doi/pdf/10.2514/1.16632

Theofilis, Vassilios, 2011 "Global Linear Instability", Annual Review of Fluid Mechanics 43, pp. 319-352, http://www.annualreviews.org/doi/suppl/10.1146/annurev-fluid-122109-160705

7. RESPONSIBILITY NOTICE

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