Coupling time-stepping numerical methods and standard aerodynamics codes for instability analysis of flows in complex geometries

F. Gómez\textsuperscript{1,*}, R. Gómez\textsuperscript{1}, V. Theofilis\textsuperscript{2,†}

\textsuperscript{1} INTA Instituto Nacional de Técnica Aeroespacial, Ctra. Ajalvir Km.4 Torrejon de Ardoz, E-28850 Madrid, Spain
\textsuperscript{2}School of Aeronautics, Universidad Politécnica de Madrid, Plaza Cardenal Cisneros 3, E-28040 Madrid, Spain

Abstract

The development of a global instability analysis code coupling a time-stepping approach, as applied to the solution of BiGlobal and TriGlobal instability analysis\textsuperscript{1,2} and finite-volume-based spatial discretization, as used in standard aerodynamics codes is presented. The key advantage of the time-stepping method over matrix-formulation approaches is that the former provides a solution to the computer-storage issues associated with the latter methodology. To-date both approaches are successfully in use to analyze instability in complex geometries, although their relative advantages have never been quantified. The ultimate goal of the present work is to address this issue in the context of spatial discretization schemes typically used in industry. The time-stepping approach of Chiba\textsuperscript{3} has been implemented in conjunction with two direct numerical simulation algorithms, one based on the typically-used in this context high-order method and another based on low-order methods representative of those in common use in industry. The two codes have been validated with solutions of the BiGlobal EVP and it has been showed that small errors in the base flow do not have affect significantly the results. As a result, a three-dimensional compressible unsteady second-order code for global linear stability has been successfully developed based on finite-volume spatial discretization and time-stepping method with the ability to study complex geometries by means of unstructured and hybrid meshes.

I. Introduction

Investigation of instability mechanisms is essential for the understanding of the transition process from laminar to turbulent flow. Linear instability theory, in either its modal or non-modal flavor, provides insight in these instability mechanisms in a large number of fluid dynamics problems. This theory is based on decomposing the flow into a steady or unsteady laminar part and an infinitesimal unsteady part, the so-called basic and perturbation flow, respectively. If a basic flow is established, the Navier-Stokes equations can be written in terms of disturbance variables and linearized for small-amplitude disturbances. A large number of studies over several decades have reported results in simple one-dimensional basic flows, such as boundary layers or shear layers. However, most flows of practical engineering significance remain unexplored. The reason is that the underlying basic state of most practical flows depends in an inhomogeneous manner on more than one spatial direction and the cost of performing a complete parametric instability analysis can be formidable when the matrix discretizing the EVP is formed and stored.\textsuperscript{4} In principle, the assumptions underlying BiGlobal\textsuperscript{1,2} linear instability lead to a problem easier to solve numerically than the direct
numerical simulation (DNS), but the large size of the discretized matrices makes the numerical solution challenging. The most effective techniques to solve the resulting generalized eigenproblem are based on subspace projection-iterative methods such as the Arnoldi iteration which is based on the Krylov-subspaces. The Arnoldi method delivers a window of the eigenspectrum, but it favors the eigenvalues with the largest modulus, thus inversion of the matrix is required in order to introduce an eigenvalue shift towards the interested part of spectrum. This technique was first introduced in fluid mechanics by Natarajan and Acrivos and used by many others.

On the other hand, TriGlobal linear stability problems, when using a matrix-forming approach, add a new inhomogeneous spatial direction that could lead to prohibitively expensive computing requirements when the matrix is formed, stored and inverted. Time-stepping (TS) approaches can provide one solution for this class of problems using a Jacobian-free methodology. The key advantage of time-marching methods, over explicit formation of the matrix which describes linear instability, is that the matrix need never be formed. This enables the study of global linear stability problems on small-main-memory machines at the expense of long time integrations. To date this has been the only viable approach to perform TriGlobal instability analysis. A potential pitfall of the time-integration approach is that results are sensitive to the quality of spatial integration of the linearized equations, such that this approach should preferably be used in conjunction with high-order spatial discretization methods that provide fast convergence; this problem has been discussed by Tuckerman and Theofilis et al. where preconditioners are proposed in order to avoid long integration times due to the temporal exponential decay of the perturbations. A rather complete discussion of time-stepping has recently been presented by Barkley, Blackburn and Sherwin. However, there are no conclusions in the literature about the required order of the numerical scheme for successful eigenspectrum calculations. The first successful time-stepping methodology was introduced in Fluid Mechanics by Eriksson & Rizzi, where a numerical differentiation of the DNS was used along a temporal polynomial approximation. In that work, finite differences were used in order to study an inviscid incompressible flow over a NACA airfoil. Chiba improved the Erikson approach by introducing a temporal exponential transformation and using the Navier-Stokes equations. Following this method, Tezuka and Suzuki successfully solved the first TriGlobal problem. Earlier to this, Edwards et al. developed a similar time-stepping methodology in conjunction with the linearized Navier-Stokes equations, which have been successfully used by Barkley et al. especially for the analysis of time-periodic flows. Mack & Schmid contributed to both approaches by introducing another JFNK using different spectrum transformations and preconditioners in the study of compressible flows with high-order compact finite differences. Other time-stepping methods make use of the Snapshots method, first introduced in fluid mechanics by Sirovich and later used in global stability analysis by Bagheri et al.. Despite second-order methods have been successfully used in a global linear instability theory in a matrix-forming context none has been used in a time-stepping context. Therefore, the first objective of this paper is to develop a time-stepping code coupling JFNK methods with a flexible finite volume standard aerodynamic code. Finite volume (FV) discretization has been selected due to its flexibility. Then, the second objective of this paper is to provide insight in the spatial discretization issue by employing different time-stepping schemes, based on commonly available spatial integration techniques: (high-order) spectral collocation and (second order) FV techniques. The methodology here is first to implement the different DNS into a time-stepping approach and second to compare the results with known BiGlobal EVP results. The rest of this paper is organized as follows. First, in Section II, linear stability theory in a time-stepping context is discussed. Next, in Section III the most important characteristics of the DNS codes used in the time-stepping are highlighted. Then, in Section IV results are discussed, and conclusions are obtained in Section V.

II. Theory

II.A. TriGlobal Linear Stability Theory

Linear stability theory is concerned with the evolution of a small amplitude disturbances superimposed upon a basic state or "base flow". Following Theofilis formulation for steady basic flows, the TriGlobal linear stability is decomposed in a base flow and three-dimensional amplitude function of the unsteady small perturbations. According to the TriGlobal Ansatz

---

*See the discussion on this point by Theofilis & Le Clainche in this conference*
\[ q(x, y, z, t) = \bar{q}(x, y, z) + \epsilon \hat{q}(x, y, z)e^{-i\Omega t} \]  

where \( \epsilon \ll 1, \Omega = \Omega_r + \Omega_i \), with \( \Omega_r \) representing a frequency and \( \Omega_i \) being the amplification/damping rate of the disturbance sought, while barred and hatted quantities denote basic and disturbance flow quantities, respectively.

The three-dimensional and dimensionless Navier-Stokes equations of a viscous, incompressible fluid in Cartesian coordinates can be written as

\[ \nabla u = 0 \]  
\[ \frac{\partial u}{\partial t} + u \nabla u = -\nabla p + \frac{1}{Re} \nabla^2 u \]  
and be summarized as:

\[ \frac{\partial u}{\partial t} = f(u) \]

where \( u \) contains the (divergence-free) three velocity components in the computational nodes.

A linearized eigenvalue problem can be written taking the TriGlobal Ansatz into the Navier-Stokes equation, retaining the infinitesimal terms:

\[ \frac{\partial \hat{u}}{\partial t} = \frac{\partial f(\bar{u})}{\partial u} \hat{u} \equiv A \hat{u}. \]

Taking the eigenvalues \( \mu_i \) and eigenvector \( \tilde{y}_i \) of the Jacobian matrix \( A \) into equation (5), it can be written that:

\[ \frac{\partial \tilde{y}_i}{\partial t} = \mu_i \tilde{y}_i \]

Finally, the evolution in time of the perturbation can be written by a linear combination of its eigenvectors,

\[ \hat{u}(t) = \sum_{i=1}^{n} \kappa_i \tilde{y}_i e^{\mu_i t}. \]

II.B. The time-stepping approach

Time-stepping techniques are application of the Jacobian-free Newton-Krylov (JFNK) methods with temporal spectrum transformation. These JFNK methods are a combination of Newton-type methods for solving a set of nonlinear equations and Krylov subspace methods for solving the eigenvalue problems. The strength of this class of methods is that it is not required to form a Jacobian and only a Jacobian-vector product is required. This Jacobian-vector product is usually formed by numerical differentiation. Knoll and Keyes\textsuperscript{27} surveyed in deep the capabilities of these methods.

II.B.1. Newton method

The Newton equation is derived by applying a Taylor series expansion to equation (4) around the flow base \( \bar{u} \) and considering that the second and higher order terms with respect to \( \bar{u} \) are negligibly small.

\[ f(\bar{u} + \epsilon \hat{u}) = f(\bar{u}) + \frac{\partial f(\bar{u})}{\partial u} \epsilon \hat{u} + O(\epsilon^2) \]

The Jacobian-vector product can be then obtained means of:

\[ \frac{\partial f(u)}{\partial u} \hat{u} \approx \frac{f(\bar{u} + \epsilon \hat{u}) - f(\bar{u})}{\epsilon} \]

This equation is also known as a Fréchet derivative\textsuperscript{28} and it numerically differentiates the equation system. In addition it is possible to create high order Fréchet derivative by increasing the stencil. It is direct to form an eigenvalue problem from the Fréchet derivative:

\[ \frac{\partial \hat{u}}{\partial \epsilon} = \frac{\partial f(\bar{u})}{\partial u} \hat{u} \equiv A \hat{u} \]
II.B.2. Krylov subspace iteration

The use of a Krylov subspace iteration is the second part of the JFNK methods. In particular, the Arnoldi\textsuperscript{5,6} method makes use of the Jacobian-vector product obtained with the Fréchet derivative in order to get a finite but small number of eigenvalues (equal to the Krylov subspace dimension) \( m \). The Arnoldi algorithm works as follows:

1. Choose an initial random vector \( v_1 \) and normalize it.
2. \( \text{do } j = 1, 2...m \)
3. Calculate \( w_j \) as \( Av_j = w_j \).
4. \( \text{do } i = 1, 2...j \)
   \[
   h_{ij} = (Av_j, v_i),
   \]
   \[
   a = \sum_{i=1}^{j} h_{ij} v_i,
   \]
   \[
   \hat{v}_{j+1} = w_j - a,
   \]
   \[
   h_{j+1,j} = \|\hat{v}_{j+1}\|,
   \]
   \[
   v_{j+1} = \frac{\hat{v}_{j+1}}{h_{j+1,j}}
   \]
   \( \text{endo} \)
   \( \text{endo} \)

This algorithm delivers an orthonormal basis \( V_m = [v_1, v_2, ..., v_m] \) of the Krylov subspace \( K_m = \text{span}\{v_1, Av_1, ..., A^{m-1}v_1\} \). The restriction from \( A \) to \( K_m \) is represented by the matrix \( H_m = \{h_{ij}\} \). The eigenvalues of the latter matrix are an approximation of the \( m \) largest eigenvalues of the original problem \( \frac{\partial \hat{u}}{\partial t} = A \hat{u} \). The eigenvectors associated with these eigenvalues may be obtained from

\[
\hat{q}_i = V_m \hat{y}_i
\]

where \( \hat{y}_i \) is an eigenvector of \( H_m \) associated with the \( \mu_i \)-th eigenvalue.

The total time needed for a complete Arnoldi analysis depends mostly on the efficiency of the linear solver described above, as well as on the Krylov space dimension \( m \) used to approximate the most important eigenvalues. This method is commonly employed together with the shift-and-invert strategy for BiGlobal problems. In addition, Theofilis\textsuperscript{1} recently listed the available spectrum transformations.

In our case, in order to find the most interesting part of the spectrum, a temporal transformation will be used next.

III. Numerical Method

III.A. Temporal Spectral Transformation

As mentioned, a temporal transformation is required to obtain the most interesting part of the spectrum. By integrating equation (5) in time, the time evolution of the perturbations is obtained

\[
\hat{u}(t) = \hat{u}(0)e^{At},
\]

which permits calculation of the inner part of the \( A \) spectrum using the transformation \( B = e^{At} \) and the Arnoldi method. Moler et al\textsuperscript{29} discuss this matrix exponential techniques in their work and provide several different approaches. In figure (1), this exponential transformation is schematically presented. A spectrum is shown on the left and its transformed spectrum \( B \) is shown on the right. All eigenvalues are moved into a unit circle and small eigenvalues (circle) becomes the largest eigenvalue in the \( B \) spectrum. Large eigenvalues in \( A \) (triangle, square) move to the unit circle origin in \( B \).
III.B. The full time-stepping algorithm

In this paper, we will only follow Chiba’s\(^3\) approach, which have been successfully used by Tezuka and Suzuki\(^{12,13}\).

1. Choose an initial random vector \( \zeta_1 \) and normalize it.
2. do \( j = 1, 2 \ldots m \)
3. Integrate the equations over \( \tau \) time with perturbation as initial value \( u_{j+} = \bar{u}(t) + \epsilon \zeta_j(\tau) \) and \( u_{j-} = \bar{u}(\tau) - \epsilon \zeta_j(\tau) \)
4. Form \( B\zeta_j = \frac{u_{j+}-u_{j-}}{2\epsilon} \)
5. do \( i = 1, 2 \ldots j \)
   \[
   h_{ij} = \zeta_i^T B\zeta_j, \\
   a = \sum_{i=1}^{j} h_{ij} \zeta_i, \\
   \zeta_{j+1} = B\zeta_j - a, \\
   h_{j+1,j} = \|\zeta_{j+1}\|, \\
   \zeta_{j+1} = \frac{\zeta_{j+1}}{h_{j+1,j}}
   \]
enddo
endo do

6. Following the Arnoldi algorithm, the eigenvectors and eigenvalues from matrix \( B \) are calculated from \( H \), and then the eigenvalues \( \mu^A \) belonging to matrix \( A \) are calculated with the relation:
   \[
   \mu^B = e^{\mu^A \tau}
   \]
7. which leads to:
   \[
   \mu^A = \frac{\log|\mu^B| + i\arg(\mu^B)}{\tau}
   \]
8. Finally, the eigenvectors \( \tilde{y}_i^A \) are the same as for matrix \( B \):
   \[
   \tilde{y}_i^A = \tilde{y}_i^B
   \]

III.C. The two DNS utilized and compared

As it has been mentioned, the quality of the DNS play a crucial role in the time-stepping approach because of the time integration of the evolution of the perturbations at every Arnoldi step. In this paper, two different DNS solvers have been used: a high-order spectral method and a second order finite volume method.

III.C.1. Spectral Collocation Method

A spectral collocation technique have been used in this DNS.\(^{30}\) Using this methodology computing the incompressible Navier-Stokes steady problem is reduced to the time-advancement of a linear system problem where the explicit terms are updated in every time step. The differentiation matrix is based on a rectangular Jacobi-Gauss-Lobatto grid defined by

\[
\mathcal{D} = [d^1_{i,j}], \quad d^1_{i,j} = h^1_j(x_i)
\]
The expression of the coefficients \( d_{i,j} \) can be found in Canuto\(^\text{30}\) or Boyd\(^\text{31}\)

\[
(d^1)_{i,j} = \begin{cases} 
\frac{2}{\zeta_j (z_i - z_j)} & \text{if } 0 \leq i, j \leq N, i \neq j \\
\frac{2}{(1 - z_i^2)} & \text{if } 1 \leq i = j \leq N, i \neq j \\
\frac{2N^2 + 1}{6} & \text{if } 0 = i = j \\
\frac{2N^2 + 1}{6} & \text{if } i = j = N
\end{cases}
\] (27)

The way to define the differential matrices in 2-D is to use tensor products, also known as Kronecker product denoted by \( \otimes \). If \( D_x^1 \) represents the \((N_x + 1)^2\) first derivative matrix in \( x \) and \( D_y^1 \) the \((N_y + 1)^2\) first derivative matrix in \( y \) direction, then \( D_x^{(1)} = I \otimes D_x^1 \) and \( D_y^{(1)} = D_y^1 \otimes I \) where \( I \) is the \((N_y + 1)^2\) or \((N_x + 1)^2\) respectively identity matrix, are the \([(N_x + 1) \times (N_y + 1)]^2\) 2-D derivatives matrices in \( x \) and \( y \) direction.

Using the spectral collocation method as spatial scheme, the chosen temporal scheme is the SMR proposed by Spalart et al.\(^\text{32}\) This SMR algorithm may be written in compact form as

\[
q''' = q'' + \Delta t \left\{ \mathcal{L}(q'''' + \lambda q''') + \mu N(q''') + \nu N(q''') \right\},
\] (28)

where the superscript denotes fractional time–step, \( \mathcal{L}(q) \) and \( N(q) \) are, respectively, the linear and nonlinear operators in the problem to be solved and \( \Delta t \) is the time–step. The rationale behind the derivation as well as sample values of the constants \( \kappa, \lambda, \mu \) and \( \nu \) of a self–starting algorithm may be found in the original reference.\(^\text{32}\) Explicitly, the operators are

\[
\mathcal{L} = \frac{1}{Re} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \zeta - \lambda f(x)(\zeta - Z)
\] (29)

\[
\mathcal{N} = -(\psi_y \zeta_x - \psi_x \zeta_y).
\] (30)

Applying (28) delivers the problem to be solved for \( (\zeta, \psi) \) at each fractional time–step,

\[
\mathcal{M}_1 \psi''' + \zeta''' = 0,
\] (31)

\[
\mathcal{M}_2 \zeta''' = \mathcal{R}_2.
\] (32)

Here

\[
\mathcal{M}_1 = \partial_{xx} + \partial_{yy},
\] (33)

\[
\mathcal{M}_2 = \partial_{xx} + \partial_{yy} - Re \left( \frac{1}{\lambda \Delta t} + \lambda f \right),
\] (34)

\[
\mathcal{R}_2 = -Re \lambda f \left( 1 + \frac{\kappa}{\lambda} \right) Z - \frac{\kappa}{\lambda} \left[ \zeta'' + \psi_{yy} + Re \left( \frac{1}{\kappa \Delta t} - \lambda f \right) \zeta'' \right] + \frac{\mu Re}{\lambda} \left( \psi_y \zeta_x - \psi_x \zeta_y \right) + \frac{\nu Re}{\lambda} \left( \psi_y \zeta_x - \psi_x \zeta_y \right),
\] (35)

After applying proper boundary conditions, these two equations can be written as:

\[
\begin{pmatrix} \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + c \end{pmatrix} q = RHS
\] (36)

where the right hand side contains the independent term form boundary conditions. Because the constant time step, the derivative matrix can be diagonalized as:

\[
M^{-1} \mu_i \mathcal{M} q + q N^{-1} \nu_i N q + cq = RHS
\] (37)
where $M$ and $N$ contains the eigenvectors and $\mu$ and $\nu$ are the corresponding eigenvalues. By multiplying by $M$ and post-multiplying by $N^{-1}$ and introducing the transformed variable $\hat{q} = MqN^{-1}$, the equations can be written in a set of algebraic equations as:

$$
\hat{q}_{ij} = \frac{RHS}{\mu_{ii} + \nu_{ii} + c}
$$

(38)

### III.C.2. BERTA Code: Finite Volume Method

BERTA\textsuperscript{33,34} is a Finite Volume solver developed at the National Institute for Aerospace Technology "Esteban Terradas" (INTA). The code presents an edge-based data structure, which allows to get numerical solutions of compressible Euler and Navier-Stokes equations independently of the selected mesh type to discretize the computational domain. Based on this data structure a complete solution technique has been formulated, which allows to handle structured grids, block structured grids, and unstructured grids of tetrahedra or mixed elements without any modification. This methodology provides to the code a grid-transparent property. Despite the use of mixed elements in order to build the mesh in a discrete domain is not new, the advantage of this code lies in the possibility of directly extract the needed information to solve the equations from the geometric data of the mesh. This is achieved by means of a pre-process where a new linear structure is build. Allocating original geometric data to the mesh edges, subsequent references to the original grid are not needed. For this purpose, a efficient data structure has been constructed, minimizing memory overhead and minimizing amount of gather/scatter, in comparison to other structures (element-based or face-based). The efficient data structure is achieved by means of the calculation of the the residual terms by loops on edges. The flux on each edge is only calculated once, and added and subtracted to the control volume of the two nodes corresponding to that edge. This way the conservative characteristic of the numerical methods is preserved. This data structure is presented as an alternative to the classic data structure based on grid cells. In this classic data structure, the information is obtained and computed in every cell, and then the information is sent to every vertex. On the other hand, the edge-based algorithm obtains the information from the nodes of each edge, operates in the edge, and returns the information to each node of the edge. The number of memory access operations are minimized with data structure based on edges. For example, in a tetrahedral mesh with $N$ nodes, the number of cells $N_{\text{cell}}$ is $5.5N$, and the number of edges $N_{\text{edge}}$ is $7N$. The classic data structure based on cell requires $2 \times 4 \times N_{\text{cell}} = 44N$ operations of memory access operations while for a edge-based data structure is required $2 \times 2 \times N_{\text{edge}} = 28N$ memory access operations Therefore, the code efficiency has been significantly improved. As it has been said, a finite-volume schemes, when the flow variables are stored at the vertices of the mesh, has been used for the discretization of the equations. The basic spatial discretization is formed using a central difference finite-volume scheme with added artificial dissipation. The resulting discretized equations form a set coupled ordinary differential equations which are integrated in time using a multi-stage time-stepping scheme. In addition, convergence to a steady flow can be accelerated by local time-stepping and implicit residual averaging. It has been fully validated\textsuperscript{34} for both internal and external flows.

### IV. Results

#### IV.A. Validation case

First, the instability of a two dimensional regularized lid-driven cavity (LDC)\textsuperscript{35–37} has been analyzed using the three different methodologies, the finite volume method (FVM), the spectral collocation method (SCM), both in a TS framework, and the solution of the BiGlobal EVP (BG). Since the eigenvalue spectrum is known in this problem from BG results, a parameter sweep has been carried out in order to identify the optimum parameters for its recovery in a TS framework. The determining parameters in this exercise have been found to be the order of magnitude of the initial perturbation, together with the time integration length within each step of the Arnoldi algorithm. In particular, this two-dimensional physical problem is stable against both two- and three-dimensional perturbations at $Re = 200$, and due to the strong exponential decay, a compromise between initial perturbation magnitude and final time integration must be reached.
IV.A.1. Analysis Requirements

The parameters of the time-stepping approach are the Krylov subspace dimension \( m \), the integration time \( \tau \), the perturbation size control parameter \( \epsilon \) and the mesh size, \( n \) for spectral collocation method and minimum mesh size \( \Delta x \) and number of nodes \( n_{\text{nodes}} \) for the finite volume method.

As a basic rule of thumb, the integration time \( \tau \) must be long enough in order to correctly perform the exponential transformation, but not too long, so the linear perturbations remain up to an order of magnitude bigger than the numerical error. In agreement with Goldhirsch et. al., following this idea, it has been found that one of the required conditions in order to obtain a correct \( i \)-th eigenvalue is:

\[
|\text{Re}(\lambda_i - \lambda_m)| \tau \gg 1,
\]

where \( \tau \) is the time integration length and \( \lambda_m \) is the the \( m \)-th eigenvalue.

If the spectrum of the problem presents significant gaps, small Krylov spaces can lead to good results. In particular, for the first eigenvalue of this problem it is obtained that \( |\text{Re}(\lambda_1 - \lambda_2)| \tau \sim 16 \). Krylov subspace dimensions from \( 10 \leq m \leq 4000 \) have been explored in order to recover different sizes of the spectrum.

In addition, and in agreement with Tezuka & Suzuki, if a second-order Fréchet differentiation is used, the perturbation magnitude control parameter \( \epsilon \) do not have a significant influence on the results, as long as the perturbation preserve a linear behavior and a larger magnitude than the roundoff error. This can be achieved by means of controlling the perturbation size at each Arnoldi step, for example using the variable \( \epsilon \) equation (40) proposed by Eriksson & Rizzi or carefully choosing an initial perturbation and a constant \( \epsilon \).

\[
\epsilon = \epsilon_0 \times \frac{\mathbf{u}}{\hat{\mathbf{u}}}
\]  

(40)

Eriksson recommends \( \epsilon_0 = 0.001 \) using fourth-order accurate numerical differentiation, while Mack & Schmid use \( \epsilon_0 = 10^{-8} \) with a first-order numerical variation. A value of \( \epsilon_0 \) between \( 10^{-6} \) and \( 10^{-8} \) seems proper for our calculations. In the compressible case, due to the different order of magnitude between the different non-dimensional variables, it has been carefully chosen a constant \( \epsilon \) and an initial condition. Tezuka recommends a constant \( \epsilon \) between 0.01 and 1.

Analyses were performed using \( 48 \times 48, 64 \times 64 \) and \( 80 \times 80 \) spectral collocation meshes both in the BiGlobal and SCM analysis, which are known to be adequate at this Reynolds number value. Unstructured low and high quality meshes were used in the FVM approach.

IV.A.2. Validation results

In figure (2) the obtained spectrum with three different methods can be seen. Numbers have been assigned to the eigenvalues to aid their classification. For the BG a 64x64 resolution mesh and Krylov subspace \( m = 300 \) have been used. SCM results have been obtained with a 64x64 mesh and the FVM with a mesh \( \Delta x = 10^{-3} \) and number of nodes \( n_{\text{nodes}} = 900 \). Time-stepping parameters Krylov subspace \( m = 60 \), non-dimensional integration time \( \tau = 10 \) and variable \( \epsilon \) for the SCM and a constant \( \epsilon = 10^{-2} \) for the FVM results.

The most relevant aspect of the spectrum is that, despite the convergence of the TS spectrum has been reached, only a few correct eigenvalues are recovered because of the relatively small Krylov subspace dimension. If more eigenvectors are to be recovered, the Krylov subspace dimension should be increased. In this case, the two less stable eigenvectors are correctly recovered. The eigenvectors corresponding to \( \lambda_1 = 0.3322 \) and \( \lambda_3 = 0.5437 \) can be seen in figures (3) and figure (4), respectively. The different isolines accurately match in both eigenvectors, except for the corresponding to \( u = 0 \) or \( v = 0 \), where the numerical error due to the numerical scheme are more visible. It is remarkable that, despite the problem being two-dimensional, there still exist three-dimensional \( \mathbf{\hat{w}} \) eigenvector component, for example associated with \( \lambda_2 = 0.3979 \). This is explained by the fact that the BiGlobal analysis implies a \( \beta \sim 0 \) and the FVM code handles the two-dimensional problems by means of periodic boundary conditions in the \( z \) coordinate. Obviously, this does not occur in the pure two-dimensional spectral collocation method code. The eigenvectors corresponding to \( \lambda_5 = 0.4627 \pm 0.10987i \) and \( \lambda_6 = 0.0144 \pm 0.5503i \) can be seen in figure (5). These three-dimensional modes do not need to perfectly match, since the \( \beta \) factor is similar to both codes, but not the same. Another interesting observation is that the base flow can be recovered as a global mode in some cases with \( \lambda_4 \sim 0 \). The FVM spectrum presents additional eigenvalues compared to those recovered by the BG and SCM methods. These additional modes appear due to the compressibility of the FVM method. For example, the eigenvector corresponding to \( \lambda_7 = 0.23218 \) can be seen in figure (6) and it clearly represents a global mode in density.
due to the compressibility. As observed by Tezuka\textsuperscript{13}, another important issue is that, because the complex logarithm is a multi-evaluated function, an aliasing problem may occur. Equation (24) can also be written as:

\[ \mu_A = \log|\mu_B| + i(\arg(\mu_B) + 2\pi n) \frac{\tau}{\tau} \]  

(41)

being \( n \) any natural number. This means that the frequencies of the modes cannot be correctly recovered without additional help from the DNS. This explains why both the FVM and SCM complex eigenvalues only matches the real part of BG results in some cases.

\textbf{IV.A.3. Parameter dependence}

Surprisingly, obtaining the correct eigenspectrum with this method has been challenging. From a physical point of view this might be explained by the strong stability of this physical problem at this low Reynolds numbers. At present it appears that the large damping rates of the two-dimensional perturbations pose a challenge to the time-stepping approach: if the time integrations are too large, the perturbation vanishes quickly due to its (physical) strong damping, while if the time integration is kept too small, large numerical errors make extraction of the linear perturbation challenging. This effect was found to be substantially more important than the influence on the results of the size of the Krylov subspace dimension or the mesh resolution.

Table (1) shows the influence of the integration time \( \tau \) on the FVM results. It can be seen that, according to the theory, when the integration time is long enough there is no significant improvement in the results that can be obtained, however for short integration times the spectrum cannot be recovered. In addition, if the integration time is too long, most of the transformed spectrum will vanish into the unit circle origin, so very few eigenvalues could be recovered. This maximum integration time can be estimated by \( |\lambda_1\tau| \sim Ln(\epsilon_n) \), where \( \epsilon_n \) is the numerical error corresponding to the spatial discretization. The relative error is defined by

\[ \text{error} = \left| \frac{\lambda_{BG} - \lambda_{TS}}{\lambda_{BG}} \right| \]

(42)

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\( \tau \) & \( \lambda_T \) & \( \lambda_i \) & error \\
\hline
0.05 & 0.37727 & 5.30130 & 13.547\% \\
0.25 & 0.34300 & 0.00000 & 3.233\% \\
0.625 & 0.33975 & 0.00000 & 2.253\% \\
1.125 & 0.33997 & 0.00000 & 2.261\% \\
2.5 & 0.33984 & 0.00000 & 2.282\% \\
5 & 0.33989 & 0.00000 & 2.295\% \\
7.5 & 0.33981 & 0.00000 & 2.271\% \\
10 & 0.33983 & 0.00000 & 2.277\% \\
\hline
\end{tabular}
\caption{Influence of \( \tau \) on FVM results. First Eigenvalue with \( \Delta x = 10^{-3}, n_{\text{nodes}} = 900, m = 60, \epsilon = 10^{-2} \). Relative error obtained comparing first eigenvalue obtained with BG}
\end{table}

Regarding the accuracy of the numerical differentiation, in table (2) it can be seen the effect of the parameter \( \epsilon \) using second order numerical differentiation with the Eriksson’s control equation. As expected, the recommended values available in the literature leads to correct results.

The effect of using a constant \( \epsilon \) can be seen in table (3). If the initial value of the perturbation is properly chosen, this parameter do not have a significant importance. It must be paid attention to the relation between the integration time and the perturbation magnitude. As a first estimation, it can be written that \( |\lambda\tau| \gg Ln(\epsilon_n/|\epsilon u|) \), which means that small initial perturbations can vanish in a shorter integration time \( \tau \) if \( \epsilon \) is too small, so the spectrum will not be recovered.

As expected, higher mesh resolution provides more accurate results. This can be appreciated in table (4). However this does not significantly affect the shape of the spectrum, therefore it does not change the number of correctly recovered eigenvalues, although it significantly increases the computational time needed.
Table 2. Influence of $\epsilon_0$ using Eriksson equation on SCM results. Relative error obtained comparing first eigenvalue obtained with BG

<table>
<thead>
<tr>
<th>$n$</th>
<th>$m$</th>
<th>$\tau$</th>
<th>$\epsilon_0$</th>
<th>$\lambda_r$</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>20</td>
<td>1</td>
<td>$10^{-5}$</td>
<td>0.33918</td>
<td>2.10%</td>
</tr>
<tr>
<td>48</td>
<td>20</td>
<td>1</td>
<td>$10^{-6}$</td>
<td>0.33654</td>
<td>1.30%</td>
</tr>
<tr>
<td>48</td>
<td>20</td>
<td>1</td>
<td>$10^{-7}$</td>
<td>0.33654</td>
<td>1.30%</td>
</tr>
<tr>
<td>48</td>
<td>20</td>
<td>1</td>
<td>$10^{-8}$</td>
<td>0.33654</td>
<td>1.30%</td>
</tr>
</tbody>
</table>

Table 3. Influence of $\epsilon$ on FVM results. First Eigenvalue with $\Delta x = 10^{-3}$, $n_{nodes} = 900$, $m = 60$, $\tau = 10$. Relative error obtained comparing first eigenvalue obtained with BG

<table>
<thead>
<tr>
<th>$\epsilon$</th>
<th>$\lambda_r$</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^0$</td>
<td>0.33987</td>
<td>2.289%</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>0.33983</td>
<td>2.277%</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>0.33954</td>
<td>2.189%</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>0.33948</td>
<td>2.171%</td>
</tr>
</tbody>
</table>

Regarding the FVM mesh influence, the maximum achievable accuracy is given by the mesh characteristics. The modes can be recovered with great accuracy if a good mesh is used along proper time-stepping parameters as it can be seen in table (5).

IV.A.4. On memory requirements and CPU times

The codes have been run on a standard desktop computer having 4GB of RAM memory and 3MB of L2 Cache. CPU times in serial machines are estimated by $ET \sim m \cdot \tau \cdot N_{nodes}/\Delta x$. Table (5) shows that high quality meshes leads to significantly better results that the provided by low quality meshes. However, the increment in CPU times is two order of magnitude larger than the improvement in accuracy. In addition, the required memory to perform the analysis is insignificant compared to the required in the BiGlobal analysis, even with high quality meshes.

V. Summary

A three-dimensional compressible unsteady second-order code for global linear stability has been successfully developed based on finite-volume spatial discretization and JFNK time-stepping. Its strongest advantage over more accurate spectral codes is its flexibility and its ability to study complex geometries by means of unstructured and hybrid meshes. It has been shown that using this approach stability analysis can be carried out in flows over complex geometries. Some guidance for the required parameters of the time-stepping method have been provided. These parameters have a crucial importance in reliably performing the analysis. Identification of proper parameters is actually the weakest point of the time stepping method. By contrast to methods working with the linearized NS equations, such as forming the matrix or time-stepping with LNSE, the advantage of using directly a DNS code is that small errors in the base flow do not affect significantly the results. Work is underway to quantify the behavior of the solver in open flows and results will be presented elsewhere.

Table 4. Influence of mesh resolution $n$ on SCM results. Relative error obtained comparing first eigenvalue obtained with BG

<table>
<thead>
<tr>
<th>$n$</th>
<th>$m$</th>
<th>$\tau$</th>
<th>$\lambda_r$</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>20</td>
<td>1</td>
<td>0.33918</td>
<td>2.10%</td>
</tr>
<tr>
<td>64</td>
<td>20</td>
<td>1</td>
<td>0.33914</td>
<td>2.09%</td>
</tr>
<tr>
<td>80</td>
<td>20</td>
<td>1</td>
<td>0.33907</td>
<td>2.06%</td>
</tr>
</tbody>
</table>
Table 5. Influence of mesh resolution on FVM results. Relative error obtained comparing first eigenvalue obtained with BG. Serial CPU times

<table>
<thead>
<tr>
<th>Mesh</th>
<th>Method</th>
<th>$\Delta x$</th>
<th>$\lambda_r$</th>
<th>error</th>
<th>CPU time</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>64x64 SC</td>
<td>BG</td>
<td>-</td>
<td>0.3322</td>
<td>-</td>
<td>2 min</td>
<td>1.6GB</td>
</tr>
<tr>
<td>Mesh A</td>
<td>FVM</td>
<td>$10^{-3}$</td>
<td>0.3398</td>
<td>2.28%</td>
<td>0.5h</td>
<td>500KB</td>
</tr>
<tr>
<td>Mesh B</td>
<td>FVM</td>
<td>$10^{-4}$</td>
<td>0.3313</td>
<td>0.27%</td>
<td>100h</td>
<td>20MB</td>
</tr>
</tbody>
</table>

References


Figure 1. Graphical representation of the temporal spectrum transformation

Figure 2. Spectrum of the regularized LDC at $Re = 200$ obtained with three different methods. Black squares for BG, upside-down green triangles for FVM and magenta triangles for SCM.
Figure 3. First Eigenvector of the regularized LDC at $Re = 200$ obtained with three different methods. Eigenvectors are normalized with $\hat{u}_{\text{max}}$ and $\hat{v}_{\text{max}}$. Dashed lines mean negatives values. 21 equidistant isolines from $\hat{u} = -1$ to $\hat{u} = 1$ and from $\hat{v} = -1$ to $\hat{v} = 1$. 

American Institute of Aeronautics and Astronautics
Figure 4. Second Eigenvector of the regularized LDC at $Re = 200$ obtained with three different methods. Eigenvectors are normalized with $\hat{u}_{\text{max}}$ and $\hat{v}_{\text{max}}$. Dashed lines mean negatives values. 21 equidistant isolines from $\hat{u} = -1$ to $\hat{u} = 1$ and from $\hat{v} = -1$ to $\hat{v} = 1$. 

15 of 17

American Institute of Aeronautics and Astronautics
Figure 5. Three-dimensional modes for LDC at $Re = 200$ with $\beta \approx 0$ obtained with BG and FVM. Eigenvectors are normalized with $\hat{w}_{\text{max}}$. Dashed lines mean negatives values. 21 equidistant isolines from $\hat{w} = -1$ to $\hat{w} = 1$. Upper row $\lambda_3 = 0.4627 \pm 0.1099i$, Lower row $\lambda_4 = 0.9144 \pm 0.5503i$. 
Figure 6. Density mode for LDC at $Re = 200$ obtained with FVM. Eigenvectors are normalized with $\hat{\rho}_{\text{max}}$. Dashed lines mean negatives values. 21 equidistant isolines from $\hat{\rho} = -1$ to $\hat{\rho} = 1$

(a) Mesh A

(b) Mesh B

Figure 7. Different meshes for LDC analysis. Mesh A (left): $n_{\text{nodos}} = 900$, $\Delta x = 10^{-3}$. Mesh B (left): $n_{\text{nodos}} = 4500$, $\Delta x = 10^{-4}$