

A Boundary Condition Stability Analysis of Finite-Volume Methods on Unstructured Meshes

Reza Zangeneh¹, Dr. Carl Ollivier-Gooch²

^{1,2} Department of Mechanical Engineering, University of British Columbia, Vancouver, Canada

Corresponding author: r.zangeneh@alumni.ubc.ca

Abstract: The purpose of this paper is two-fold; first, a systematic approach is developed to improve the stability of cell-centered finite volume methods on unstructured meshes by optimizing boundary conditions. This approach uses the rightmost eigenpairs of the spatially discretized system of equations to determine the existence or the path to stability. This will ensure the energy stability of the system; consequently resulting in convergence. To this end, it exploits first order gradients of eigenvalues with respect to the types of boundary conditions. Secondly, the sensitivity of the rightmost eigenvalues to the solution is measured to investigate the effects of using surrogate or half-converged solutions for the purpose of linearizing the semi-discretized Jacobian.

Keywords: Spectral stability analysis, Boundary condition, Energy stability, Eigenvalue analysis.

1 Introduction

In computational aerodynamics problems, the boundary conditions specified to approximate the physical solution at the boundaries of the computational domain are of paramount importance. Thomas and Salas [1] showed that the numerical accuracy of the steady state solution can be greatly affected by careless implementation of boundary conditions at the truncated farfield boundaries. Spurious non-physical waves originating or reflected from the farfield boundaries can contaminate the interior solution, including in particular the aerodynamic forces. Hence, there have been many different efforts to derive boundary conditions at farfield which asymptotically approximate the farfield free stream conditions (e.g., see a review on these boundary conditions by Tsynkov [2] and the reference therein). One of the first methods developed to approximate the analytical solution at the farfield corrects the constant free stream conditions with a vortex flow which is dependent on the airfoil lift (see [1]). This method is successful in reducing the sensitivity of the forces on the airfoil with respect to the location of the farfield boundary, albeit with the added cost of introducing non-local features, i.e. lift coefficient, to the boundary conditions. In other method for reducing non-physical oscillations, non-reflecting boundary conditions at the farfield is imposed. These boundary conditions, which were first introduced by Engquist and Majda [3] and further improved by [4, 5], direct the oscillations out of the physical domain. Recently, there have been more work on these types of boundary conditions. In another attempt, Allmaras et al. [6] derived a boundary condition for the farfield which is both local and independent of the forces on the airfoil, relying on their modified variables and radial velocity. They showed that by applying this boundary condition, the farfield boundary can greatly be brought closer to the airfoil.

One of the shortcomings in all of the research papers in the use of boundary conditions is the lack of an applied approach to systematically choose the most stable boundary conditions amongst the set of all available boundary conditions. In this paper, instead of introducing a specific type of boundary condition which may not be always stable in practice, we devise a framework which can find the most stable boundary condition for a given problem. This work stems from our previous works [7, 8] which showed how the location of mesh vertices can be modified to improve convergence and stability. We calculate the gradients of the

rightmost eigenvalues of the semi-discretized Jacobian with respect to the types of the boundary conditions to optimize boundary conditions through gradient descent. The eigenvalues and eigenvectors of the linearized system is calculated by SLEPC [9] which is a large scale sparse eigenvalue solver. It is also fully adaptable with the data structure and linear algebra package, PETSC [10] which we use in our in-house libraries.

The paper is organized as follows. In Section 2 we describe the spatial and time discretization schemes we use to solve Euler or Navier-Stokes problems to steady state solution. We then describe different types of commonly-used farfield boundary conditions for 2D external aerodynamics. In Section 3, we first introduce a systematic approach to optimize different types of boundary conditions to improve the numerical stability by calculating the derivatives of the rightmost eigenvalues of the semi-discretized Jacobian. Secondly, the sensitivity of these eigenvalues to the solutions at which the global Jacobian of the flux integral was linearized is presented. Section 4 showcases the numerical stability results for the aforementioned optimization approach as well as the sensitivity of the eigenvalues with respect to the solution. Finally, Section 5 gives conclusion.

2 Background

2.1 Discretization Scheme

The partial differential equations solved in CFD can be written as:

$$\frac{\partial u(x, t)}{\partial t} + \nabla \cdot F(x, t) = S(x, t) \quad (1)$$

where Eq. 1 is recast to the following formulation by finite volume discretization schemes:

$$\frac{dU}{dt} = \frac{1}{V} \int_V \frac{du(x, t)}{dt} dV = -\frac{1}{V} \oint_A F dA + \frac{1}{V} \int_v S dV = R(U) \quad (2)$$

In Eq. 2, U is the global vector of unknowns which are the control volume averaged solutions. F and S are the flux and source functions respectively. The residual is denoted as R in Eq. 2. To calculate the flux integrals, suitable flux functions (in this paper, we use Roe's flux scheme [11]) are computed and accumulated at each Gauss integration point along control volume boundaries. These flux functions calculated at Gauss points are evaluated by reconstructed solutions at those points from control volume solution averages (see [12, 13] for more details). This solution reconstruction from piecewise constant control volume solution averages will result for a more accurate solution approximation. Finally, implicit time advance methods are applied to Eq. 2 to discretize the equation in time and evolve the solution to steady state. To evolve in time using implicit Euler time advance, Eq. 2 is recast as:

$$\frac{U^{n+1} - U^n}{\Delta t} = R(U^{n+1}) = R(U^n) + J\delta U + \mathcal{O}(\delta U^2) \quad (3)$$

where the residual function is linearized about the solution state U^n . The system of semi-discretized equations (Eq. 2) can also be linearized to form a matrix vector product (Eq. 4) and a global Jacobian matrix, J :

$$\frac{dU}{dt} = \frac{\partial R}{\partial u} U = JU \quad (4)$$

From energy stability analysis, Eq. 4 and hence the corresponding finite volume discretization is stable if and only if the Jacobian matrix, J , is negative semi-definite. This means that in the eigen-decomposition (Eq. 5 and 6) of Jacobian matrix, the rightmost eigenvalues, which have the largest positive real parts, determine the stability of the problem.

$$Jx_i = \lambda_i x_i \quad (5)$$

$$y_i J = y_i \lambda_i \quad (6)$$

In Eq. 5 and 6, λ_i , x_i , and y_i are respectively the i^{th} rightmost eigenvalue, its right eigenvector and its left eigenvector.

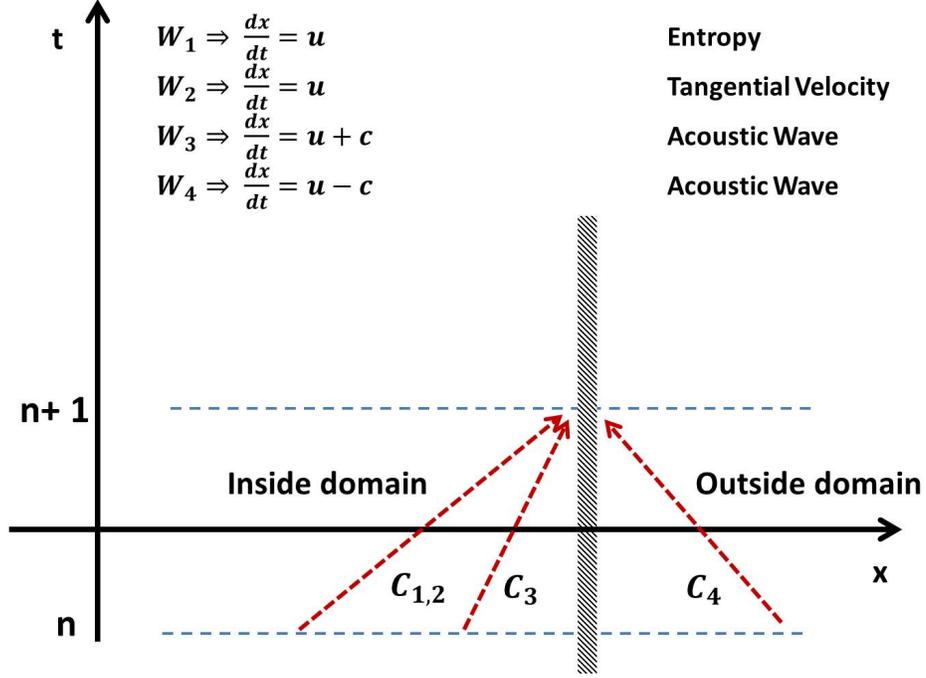


Figure 1: Characteristic lines for a subsonic two-dimensional flow at outlet. The y direction is normal to the surface.

2.2 Farfield Boundary Conditions

In this sub-section we review some of the common boundary conditions used for external compressible subsonic flows. If all variables were known at the boundary, evaluation of Eq. 2 would be seamless, however this is not usually the case. To determine all the states at boundaries for the discretized equations, numerical and physical conditions need to be set. As each characteristic direction carries a piece of information into or out of the discretized domain, only these transported variables can be set freely at the boundary; these are the physical boundary conditions. The rest of the variables needed by the set of discretized equations can be determined by the discretization scheme itself; these are the numerical boundary conditions. For example, Figure 1 shows the characteristic lines for a subsonic outflow in a two-dimensional flow.

All the numerical and physical boundary conditions in this paper are enforced weakly through proper flux functions. The following shows different types of boundary conditions used for a two-dimensional subsonic flow in this paper.

2.2.1 Characteristics with vortex correction BC

The boundary conditions can be fixed in a straightforward way by specifying the propagated variables directly from the characteristic information or similarly by the Riemann invariants. In Eq. 7 and 8 the primitive variables that are directly transferred with the characteristic variables are determined both at in-flow and out-flow. The rest of the required variables are determined from the interior discretization.

- In-flow:

$$\begin{aligned}
 \text{Negative acoustic wave:} \quad & (p - p_\infty) - \rho c (V_n - V_{\infty n}) = 0 \\
 \text{Entropy:} \quad & (p - p_\infty) - c^2 (\rho - \rho_\infty) = 0 \\
 \text{Tangential velocity:} \quad & (V_t - V_{\infty t}) = 0
 \end{aligned} \tag{7}$$

- Out-flow:

$$\text{Positive acoustic wave: } (p - p_\infty) + \rho c (V_n - V_{\infty n}) = 0 \quad (8)$$

where variables subscripted with ∞ represent farfield variables and c is the sound velocity. In external flows (and similarly internal flows) the disturbances at the farfield (or inlet and outlet) can be transferred to the interior of the flow. To minimize these effects, normally the physical boundaries are placed far from the domain of interest.

However by better approximating the solution at the farfield, the farfield boundary can be taken closer to the flow region of interest (e.g. airfoil) without affecting the flow solution. For the inviscid, compressible flow condition, the farfield flow perturbations can be modeled by the linear potential equation. The farfield uniform free stream variables in Eq. 7 and 8 can be corrected by the asymptotic perturbations to the lowest order from the thin airfoil theory and the linear potential equation (see [1] for more details):

$$\begin{aligned} V_{\infty x} &= M_\infty c_\infty \cos(\alpha) + F \sin(\theta) \\ V_{\infty y} &= M_\infty c_\infty \sin(\alpha) - F \cos(\theta) \end{aligned} \quad (9)$$

where M_∞ is the free stream Mach number, α is the angle of attack, and c_∞ is the free stream sound velocity. The parameter F is:

$$F = \frac{C_l L M_\infty c_\infty}{4\pi} \frac{\sqrt{1 - M_\infty^2}}{r} \frac{1}{1 - M_\infty^2 \sin^2(\theta - \alpha)} \quad (10)$$

The perturbations in Eq. 10 are governed by the circulation of a vortex centered at the airfoil quarter-chord. The circulation itself is dependent on the lift coefficient, C_l . The lift coefficient is measured around any arbitrary surface enclosing the airfoil. In Eq. 10, parameters L , r , θ are respectively the chord length, the radius measured from the center of the vortex to the boundary location, and the angle between the chord and the line connecting the center of the vortex to the boundary location.

2.2.2 Pressure BC

Another commonly used boundary condition at the farfield is specifying the pressure. As pressure is constant across a viscous wake, this does not disturb the transport of viscous perturbations. Thus, imposing a constant pressure downstream of the flow is a common practice. In this boundary condition, we specify the free stream back pressure at outflow and we use total pressure, total temperature, and angle of attack at the free stream to determine the three physical boundary condition at the inflow using isentropic relations.

- In-flow:

$$P_{tot}, T_{tot}, \alpha$$

- Out-flow

$$p = p_\infty$$

2.2.3 Dirichlet BC

In this type of boundary condition, all the primitive (or conservative) variables are directly determined from free stream variables at farfield. The boundary fluxes are then calculated using these fixed values. This problem is over-specified at both inflow and outflow. However, it has been shown that the numerical errors tend to be localized.

- In-flow:

$$\rho, u, v, p$$

- Out-flow

$$\rho, u, v, p$$

2.2.4 Radial Velocity BC

To bring the far field boundary closer to the airfoil without affecting the lift and drag coefficients, Allmaras et al. [6] suggested setting properties at the far field that are to first order independent of the airfoil lift:

- In-flow:

$$\begin{aligned} \text{Total Enthalpy: } \left[\frac{\gamma}{\gamma-1} \frac{p}{\rho} + \frac{V^2}{2} \right] &= \left[\frac{\gamma}{\gamma-1} \frac{p}{\rho} + \frac{V^2}{2} \right]_{\infty} \\ \text{Entropy: } \frac{p}{\rho^{\gamma}} &= \frac{p_{\infty}}{\rho_{\infty}^{\gamma}} \\ \text{Radial Velocity: } V_r &= \vec{V}_{\infty} \cdot \hat{e}_r \end{aligned} \quad (11)$$

- Out-flow

$$V_r = \vec{V}_{\infty} \cdot \hat{e}_r \quad (12)$$

In Eq. 11 and 12, \hat{e}_r is the radial direction calculated from the vortex center, situated at a quarter of the chord, to the boundary location.

2.2.5 Non-reflecting outflow BC

As explained earlier, the physical information crosses the boundaries with characteristic variables. This in turn causes reflection of the perturbation waves at the boundaries. For example, specifying only the back pressure at outflow does not satisfy the perturbations of the characteristic waves $\Delta W = (p - p_{\infty}) - \rho c (V_n - V_{\infty n})$; hence, pressure waves with magnitudes of $\rho c (V_n - V_{\infty n})$ will be reflected back into the domain. For steady-state computations with constant pressure boundary conditions, Rudy et al. [4] proposed a method which damps out the pressure waves leaving the computational domain in the transient and satisfies the constant pressure at steady-state:

- In-flow:

$$P_{tot}, T_{tot}, \alpha$$

- Out-flow

$$\frac{\partial p}{\partial t} - \rho c \frac{\partial V_n}{\partial t} + \omega (p - p_{\infty}) = 0$$

where ω decreases from roughly 0.7 – 0.8 in subsonic to 0.1 – 0.3 in transonic flows.

3 Methodology

3.1 Boundary Condition Optimization

In this work, we optimize boundary conditions to improve the numerical stability of the semi-discrete system of equations arising from finite volume discretization on unstructured meshes. In previous work [8] we showed that mesh related instabilities can be eliminated by modifying the mesh vertices locally based on the gradients of the right-most eigenvalues with respect to the mesh vertex displacement vectors. Herein, we are looking into the stability effects of different boundary condition implementations. Throughout this work, we apply boundary conditions weakly using flux calculations. The premise of this work is to use the semi-discretized Jacobian of the flux integral, and its right most eigenvalues and eigenvectors to calculate the gradients of eigenvalues of the Jacobian matrix with respect to the scheme design parameters b , which are boundary flux weights in this work.

$$\frac{\partial}{\partial b}(Jx_i = \lambda_i x_i) \quad (13)$$

$$y_i \left(\frac{\partial J}{\partial b} x_i + J \frac{\partial x_i}{\partial b} \right) = \frac{\partial \lambda_i}{\partial b} x_i + \lambda_i \frac{\partial x_i}{\partial b} \quad (14)$$

$$(y_i J - y_i \lambda_i) \frac{\partial x_i}{\partial b} = y_i \frac{\partial \lambda_i}{\partial b} x_i - y_i \frac{\partial J}{\partial b} x_i \quad (15)$$

We take the derivative of the right eigen-system with respect to the variable of interest in Eq. 13. While keeping the solution U fixed, we multiply from left by the left eigenvector in Eq. 14. The left hand side of Eq. 15 contains the left eigen-problem, leaving us with Eq. 16. Since the eigen-vectors are normalized so that $x_i \cdot y_i = 1$, we take the gradients of the eigenvalues:

$$\frac{\partial \lambda_i}{\partial b} = y_i \frac{\partial J}{\partial b} x_i \quad (16)$$

where parameter b is the scheme design variable which we are taking the derivative with respect to. One key step in evaluating Eq. 16 is approximating the gradient of the Jacobian matrix with respect to the variables of interest, $\frac{\partial J}{\partial b}$. We choose to use finite differencing on the Jacobian matrix, however this gradient can also be calculated using reverse automatic differentiation.

Moreover, we are aiming to systematically select and optimize the boundary conditions to stabilize the finite volume methods. To find the most stable boundary condition configuration, we first linearly superimpose the different boundary fluxes for each boundary control volume. Second, we optimize these superposition parameters based on the gradient of the right-most eigenvalues of the Jacobian matrix based on these parameters. For a simple case in which optimization is between only two types of boundary conditions, we combine the two different boundary fluxes with weights b_{cv_i} and $1 - b_{cv_i}$ for every boundary control volume:

$$Flux = b_{cv_i} F_1 + (1 - b_{cv_i}) F_2 \quad (17)$$

$$\frac{\partial J}{\partial b_{cv_i}} = J_1 - J_2 \quad (18)$$

Now by using Equations 16 and 18, where y and x are respectively the left and right eigenvectors of the right most eigenvalue, we can measure the gradient of the eigenvalue with respect to the boundary flux weights b . Hence, by expanding the Taylor series of the new eigenvalue of the Jacobian matrix J , we can find the best boundary flux configuration for each boundary control volume so that the new projected eigenvalue λ^{n+1} is on the left hand side of the complex plane of eigenvalues:

$$\lambda^{n+1} = \lambda^n + \overrightarrow{\Delta b} \frac{\partial \lambda}{\partial b} \leq 0 \quad (19)$$

$$\overrightarrow{\Delta b} = -|K| \frac{\partial \lambda}{\partial b} \quad (20)$$

$$K = \frac{\lambda^n}{\left(\frac{\partial \lambda}{\partial b}\right)^2} \quad (21)$$

$$b_{opt} = b + \overrightarrow{\Delta b} \quad (22)$$

In this way the best boundary condition configuration at each boundary control volume is attained by optimizing the boundary flux weights b .

3.2 Sensitivity of Eigenvalues to Solution

As we have typically used surrogate solutions or lower order solutions to approximate the semi-discrete Jacobian for eigenvalue analysis, it is of paramount importance to understand the effects of these approximate

solutions on the rightmost eigenvalues of the Jacobian matrix. One common way to compute the sensitivity is to use the adjoint methods (e.g., see the work done by Giles and Pierce [14]). In an adjoint problem setting, we specify the output as the rightmost eigenvalues, λ , and the sensitivity parameter in theory can be any measurable variable. However since we are calculating the sensitivity with respect to type of boundary conditions in the previous section, we use the same variables, b , for the sensitivity parameter:

$$\frac{d\lambda}{db} = -\Psi^T \frac{\partial R}{\partial b} + \frac{\partial \lambda}{\partial b} \quad (23)$$

$$\left[\frac{\partial R}{\partial U} \right]^T \Psi = J^T \Psi = \left[\frac{\partial \lambda}{\partial U} \right]^T \quad (24)$$

where in the functional output sensitivity equation, Eq. 23, and the adjoint linear problem, Eq. 24, Ψ is the adjoint variable and R is the residual of the primal PDE. Now the sensitivity of eigenvalue with respect to the solution can be calculated by:

$$\frac{\partial \lambda}{\partial U} = \left(\frac{\partial R^T}{\partial b} \right)^{-1} \left(\frac{d\lambda}{db} - \frac{\partial \lambda}{\partial b} \right)^T \quad (25)$$

where the derivatives can be calculated by doing finite differences on the mesh and solution:

$$\begin{aligned} \frac{d\lambda}{db} &= y \left(\frac{J(U(b+db), b+db) - J(U(b), b)}{\|db\|} \right) x \\ \frac{\partial \lambda}{\partial b} &= y \left(\frac{J(b+db) - J(b)}{\|db\|} \right) x \\ \frac{\partial R}{\partial b} &= \frac{R(b+db) - R(b)}{\|db\|} \end{aligned}$$

However, calculating the sensitivity of the eigenvalues with respect to the solution using Eq. 25 as it requires calculating multiple derivatives for each control volume in the problem. Instead, we use Eq. 16 and the Frechet derivatives to calculate these sensitivities. In this way, the sensitivity of eigenvalues to the solution can be calculated by substituting parameters b in Eq. 16 with control volume solution averages:

$$\frac{\partial \lambda}{\partial U} = y \frac{\partial J}{\partial U} x \quad (26)$$

where the right hand side of Eq. 26 is calculated by multiplying the left eigenvector by the following Frechet derivative:

$$\frac{\partial J}{\partial U} \hat{e}_k x = \frac{J(U + \varepsilon \hat{e}_k x) - J(U)}{\varepsilon} \quad (27)$$

In Eq. 26 and 27, U is the vector of control volume solution averages and ε is an infinitesimal number for differencing. Eq 27 is calculated row by row for each control volume to break up an otherwise third rank tensor. However, in evaluation of Eq. 27, there is no need to recalculate the whole Jacobian matrix, J , for all control volumes in the mesh everytime as this would result in a $O(N^2)$ computational time where N is the number of control volumes. Instead, when calculating each row of the right hand side of Eq. 26, we determine the specific rows in the Jacobian matrix which need to be updated after solution within a control volume has been perturbed. In other words, once the solution within control volume α changes, the flux integral at all the nearby control volumes which have control volume α in their reconstruction stencil changes. As the discretization is constant and it does not change once the solution changes, the sparsity of the Jacobian matrix is known a prior to its evaluations. This means that for identifying the control volumes for which the flux integral changes after a solution perturbation at control volume α , we look at non-zero entries of the column of the Jacobian matrix associated with control volume α . In this way, the evaluation of Eq. 27 is of order of $O(N)$.

4 Results and Discussion

4.1 Boundary Condition Optimization

In our previous works [7, 8] to investigate and improve the stability of finite volume methods on unstructured meshes by optimizing the location of the mesh vertices, we realized that not all the unstable eigenmodes of all the problems were fixable by merely modifying the mesh vertices. To showcase the difference between different eigenmodes, we show here two eigenvectors for an inviscid Euler problem with Mach 0.8 and an angle of attack of 2° (Figure 2). We measured the gradient of eigenvalues with respect to the mesh vertex locations for different boundary conditions at farfield. Figure 2a clearly shows that modifying the vertex locations for different boundary conditions result in the same values of eigenvalue gradients. In other words, the gradients in this case are independent of the types of boundary conditions but the eigenvalue is strongly influenced by the changes in the mesh (notice the high values of gradients in Figure 2a). On the other hand, changing the boundary conditions at the farfield has noticeably changed the gradients of another eigenvalue which is more dependent on the physics of the problem (Figure 2b). Additionally, in Figure 2b, all the gradients are orders of magnitude smaller than the ones in Figure 2a. This in turn signals that this specific eigenmode is not sensitive to the changes in the mesh vertices.

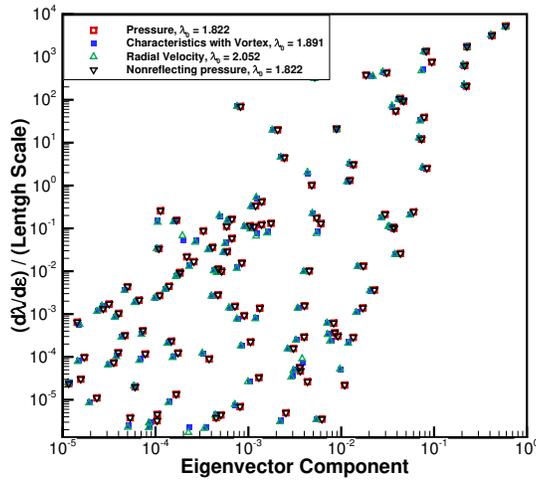
To investigate the sensitivity of the eigenvalues with respect to the boundary conditions and to validate the boundary condition stabilization in sub-section 3.1, we stabilize or improve the convergence rates for an inviscid Euler problem. Figure 3 shows the boundary condition types at the far field and their best configuration for an Euler problem on NACA 0012 airfoil with Mach 0.55, and angle of attack of 2° . In this problem, four different far field boundary conditions of vortex characteristics, radial [6], pressure, and Dirichlet were used. It can be observed from Figure 4 that these optimized boundary condition configurations have been able to stabilize an originally unstable problem or in some cases improve the convergence rate. In all the cases, the bundled boundary conditions are showing better convergence characteristics. After the optimization process, the boundary conditions are linear averages of each individual boundary conditions where the weights of these linear combinations are the results of the optimization.

4.2 Sensitivity of Eigenvalues to Solution

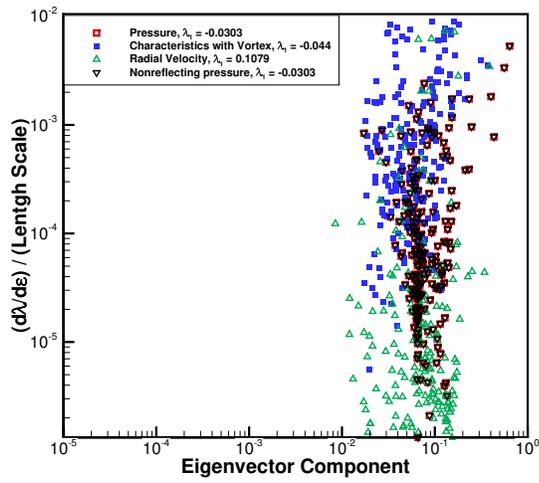
To present the effect of the solution on the eigenvalues, we developed a methodology to calculate the sensitivity of each of the eigenvalues to the solution in Section 3.2. Here, we showcase this sensitivity for a second order inviscid transonic Euler problem with Mach number 0.8 and an angle of attack of $\alpha = 1.25^\circ$. This problem with 4200 control volumes is initialized with farfield free stream flow conditions everywhere. It is evolved to steady-state using pseudo-time stepping with an implicit time advance scheme. Figure 5 shows the L_2 norm of residual for this problem and the three iterations at which we have measured the sensitivity of the eigenvalues with respect to the solution.

Figures 6, 7, 8, and 9 are respectively showing the gradients of the rightmost eigenvalue with respect to the four primitive variables, ρ , u , v , p at all control volumes. In addition to the three iterations, as shown in Figure 5, at which the sensitivity is measured, we are also showing the sensitivity of the rightmost eigenvalue of the same second-order discretization evaluated using the first-order solution as a linearization. This sheds light on our previous works [7, 8] where we used a lower-order steady-state solution to linearize a higher-order discretization (e.g., using a first-order steady-state solution as a surrogate solution to the second order-solution for linearizing the Jacobian).

Comparing the sensitivity maps between all four different solutions and for each of the primitive variables reveals that both the half converged solution and the lower order solution have similar features to the sensitivity of the fully converged solution at near field. The rightmost eigenvalues of all the four solutions are quite close while the eigenvalues at iteration 35 (with a flux integral residual of order of 10^{-1}) and the steady-state solution are exactly identical. However, sensitivity maps at the first iterations of convergence (Figures 6b, 7b, 8b, and 9b) show high gradients mostly further away from the airfoil and they mostly fall short of capturing the important features adjacent to the airfoil. This is indeed aligned with the intuition



(a) Boundary condition independent eigenmode



(b) Boundary condition dependent eigenmode

Figure 2: Derivative of the eigenvalue with vertex movement for two eigenmodes of Euler problem with Mach 0.8, and angle of attack of 2° on a NACA 0012 airfoil with 2000 control volumes. Vertical axes are the average of the gradients of eigenvalues with respect to the mesh vertices for all the vertices of each control volume. The gradients of eigenvalues with respect to mesh vertices are calculated at each vertex. However, to project them on the eigenvector components, i.e. each unknown at each control volume, the gradients are averaged for each CV. The horizontal axes are the average of the values of all components of eigenvector corresponding to each control volume. Each control volume has the same number of entities in the eigenvector as the number of unknowns.

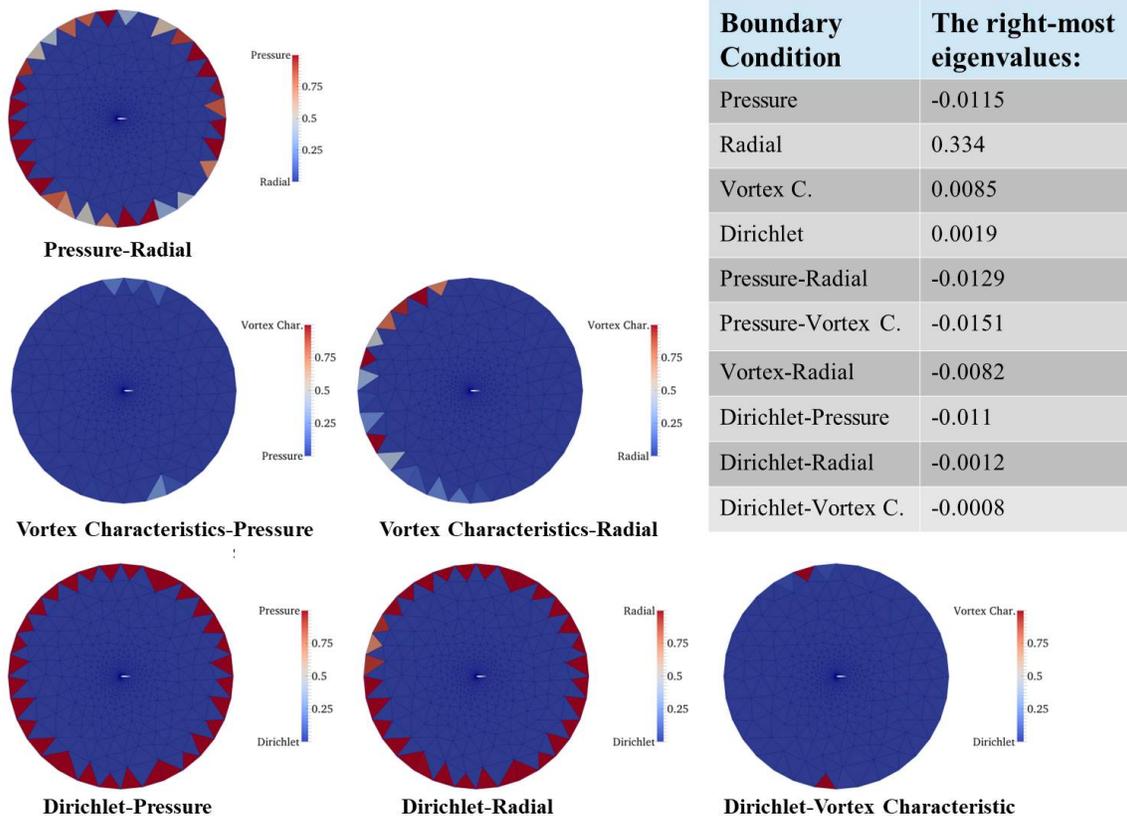


Figure 3: Boundary condition configurations: far-field boundary condition optimization for an Euler problem with Mach 0.55, and angle of attack of 2° on a NACA 0012 airfoil with 2000 control volumes

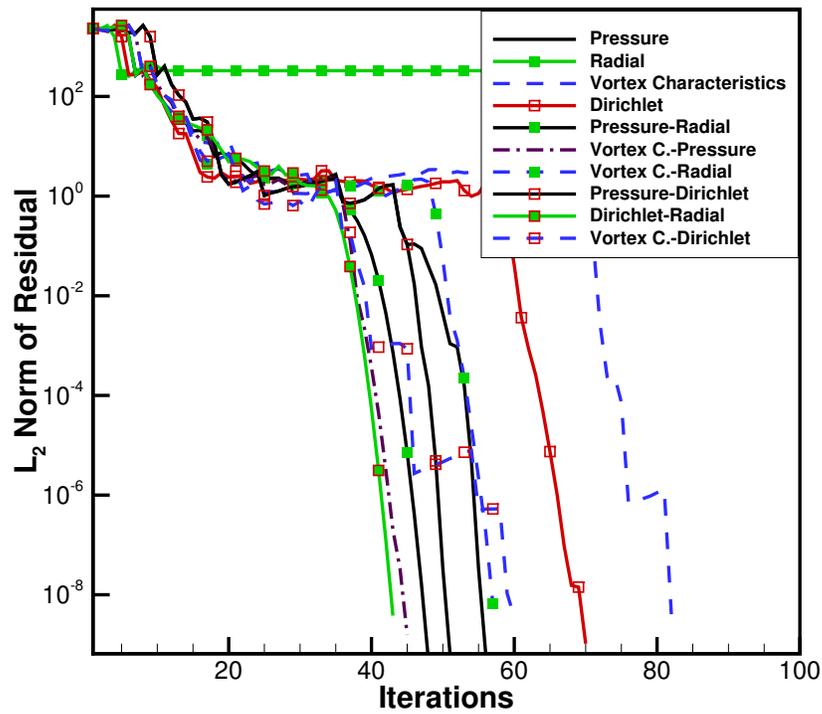


Figure 4: History of convergence: far-field boundary condition optimization for an Euler problem with Mach 0.55, and angle of attack of 2° on a NACA 0012 airfoil with 2000 control volumes

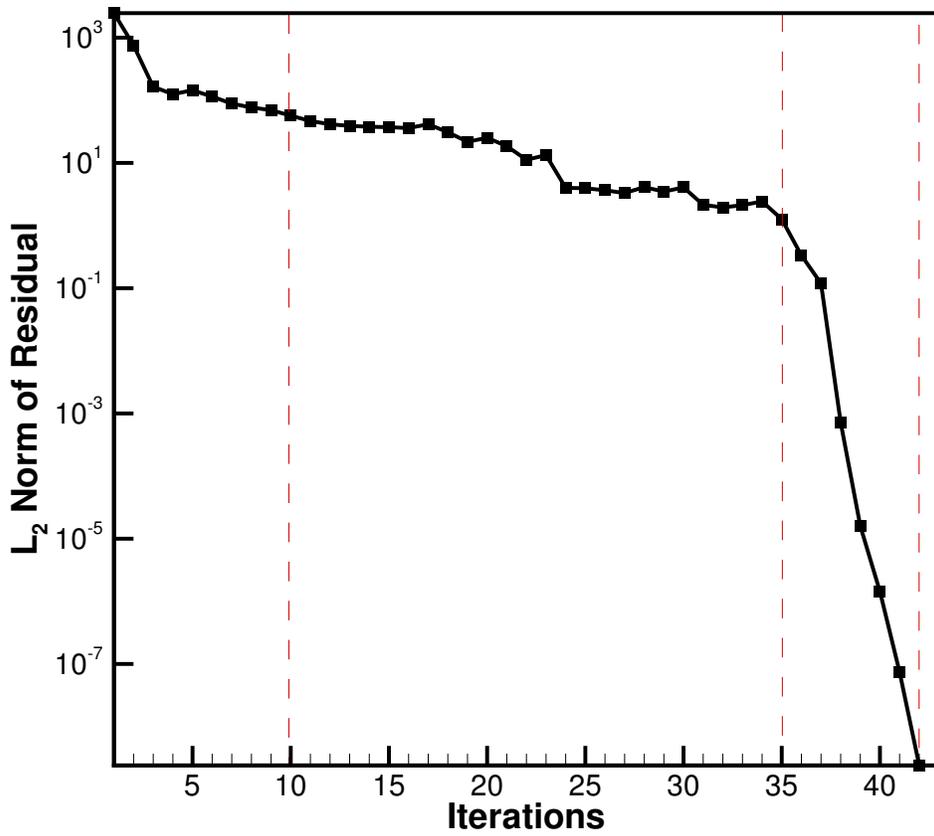
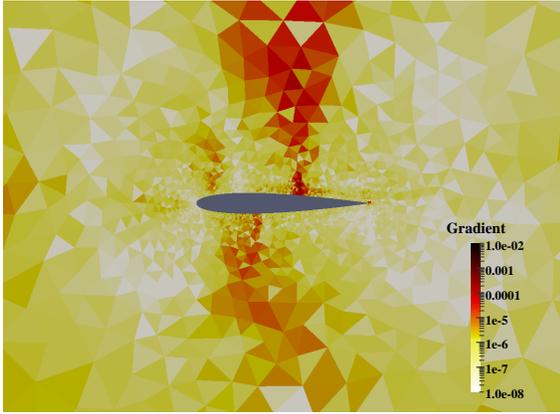
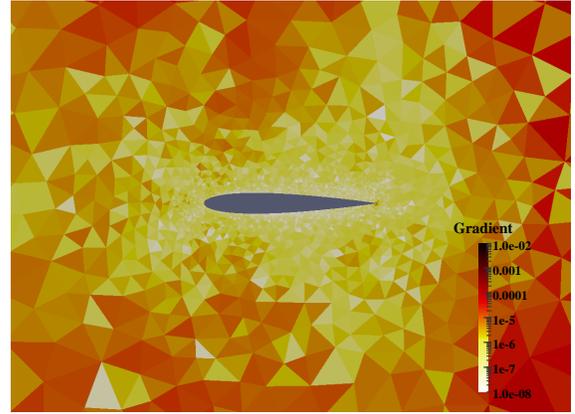


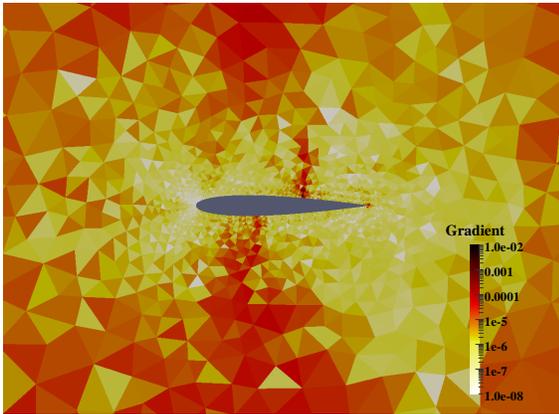
Figure 5: The residual history for an Euler problem with Mach 0.8 and angle of attack of $\alpha = 1.25^\circ$ on NACA 0012 airfoil with 4200 control volumes.



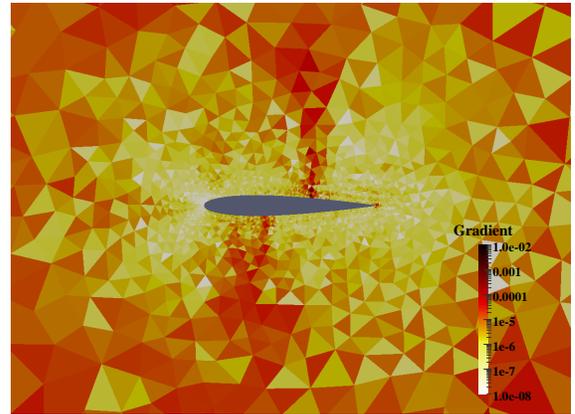
(a) First order solution, $\lambda_0 = -0.0301$



(b) At 10th iteration, $\lambda_0 = -0.0312$

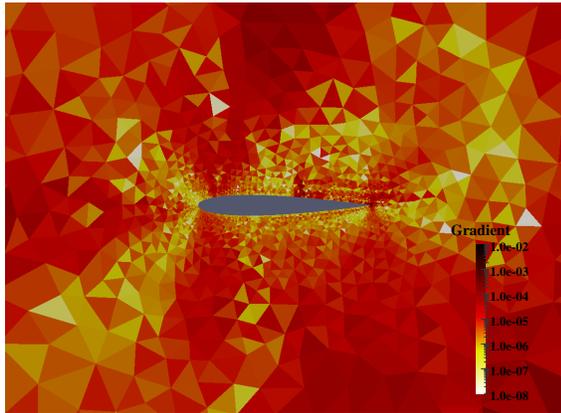


(c) At 35th iteration, $\lambda_0 = -0.0318$

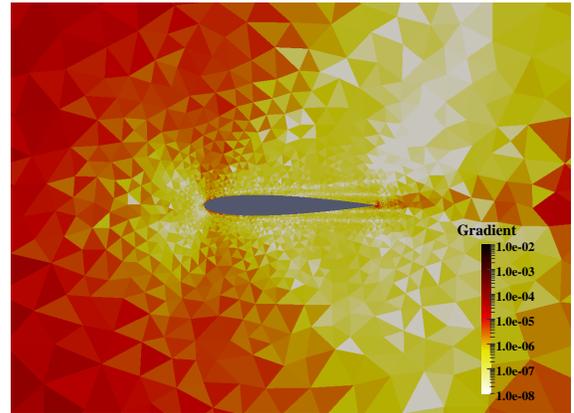


(d) At 42nd iteration, $\lambda_0 = -0.0318$

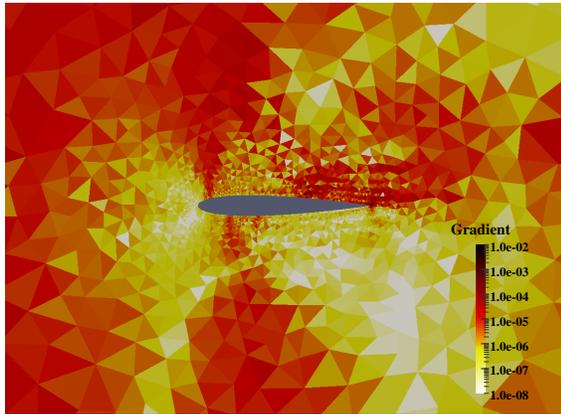
Figure 6: Sensitivity of the rightmost eigenvalue with respect to the density, ρ , at three different stages of convergence (see Figure 5).



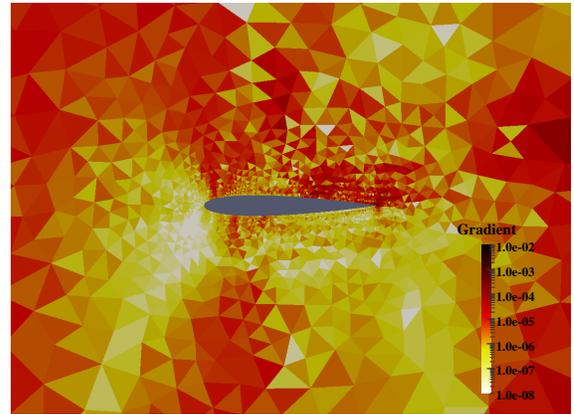
(a) First order solution, $\lambda_0 = -0.0301$



(b) At 10th iteration, $\lambda_0 = -0.0312$

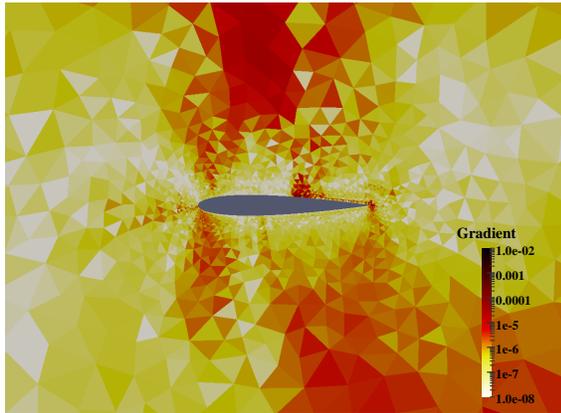


(c) At 35th iteration, $\lambda_0 = -0.0318$



(d) At 42nd iteration, $\lambda_0 = -0.0318$

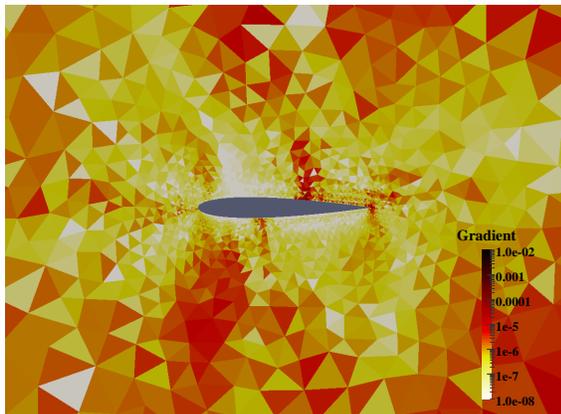
Figure 7: Sensitivity of the rightmost eigenvalue with respect to the axial velocity, u , at three different stages of convergence (see Figure 5).



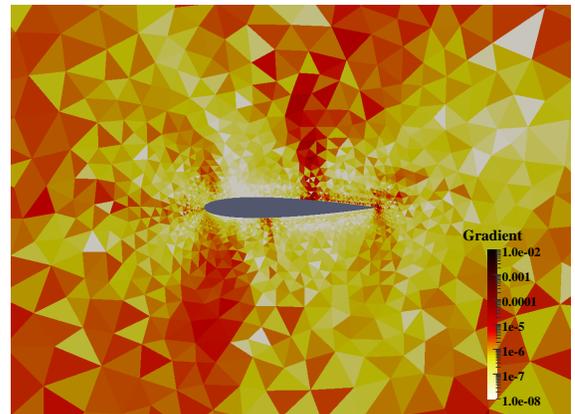
(a) First order solution, $\lambda_0 = -0.0301$



(b) At 10^{th} iteration, $\lambda_0 = -0.0312$

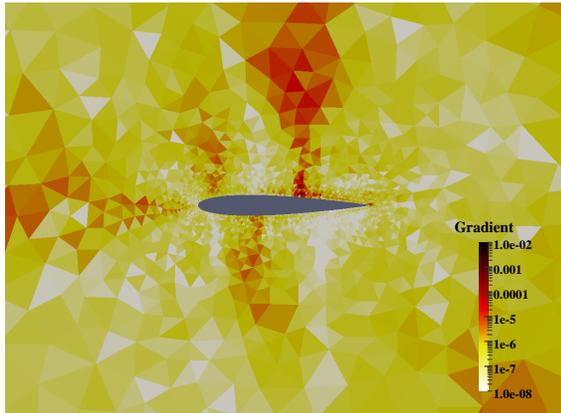


(c) At 35^{th} iteration, $\lambda_0 = -0.0318$

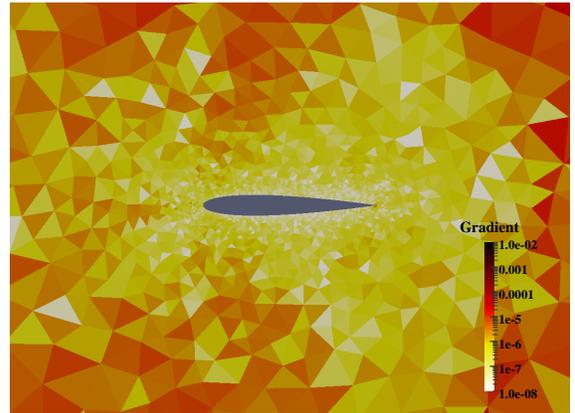


(d) At 42^{nd} iteration, $\lambda_0 = -0.0318$

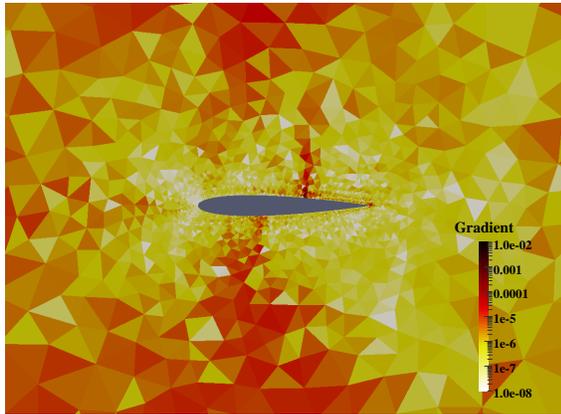
Figure 8: Sensitivity of the rightmost eigenvalue with respect to the vertical velocity, v , at three different stages of convergence (see Figure 5).



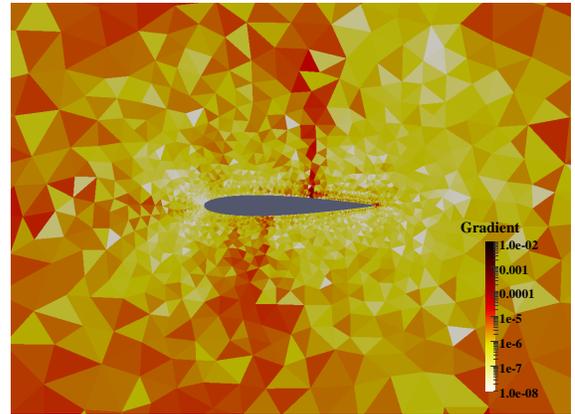
(a) First order solution, $\lambda_0 = -0.0301$



(b) At 10^{th} iteration, $\lambda_0 = -0.0312$



(c) At 35^{th} iteration, $\lambda_0 = -0.0318$



(d) At 42^{nd} iteration, $\lambda_0 = -0.0318$

Figure 9: Sensitivity of the rightmost eigenvalue with respect to the pressure, p , at three different stages of convergence (see Figure 5).

that the solution near the airfoil had not yet had the chance to evolve from the free stream conditions. The study of the sensitivities of the eigenvalues with respect to the solution further confirms the idea of using either a lower-order or half-converged solution as a surrogate solution for linearizing the Jacobian in our stability algorithms.

5 Conclusion and Future Works

In this work, we first proposed and studied a new approach to optimize boundary conditions for stabilization of finite volume discretization on unstructured meshes. This method superimposes different boundary conditions for aerodynamic problems at far-field. It automatically optimizes this mixed boundary condition configuration through gradients of the right most eigenvalues of the semi-discretized Jacobian. In this way, a stable boundary condition configuration is designed that produces a stable finite volume method in terms of steady state convergence. Secondly, we presented a methodology to investigate the sensitivities of the eigenvalues with respect to the solution at any point when converging to the steady state solution. Our results showed that both the lower-order and half-converged solutions capture enough variations in the solution to be able to approximate the semi-discretized Jacobian of a higher-order solution at steady state.

References

- [1] J. L. Thomas and M. D. Salas. Far-field boundary conditions for transonic lifting solutions to the Euler equations. *AIAA Journal*, 24(7):1074–1080, 1986.
- [2] S. V. Tsynkov. Numerical solution of problems on unbounded domains. A review. *Applied Numerical Mathematics*, 27(4):465–532, 1998.
- [3] B. Engquist and A. Majda. Absorbing boundary conditions for numerical simulation of waves. In *Proceedings of the National Academy of Sciences*, volume 74, pages 1765–1766, 1977.
- [4] D. H. Rudy and J. C. Strikwerda. A nonreflecting outflow boundary condition for subsonic Navier–Stokes calculations. *Journal of Computational Physics*, 36:55–70, 1980.
- [5] M. B. Giles. Nonreflecting boundary conditions for Euler equation calculations. *AIAA Journal*, 28:2050–2058, 1990.
- [6] S. R. Allmaras, V. Venkatakrishnan, and F. T. Johnson. Far-field boundary conditions for 2-D airfoils. In *17th AIAA Computational Fluid Dynamics Conference*, 2005.
- [7] R. Zangeneh and C. Ollivier-Gooch. A posteriori stability analysis and improvement for finite volume methods on unstructured meshes. In *55th AIAA Aerospace Sciences Meeting*, 2017.
- [8] R. Zangeneh and C. Ollivier-Gooch. Mesh optimization to improve the stability of finite-volume methods on unstructured meshes. *Computers & Fluids*, 156:590–601, 2017.
- [9] Carmen Campos, Jose E. Roman, Eloy Romero, and Andres Tomas. SLEPc homepage. <http://www.grycap.upv.es/slepc/>, 2011.
- [10] Barry Smith, Satish Balay, Jed Brown, Lisandro Dalcin, Dmitry Karpeev, Matthew Knepley, and Hong Zhang. PETSc home page. <http://www.mcs.anl.gov/petsc>, 2011.
- [11] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.
- [12] T. J. Barth. Aspects of unstructured grids and finite-volume solvers for the Euler and Navier-Stokes equations. In *Lecture Series 1994-05*, Rhode-Saint-Genèse, Belgium, March 1994. von Karman Institute for Fluid Dynamics.
- [13] Carl F. Ollivier-Gooch and Michael Van Altena. A high-order accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *Journal of Computational Physics*, 181(2):729–752, 2002.
- [14] M. B. Giles and N. A. Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion*, 65(3–4):393–415, 2000.