**Seventh International Conference on Computational Fluid Dynamics (ICCFD7), Big Island, Hawaii, July 9-13, 2012**

**ICCFD7-2802**

# High Order Residual Distribution Scheme for the RANS Equations

R. Abgrall*, D. De Santis* and M. Ricchiuto*
Corresponding author: remi.abgrall@inria.fr

* INRIA Bordeaux Sud-Ouest, 351 cours de la Liberation 33405 Talence Cedex - France.

**Abstract:** In this work we describe the use of the Residual Distribution schemes applied to the discretization of conservation laws. In particular, emphasis is put on the construction of a third order accurate scheme. We first recall the properties of a Residual Distribution scheme and we show how to construct a high order scheme for advection problems. Furthermore, we show how to speed up the convergence of the implicit scheme to the steady solution by the means of the Jacobian-free technique. We then extend the scheme to the case of advection-diffusion problems. In particular, we propose a new approach in which the residuals of the advection and diffusion terms are distributed together to get high order accuracy. Due to the continuous approximation of the solution, the gradients of the variables are reconstructed at the nodes and then interpolated on the elements. The numerical scheme is used to discretize the advection-diffusion scalar problem and the compressible Navier-Stokes equations.

*Keywords:* Navier-Stokes equations, High order schemes, Residual distribution, Computational Fluid Dynamics.

## 1 Introduction

The high order discretization of the Navier-Stokes equation is still a challenging task, especially when complex flow regimes are considered like turbulent flows. The first step to obtain an accurate discretization of these problems is the construction of an accurate and robust solver for the laminar Navier-Stokes problems.

In the last years different high order schemes have been considered to obtain high order (more than two) solutions of the Navier-Stokes equations, one of the most attractive scheme seems to be the discontinuous Galerkin (DG) scheme [13]. Residual Distribution (RD) schemes represent an interesting alternative to DG schemes. While computationally compact and probably more flexible, DG schemes suffer from the serious drawback of a very fast growth of the number of degrees of freedom with the cell polynomial degree and also the limiting strategy is not completely clear in the contest of the DG schemes. In RD schemes the formulation remains local, as in DG, but the number of degree of freedom growth less quickly. The price to pay is to impose a continuous approximation of the solution, even though some papers report their extension to discontinuous approximation [8, 2, 16]. Results for RD schemes in the case of order more that two have been presented for the system of the Euler equations in [4], but the high order discretization of the Navier-Stokes equation is still an open question.

One of the first strategy considered [27] for the discretization of the Navier-Stokes equations was the coupling of the RD scheme for the advection with the standard Galerkin discretization for the diffusion, unfortunately the simply combination of the two schemes results in a scheme which is in general only first order accurate, unless advection or diffusion dominated problems are considered. An improved discretization consists of using a blending function to couple the advective and diffusive scheme, and has been use to obtain a third order discretization of the Navier-Stokes equations on triangular meshes [33]. Another approach treats the advection and the diffusion together with the same numerical scheme and has shown to obtain a third order discretization on grids composed by triangles (2D) and tetrahedrons (3D) [12]. In a more recent

approach [24] the advection-diffusion problem is rewritten as a first order, hyperbolic, system in which the unknowns are the variables and their derivatives. This approach has been already considered in the framework of the DG scheme [9], where the use of discontinuous solutions allows one to update explicitly the gradients of the solution without considering additional equations. This approach, although promising, needs further investigations and must still be tested on complex test cases.

In this work we extend the RD scheme proposed in [1] for advection problems to the case of advection-diffusion problems in order to construct a numerical scheme with an uniform order of accuracy for the Navier-Stokes equations. Furthermore the procedure used to imposed the boundary condition is detailed. The paper is organized as follows. In the section 2, we present the basic principle of the RD scheme, and the accuracy properties of the scheme for the scalar advection problem. In the section 3, we extend the RD scheme to the case of the advection-diffusion problem and in the section 4 the ideas introduced for the scalar problem are applied to the case of the system of equations and an efficient iterative procedure for the solution of non linear steady-state problem is presented. In the section 6 are presented the results of the discretization of the scalar advection-diffusion problem in term of the accuracy of the scheme. The proposed scheme is also used to obtain the solution of laminar flows in two and three spatial dimensions. In the last section, we give concluding remarks.

## 2 High order RD schemes for hyperbolic conservation laws

In this section we describe the RD scheme for the discretization of hyperbolic conservation laws in order to report the main properties of the scheme and introduce the nomenclature used through the paper. For simplicity a scalar problem is considered first.

The model equation, for a multidimensional problem, reads

$$\boldsymbol{\nabla}\cdot\boldsymbol{f}(u)=0,\quad \forall\,\boldsymbol{x}\in\Omega\subset\mathbb{R}^d,\ d=2,3 \tag{1}$$

where $u(\boldsymbol{x})\in\mathbb{R}$ and $\boldsymbol{f}\in\mathbb{R}^d$ is the flux function associated to the unknown $u$. The Eq. (1) must be supplemented by the proper inflow boundary conditions

$$u|_{\partial\Omega^-}=g(\boldsymbol{s}),\quad \boldsymbol{s}\in\partial\Omega^-,$$

where $\partial\Omega^-=\{\boldsymbol{x}\in\partial\Omega\,|\,\boldsymbol{n}\cdot\boldsymbol{\nabla}_u\boldsymbol{f}<0\}$, with $\partial\Omega$ boundary of $\Omega$ and $\boldsymbol{n}$ the outward normal vector to the boundary of the domain. The function $g$ is known and represents the weak Dirichlet boundary conditions of the problem on the boundary $\partial\Omega^-$.

The domain $\Omega$ is first discretized with $N_E$ non-overlapping elements with characteristic length $h_E$. The set of all the elements is denoted with $\mathcal{E}_h$. In the RD scheme the DOF are associated with the points of the mesh and not with the control volumes as in the Finite Volume or DG methods. We denote by $\{\sigma_l\}_{l=1,N^E_{\text{dof}}}$ the list of the DOF of the generic element $E$. When a linear interpolation of the solution is used, the DOF of each element coincide with the vertices. The higher order interpolation of the solution, necessary to construct an high order RD scheme, is obtained by adding extra DOF on each element. In order to keep a local formulation, the extra DOF are added inside the element. Since we use a continuous interpolation, all the DOF on the elements boundaries are shared by neighboring elements, this results in a number of DOF smaller than the DG scheme. Clearly, in the case of the continuous approximation the number of DOF increases more rapidly than in the discontinuous one and both cases become asymptotically similar.

To obtain a discrete counterpart of the Eq. (1), the total residual is first computed for each element $E$ of $\mathcal{E}_h$ as

$$\Phi^E=\int_E\boldsymbol{\nabla}\cdot\boldsymbol{f}^h(u^h)\mathrm{d}\Omega=\oint_{\partial E}\boldsymbol{f}^h(u^h)\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega, \tag{2}$$

where $u^h$ and $\boldsymbol{f}^h$ are approximations of $u$ and $\boldsymbol{f}$, respectively. Due to the numerical approximation of the solution, the quantity $\Phi^E$ will be in general not null on each element. In order to handle only nodal values the total residual is distributed to each DOF of the element, the way in which this step is performed characterizes the behavior of the scheme. The quantities distributed to each DOF of the element are indicated for a generic

scheme as $\{\Phi_\sigma^E\}_{\sigma=1,N_{dof}^E}$. It is easy to show [6] that the following conservation constrain must be satisfied

$$\sum_{\sigma\in E}\Phi_\sigma^E = \Phi^E, \quad \forall\, E\in\mathcal{E}_h, \tag{3}$$

we assume that the residuals $\Phi_\sigma^E$ depend continuously on the values of $\{u_\sigma\}_{\sigma\in E}$. Imposing the following constrain

$$\sum_{E\ni\sigma}\Phi_\sigma^E = \mathbf{0}, \quad \forall\,\sigma\in E,\ \forall\, E\in\mathcal{E}_h, \tag{4}$$

we obtain a relation for each DOF. The numerical solution is obtained resolving the resulting non linear system.

If $\sigma$ is a DOF belonging to $\partial\Omega^-$, the boundary conditions must be taken into account into the Eq. (4). We consider a numerical flux $\boldsymbol{f}^\partial$, which depends on the boundary condition $g$, the outward normal $\boldsymbol{n}^\Gamma$ to the boundary, and the local state $u^h$. We define the residual on the boundary as $\Phi_\sigma^\Gamma$, with the following conservation constrain

$$\sum_{\sigma\in\Gamma}\Phi_\sigma^\Gamma = \int_\Gamma \left(\boldsymbol{f}^\partial(u^h,g;\boldsymbol{n}^\Gamma)\cdot\boldsymbol{n}^\Gamma \;-\; \boldsymbol{f}^h(u^h)\cdot\boldsymbol{n}^\Gamma\right)\mathrm{d}\partial\Omega = \Phi_\sigma^\Gamma, \quad \forall\,\Gamma\subset\partial\Omega^-, \tag{5}$$

where $\Gamma$ is an edge (or a face) on $\partial\Omega^-$. The Eq. (4) with the contribution of the boundary conditions becomes

$$\sum_{E\ni\sigma}\Phi_\sigma^E + \sum_{\Gamma\ni\sigma}\Phi_\sigma^\Gamma = \mathbf{0}, \quad \forall\,\sigma\in E,\Gamma \quad\text{and}\quad \forall\, E\in\mathcal{E}_h,\ \forall\,\Gamma\in\partial\Omega^-.$$

It can be shown [6] that if the sequence $u^h$ is bounded in $L^\infty$ when $h\to 0$ and if exists $w$, such that $u^h\to w$ when $h\to 0$, then $w$ is a weak solution of (1). In the proof of this statement one has to assume the continuity of the interpolant across the edges, although this constrains may be alleviated and is possible to define RD schemes on discontinuous elements [8, 2, 16].

## 2.1 Accuracy constraints

On the basis of the work done in [6], are now introduced two important properties that must be considered in the construction of a high-order monotone scheme.

The first property, historically called linearity preserving, establishes the conditions under which the scheme is effectively high order accurate. Assuming $u$ is smooth enough and $u^h$ is its $P^k$ interpolant, and writing the distributed residual as

$$\Phi_\sigma^E = \beta_\sigma^E\Phi^E \qquad\text{and}\qquad \Phi_\sigma^\Gamma = \beta_\sigma^E\Phi^\Gamma,$$

the numerical scheme is accurate with order $h^{k+1}$ providing that

$$\Phi_\sigma^E = \mathcal{O}(h^{k+d}) \qquad\text{and}\qquad \Phi_\sigma^\Gamma = \mathcal{O}(h^{k+d-1}), \tag{6}$$

and $\beta_\sigma^E$ is uniformly bounded.

The second property is related to the monotonicity of the numerical scheme. If we re-write the total residual on the element as

$$\Phi_\sigma^E = \sum_{\sigma'\in E} c_{\sigma\sigma'}^E(u_\sigma - u_{\sigma'}), \tag{7}$$

using this formalism, the Eq. (4), neglecting the boundary conditions, becomes

$$\sum_{E\ni\sigma}\sum_{\sigma'\in E} c_{\sigma\sigma'}^E(u_\sigma - u_{\sigma'}) = 0.$$

In general, the coefficients $c_{\sigma\sigma'}^E$ depend on the solution, which means that the last expression defines a set of

non linear equations that must be solved by an iterative procedure, as for example a Jacobi-like iterations

$$u_\sigma^{n+1} = u_\sigma^n - \delta_\sigma \left( \sum_{E \ni \sigma} \sum_{\sigma' \in E} c_{\sigma\sigma'}^E (u_\sigma - u_{\sigma'}) \right)^n, \tag{8}$$

with $\delta_\sigma$ a relaxation parameter. If the scheme satisfies the following positivity conditions

$$\sum_{E \ni \sigma} \sum_{\sigma' \in E} c_{\sigma\sigma'}^E \geq 0 \quad \forall \sigma, \sigma' \qquad \text{and} \qquad 1 - \delta_\sigma \left( \sum_{E \ni \sigma} \sum_{\sigma' \in E} c_{\sigma\sigma'}^E \right) \geq 0 \quad \forall \sigma, \tag{9}$$

then the solution verifies the following discrete maximum principle

$$\min_{E \ni \sigma} \min_{\sigma' \in E} u_{\sigma'}^0 \leq u_\sigma^n \leq \max_{e \ni \sigma} \max_{\sigma' \in E} u_{\sigma'}^0,$$

with $u_{\sigma'}^0$ the value of the initial solution at the DOF. A scheme that verifies the conditions (9) is said monotonicity preserving. These conditions do not imply that the iterative scheme (8) is convergent, but only that the maximum principle is satisfied, *i.e.*, the $L^\infty$-stability.

It is well know from the Godunov's theorem that a monotonicity preserving scheme with the coefficients $c_{\sigma\sigma'}^T$ that do not depend on the solution can not be linearity preserving [5]. As a consequence, a monotonicity and linearity preserving scheme must be non linear.

A systematic way to construct a non linear scheme which is both linearity and monotonicity preserving has been shown in [1]. The basic idea consists in distributing the total residual to the DOF of each element by the means of a low-order scheme, monotonicity preserving but not linearity preserving, for which the distribution coefficients

$$\beta_\sigma^E = \frac{\Phi^E}{\Phi_\sigma^E},$$

are not bounded. The high-order monotone scheme is constructed by applying a mapping $\beta_\sigma^E \mapsto \hat{\beta}_\sigma^E$, such that the distributions coefficients $\hat{\beta}_\sigma^E$ of the high order scheme are bounded and do not violate the conservation and the monotonicity constrains. The high-order distributed residuals are computed as: $\hat{\Phi}_\sigma^E = \hat{\beta}_\sigma^E \Phi^E$.

## 2.2 Construction of a high order RD scheme

In this sub-section we show how to construct a high order non upwind RD scheme from a first order scheme. The starting point is the Rusanov's scheme [1]

$$\Phi_\sigma^E = \frac{\Phi^E}{N_{\text{DOF}}^E} + \alpha^E (u_\sigma - \bar{u}), \tag{10}$$

with $\bar{u} = \frac{\sum_{\sigma \in E} u_\sigma}{N_{\text{DOF}}^E}$ and $\alpha^E$ a parameter large enough to guaranty the stability of the scheme. Approximating $u$ with its $P^k$ interpolant on the element $E$, $u^h = \sum_{\sigma \in E} u_\sigma \psi_\sigma$, the residual reads

$$\Phi^E = \int_E \boldsymbol{\nabla} \cdot \boldsymbol{f}(u^h) \, \mathrm{d}\Omega = \oint_{\partial E} \boldsymbol{f}(u^h) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega.$$

If we define now $k_\sigma^E = \int_E \boldsymbol{\lambda} \cdot \boldsymbol{\nabla} \psi_\sigma \, \mathrm{d}\Omega$, with $\boldsymbol{\lambda} = \boldsymbol{\nabla}_u \boldsymbol{f}$, it is easy to see that the Eq. (10) can be put in the form of the Eq. (7) with $c_{\sigma\sigma'}^E = \frac{k_\sigma^E - \alpha^E}{N_{\text{DOF}}^E}$, and the condition $c_{\sigma\sigma'}^E \geq 0$ is satisfied if $\alpha^E \geq \max_{\sigma \in E} |k_\sigma^E|$. The scheme is extremely dissipative, but it is very cheap and simple to code and can be easily extended to system case. The high order scheme is constructed from the Eq. (10) applying the non-linear technique previously described, and the distributed high-order residual $\hat{\Phi}_\sigma^E$ is obtained.

The use of a central scheme, like the Rusanov's scheme, in combination with the limiting technique may produce a local downwind scheme which results in not accurate and not convergent scheme. The problem

[1] Other examples can be considered, such as the rephrasing of standard finite volume schemes in term of RD schemes.

has been analyzed [1] and the solution proposed consists in modifying the scheme as follows

$$\Phi_\sigma^{\star,E} = \hat{\Phi}_\sigma^E + \int_E (\boldsymbol{\lambda}\cdot\boldsymbol{\nabla}\psi_\sigma)\,\tau\left(\boldsymbol{\lambda}\cdot\boldsymbol{\nabla}u^h\right)\,\mathrm{d}\Omega, \qquad \tau > 0. \tag{11}$$

The last term on the second member of the previous equation is a streamline dissipation term, used in SUPG schemes to suppress the spurious mode of the Galerkin scheme [10]. The formal accuracy of the scheme is preserved since the filtering term vanishes when $u^h$ is replaced with the exact solution. It is worth noting that the conservation property is still preserved because $\sum_{\sigma\in E}\Phi_\sigma^\star = \Phi^E$ since $\sum_{\sigma\in E}\boldsymbol{\nabla}\psi_\sigma = 0$. Experimentally, we can see that the non oscillatory properties of the scheme are not spoiled.

# 3 Extension to viscous terms

In this section we extend the RD scheme, developed previously for the advection problem, to the case of advection-diffusion problem. The scalar model equation for the advection-diffusion problems reads

$$\begin{aligned}
&\boldsymbol{\nabla}\cdot\boldsymbol{f}(u) = \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}u), \quad \forall\,\boldsymbol{x}\in\Omega\subset\mathbb{R}^d,\ d=2,3\\
&u = g(\boldsymbol{s}), \quad \boldsymbol{s}\in\partial\Omega^-\\
&u = g_0(\boldsymbol{s}), \quad \boldsymbol{s}\in\partial\Omega_0
\end{aligned} \tag{12}$$

where $\nu > 0$ is the viscosity, generally function of $u$ and $\boldsymbol{\nabla}u$. The portion of the boundary $\partial\Omega^-$ is the inflow boundary where the weak boundary conditions are specified and $\partial\Omega_0$ represents the part of the boundary where strong boundary conditions are imposed.

Within the RD framework, the discretization of the viscous terms has been traditionally obtained by coupling the scheme for the advection equations with the Galerkin approximation of the viscous terms [27]. It has been shown [3] that a scheme resulting from the coupling of a RD scheme with a Galerkin discretization of the viscous term is still a residual method, but this is true only for P1 elements. Furthermore it is well know [26] that the scheme obtained as the sum of the RD scheme and the Galerkin scheme is second order accurate in the diffusion and advection limits, but it is only first order accurate when advection and diffusion are equally important. One way to construct a scheme with an uniform order of accuracy has been proposed in [33], where the Galerkin scheme for the diffusion is added to the LDA scheme for the advection using a blending function.

A different approach [22, 24] is based on the idea that the steady advection-diffusion equation, Eq. (12), is equivalent to a hyperbolic relaxation system at the steady state. The reformulation of the original advection-diffusion equation as an hyperbolic system, allows one to use all the techniques already known in the field of the hyperbolic system of conservation laws to obtain a solution of the advection-diffusion problem with an uniform order of accuracy on both the solution and its gradient.

We follow an alternative way to discretize the advection-diffusion equation in the framework of the RD scheme. In order to preserve the order of accuracy of the scheme, the total residual must be take into account both the advective and diffusive parts, and a single distribution scheme must be used to distribute the total residual. This idea has been introduced in [12], where the LDA scheme is used to distribute the total residual consisting of the advective and the diffusive parts. A third order accurate solution is obtained by considering the Hermite cube interpolation along the edge of the elements (triangles or tetrahedrons). This scheme has been also considered in [26] where the distribution coefficients of the LDA scheme are modified in a such a way that they make the scheme isotropic in the diffusion limit.

## 3.1 Construction of the RD scheme for the advection-diffusion problem

The first step in the construction of a RD scheme is the computation of the total residual on each element of the mesh. As explained previously, the total residual must contains both the advective and diffusive parts, namely

$$\Phi^E = \int_E \Big(\boldsymbol{\nabla}\cdot\boldsymbol{f}(u) - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}u)\Big)\mathrm{d}\Omega,$$

thanks to the divergence theorem, the previous equation can be recasted as

$$\Phi^E = \oint_{\partial E} \left( \boldsymbol{f}(u) - \nu \widetilde{\boldsymbol{\nabla} u} \right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega, \tag{13}$$

where $\widetilde{\boldsymbol{\nabla} u}$ is the reconstructed gradient of the solution. The use of the reconstructed gradient is necessary because the normal flux to each face of the element must be continuous and in general the components of gradients normal to the element's face are discontinuous across the face shared by two neighboring elements. Different options are available for the gradient reconstruction as, for example, the Green-Gauss formula or the Least-Square procedure.

The total residual is distributed to each DOF of the elements according to the procedure described in the sub-section 2.2. The high order distributed residual for the advection-diffusion equation reads

$$\Phi_\sigma^{\star,E} = \hat{\Phi}_\sigma^E + \int_E \left( \boldsymbol{\lambda} \cdot \boldsymbol{\nabla} \psi_\sigma \right) \tau \left( \boldsymbol{\lambda} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) \right) \mathrm{d}\Omega, \qquad \tau > 0. \tag{14}$$

In the vanishing viscosity limit, the scheme reduces to the scheme (11) developed for the advection problems.

From numerical experiments it has been observed that the previous scheme it not satisfactory because it is characterized by a large numerical error and by a not uniform order of accuracy. A modification of the original scheme has been inspired by the work of Nishikawa [25], where an accurate viscous discretization is obtained by rewriting the diffusion equation as a hyperbolic first order system which is discretized by a scheme suitable for advection problems. From the discretization of the first order scheme a discrete counterpart of the viscous term can be extracted and applied back to the original advection-diffusion equation. As detailed in the reference [23], when the Lax-Wendroff scheme is applied to the diffusion scalar equation the following scheme is obtained for the case of $P1$ elements

$$\Phi_\sigma^E = \frac{\Phi^E}{3} + \frac{\nu\delta}{4} \left( \boldsymbol{\nabla} u^E - \widetilde{\boldsymbol{\nabla} u}^E \right) \cdot \boldsymbol{n}_\sigma^E, \tag{15}$$

where $\boldsymbol{n}_\sigma^E$ is the normal vector to the face opposed to the node $\sigma$ scaled by the length of the face itself, and we set the parameter $\delta = 1$. The term $\left( \boldsymbol{\nabla} u^E - \widetilde{\boldsymbol{\nabla} u}^E \right)$ represents the difference, on the element, between the gradient of the solution and the reconstructed gradient. In this work the scheme (15) is written in a more general way as follows

$$\Phi_\sigma^E = \hat{\Phi}_\sigma^E + \int_E \frac{\nu\delta}{2} \left( \boldsymbol{\nabla} u^E - \widetilde{\boldsymbol{\nabla} u}^E \right) \cdot \boldsymbol{\nabla} \psi_\sigma^E \mathrm{d}\Omega, \tag{16}$$

where $\Phi_\sigma^E$ is, as usual, the distributed residual and the integral term vanish when summed over the DOF of the element, thus it does not contribute to the total residual and it acts only as a damping term in the viscous discretization scheme.

The schemes (14) and (16) can be combined together as follows

$$\begin{aligned} \Phi_\sigma^{\star,E} = \hat{\Phi}_\sigma^E &+ \xi(Re) \int_E \left( \boldsymbol{\lambda} \cdot \boldsymbol{\nabla} \psi_\sigma \right) \tau \left( \boldsymbol{\lambda} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) \right) \mathrm{d}\Omega \\ &+ \left( 1 - \xi(Re) \right) \int_E \frac{\nu\delta}{2} \left( \boldsymbol{\nabla} u^E - \widetilde{\boldsymbol{\nabla} u}^E \right) \cdot \boldsymbol{\nabla} \psi_\sigma^E \, \mathrm{d}\Omega, \end{aligned} \tag{17}$$

where $Re_E = \frac{\|\boldsymbol{\lambda}\| h_E}{\nu}$ is the Reynolds number on the element, with $h_E$ characteristic length of the element, and the function $\xi(Re)$ is such that $\xi(Re) \to 0$ in the diffusion limit ($Re \to 0$) and $\xi(Re) \to 1$ in the advection limit ($Re \to \infty$).

### 3.2 Gradient reconstruction

The evaluation of the total residual for the advection-diffusion problem requires a gradient reconstruction in order to have a continuous normal flux at each face. Different techniques have been used, the easiest one is

the Green-Gauss formula

$$\widetilde{\boldsymbol{\nabla} u}_{|\sigma} = \frac{\displaystyle\sum_{E \ni \sigma} \boldsymbol{\nabla} u^E_{|\sigma} \, |E|}{\displaystyle\sum_{E \ni \sigma} |E|}.$$

Another approach is the Least-Square technique, in which the gradient reconstruction is obtained by solving for the values of the gradients that minimize the following function

$$\sum_{k=1}^{N} E_{ik}^2, \qquad \forall i$$

with

$$E_{ik}^2 = \left( -\Delta u_{ik} + \left.\frac{\partial u}{\partial x}\right|_i \Delta x_{ik} + \left.\frac{\partial u}{\partial y}\right|_i \Delta y_{ik} + \left.\frac{\partial^2 u}{\partial^2 x}\right|_i \Delta x_{ik}^2 + \left.\frac{\partial^2 u}{\partial^2 y}\right|_i \Delta y_{ik}^2 + \left.\frac{\partial^2 u}{\partial x \partial x}\right|_i \Delta x_{ik} \Delta y_{ik} \right)^2, \tag{18}$$

with $\Delta u_{ik} = u(x_k, y_k) - u(x_i, y_k)$, $\Delta x_{ik} = x_k - x_i$, $\Delta y_{ik} = y_k - y_i$, and where $i$ is the node where the gradient is reconstructed and $k = 1 \dots N$ are the neighboring nodes of $i$. The resulting linear system is over-determined and must be solver be the means of a Least-Square procedure. In the case of $P1$ elements only the first order derivatives are considered while in the case of $P2$ elements also the second order derivatives are considered in the Eq. (18)

The last technique considered is the L2 projection in which the reconstructed gradient is obtained solving the following equivalence $\boldsymbol{\nabla} u = \widetilde{\boldsymbol{\nabla} u}$ in a weak sense

$$\int_\Omega \psi_i \widetilde{\boldsymbol{\nabla} u} \, \mathrm{d}\Omega = \int_\Omega \psi_i \boldsymbol{\nabla} u \, \mathrm{d}\Omega, \qquad \forall i.$$

By expanding the gradient as $\boldsymbol{\nabla} u \simeq \sum_k \psi_k \widetilde{\boldsymbol{\nabla} u_k}$, ones obtain the following linear system: $M\{\widetilde{\boldsymbol{\nabla} u}\} = b$, with

$$M_{ij} = \int_\Omega \psi_i \psi_k \, \mathrm{d}\Omega, \quad \text{and} \quad b_i = \int_\Omega \psi_i \boldsymbol{\nabla} u \, \mathrm{d}\Omega.$$

# 4 Discretization of system of conservation laws

We consider now the extension of the RD scheme, developed for the scalar advection equation, to the case of the system, namely

$$\boldsymbol{\nabla} \cdot \mathbf{f}^a(\mathbf{u}) = \mathbf{0}, \tag{19}$$

where $\mathbf{u}(\boldsymbol{x}) \in \mathbb{R}^p$, with $p$ the number of unknowns and $\mathbf{f} = (\mathsf{f}_1, \dots, \mathsf{f}_d)$ is the flux function associated to $\mathbf{u}$, with $\mathsf{f}_{i=1,d} \in \mathbb{R}^p$. In the case of the system of the Euler equations for steady flows, written in conservative form, the vector unknown is the vector of the conservative variables density, momentum and total energy per unit volume

$$\mathbf{u} = (\rho, \boldsymbol{m}, E^t)^{\mathrm{T}},$$

while the the flux function is

$$\mathbf{f}^a(\mathbf{u}) = \begin{pmatrix} \boldsymbol{m} \\ \dfrac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P\mathbb{I} \\ (E^t + P)\dfrac{\boldsymbol{m}}{\rho} \end{pmatrix},$$

where $P$ is the pressure and $\mathbb{I} \in \mathbb{R}^{d \times d}$ is the identity matrix. The thermodynamic law of the polytropic ideal gas is used, namely

$$P = (\gamma - 1) \left( E^t - \frac{\|\boldsymbol{m}\|^2}{2\rho} \right).$$

The advection-diffusion problem for a system of equations with the vector unknown $\mathbf{u}$ can be written as follows

$$\frac{\partial \mathbf{u}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{f}^a(\mathbf{u}) = \boldsymbol{\nabla} \cdot \mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u}), \tag{20}$$

with $\mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u})$ the diffusion flux which generally depends on $\mathbf{u}$ and $\boldsymbol{\nabla}\mathbf{u}$. In the case of Navier-Stokes equations $\mathbf{u}$ and $\mathbf{f}^a(\mathbf{u})$ are the vector of the conservative variables and the advective flux function, respectively, as defined for the Euler equations, while $\mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u})$ is the viscous flux function, defines as

$$\mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u}) = \begin{pmatrix} 0 \\ \mathbb{S} \\ \mathbb{S} \cdot \dfrac{\boldsymbol{m}}{\rho} - \boldsymbol{q} \end{pmatrix},$$

where $\boldsymbol{q} = -\kappa \boldsymbol{\nabla} T$ is the heat flux, with $T$ the temperature and $\kappa$ the thermal conductivity, and $\mathbb{S}$ is the viscous stress tensor

$$\mathbb{S} = -\frac{2}{3}\mu \boldsymbol{\nabla} \cdot \left(\frac{\boldsymbol{m}}{\rho}\right) \mathbb{I} + \mu \left( \boldsymbol{\nabla}^{\mathrm{T}} \left(\frac{\boldsymbol{m}}{\rho}\right) + \boldsymbol{\nabla} \left(\frac{\boldsymbol{m}}{\rho}\right) \right),$$

with $\mu$ the fluid viscosity. It is well know that the viscous flux function $\mathbf{f}^v$ is homogeneous with respect to the gradient of the conservative variables $\boldsymbol{\nabla}\mathbf{u}$

$$\mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u}) = \mathbb{K}(\mathbf{u}) \boldsymbol{\nabla}\mathbf{u},$$

with the homogeneity tensor $\mathbb{K}(\mathbf{u}) = \dfrac{\partial \mathbf{f}^v(\mathbf{u}, \boldsymbol{\nabla}\mathbf{u})}{\partial \boldsymbol{\nabla}\mathbf{u}}$.

The discretization of the system of equations can be deduced in a straightforward manner from the corresponding scalar problem, in particular for the advection-diffusion problem the total residual on a generic element $E$ reads

$$\boldsymbol{\Phi}^E = \oint_e \left( \mathbf{f}^a(\mathbf{u}) - \mathbb{K}(\mathbf{u}) \widetilde{\boldsymbol{\nabla}\mathbf{u}} \right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega,$$

with $\widetilde{\boldsymbol{\nabla}\mathbf{u}}$ the reconstructed gradient of the conservative variables. The total residual is first distributed to all the DOF of the element using the low order Rusanov scheme, from that a non-linear monotone scheme is constructed and the filtering term is added together with the dumping term acting for the viscous part. The schemes shown in the section 3 for the scalar problem applies in the same way to the system, namely

$$\begin{aligned} \boldsymbol{\Phi}_\sigma^{\star,E} = \hat{\boldsymbol{\Phi}}_\sigma^E &+ \xi(Re) \int_E \left( \boldsymbol{A} \cdot \boldsymbol{\nabla}\phi_\sigma \right) \boldsymbol{\tau} \left( \boldsymbol{A} \cdot \boldsymbol{\nabla}\mathbf{u}^h - \boldsymbol{\nabla} \cdot \left( \mathbb{K} \boldsymbol{\nabla}\mathbf{u} \right) \right) \mathrm{d}\Omega \\ &+ \left( 1 - \xi(Re) \right) \int_E \tfrac{1}{2} \mathbb{K} \left( \boldsymbol{\nabla} u^E - \widetilde{\boldsymbol{\nabla} u}^E \right) \cdot \boldsymbol{\nabla} \psi_\sigma^E \, \mathrm{d}\Omega. \end{aligned}$$

## 4.1 Boundary conditions

The imposition of the boundary conditions, in the RD framework, has not reached yet a mature level of understanding, for that reason we shown here the way followed to impose the boundary conditions in the case of inviscid and viscous problems.

We propose a flexible way to distribute the residual computed with the corrections fluxes on the boundary elements, that is generalizable to all the spatial dimensions and does not depend on arbitrary parameters. Let us consider the weak form of the conservation law

$$\int_\Omega \psi \boldsymbol{\nabla} \cdot \mathbf{f}(\mathbf{u}^h) \, \mathrm{d}\Omega = 0,$$

an integration by parts gives

$$\int_{\partial\Omega} \mathbf{f}(\tilde{\mathbf{u}}) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega - \int_\Omega \boldsymbol{\nabla}\psi \cdot \mathbf{f}(\mathbf{u}^h) \, \mathrm{d}\Omega = 0,$$

where $\tilde{\mathbf{u}}$ represent the state that take into account the boundary conditions. Replacing the basis function $\psi$ with the standard Lagrangian basis functions $\psi_i, \forall\, i \in \Omega$, and applying again the integration by parts to the second member on the left hand side of the previous equation one obtains

$$\int_{E_i} \psi_i \cdot \mathbf{f}(\mathbf{u}^h), \mathrm{d}\Omega + \int_{\partial E_i \cap \partial\Omega} \psi_i \big(\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u}^h)\big) \cdot \boldsymbol{n}\, \mathrm{d}\partial\Omega = 0,$$

where $E_i$ is the set of the elements that have the node $i$ in common. The boundary integral on the previous equation can be regarded as the contribution that must be added to the residuals computed on the domain elements without considering the boundary conditions. Note that the correction flux, $\mathbf{f}(\tilde{\mathbf{u}}) - \mathbf{f}(\mathbf{u}^h)$, becomes null as the boundary conditions are correctly enforced into the numerical scheme.

In the case of an a inviscid flow of a solid wall, the boundary conditions require that the normal component of the velocity to the wall is null: $\boldsymbol{v} \cdot \boldsymbol{n} = 0$, with $\boldsymbol{v}$ the velocity vector. The flux function and the correction flux read

$$\mathbf{f}(\tilde{\mathbf{u}}_{\text{wall}}) = \begin{pmatrix} 0 \\ Pn_x \\ Pn_y \\ Pn_z \\ 0 \end{pmatrix}, \qquad \mathbf{f}(\tilde{\mathbf{u}}_{\text{wall}}) - \mathbf{f}(\mathbf{u}^h) = -v_n \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ E^t + P \end{pmatrix},$$

respectively, with $v_n = \boldsymbol{v} \cdot \boldsymbol{n}$.

The inflow/outflow boundary conditions can be imposed specifying the state $\tilde{\mathbf{u}}_{\text{in/out}}$. As standard practice the flux function is linearized as

$$\mathbf{f}(\tilde{\mathbf{u}}_{\text{in/out}}) \simeq A_n^+(\mathbf{u}^h) + A_n^-(\tilde{\mathbf{u}}_{\text{in/out}}),$$

where $A_n^{\pm}(\mathbf{u}) = R(\mathbf{u}, \boldsymbol{n})\, \Lambda^{\pm}(\mathbf{u}, \boldsymbol{n})\, L(\mathbf{u}, \boldsymbol{n})$, with $R(\mathbf{u}, \boldsymbol{n}), L(\mathbf{u}, \boldsymbol{n})$ the matrices of right and left eigenvectors respectively and $\Lambda(\mathbf{u}, \boldsymbol{n})^{\pm}$ the diagonal matrix of the eigenvalues. The operator $\pm$ selects the positive and the negative eigenvalues and $\boldsymbol{n}$ is the outward normal versor to the boundary face. The correction flux for the imposition of the inflow/outflow boundary conditions reads

$$\mathbf{f}(\tilde{\mathbf{u}}_{\text{in/out}}) - \mathbf{f}(\mathbf{u}^h) = A_n^-(\mathbf{u}^h)(\tilde{\mathbf{u}}_{\text{in/out}} - \mathbf{u}^h).$$

The same procedure used to impose the inflow/outflow boundary conditions can be used to specify the far-field boundary conditions, in this case usually the value of pressure at far-field boundary is specified.

For a viscous fluid, at the solid surface we apply the adiabatic no-slip boundary condition, which require that the velocity and normal heat flux on the wall are zero: $\boldsymbol{v}|_{\text{wall}} = 0$, $\boldsymbol{q} \cdot \boldsymbol{n}|_{\text{wall}} = 0$. In the RD scheme the adiabatic, no-slip boundary conditions are generally applied by initializing the velocity field such that the velocity components are zero on the wall and enforcing the residual associated to the momentum equation to be also zero on the boundary nodes at each iteration, this corresponds to impose strongly the boundary conditions for the momentum equation. Nothings is explicitly done for the residual associated to the energy equation, this is partially justified by the fact that on the wall

$$\int_{\partial E_i \cap \partial\Omega} (\boldsymbol{v} \cdot \mathbb{S} - \boldsymbol{q}) = -\int_{\partial E_i \cap \partial\Omega} \boldsymbol{q},$$

because the velocity is zero, so if the previous integral is not computed this is equivalent to assume that $\boldsymbol{q}$ is zero on the wall. However this consideration is not true in practice, and imposing the boundary condition in such a way does not guarantee that the heat flux is zero on the wall.

We propose an alternative way to impose the boundary condition for the energy equation in a weak sense, which uses the correction flux technique introduced in the inviscid case. Since the velocity is zero on the wall, the flux function in the direction normal to the wall becomes

$$\mathbf{f} \cdot \boldsymbol{n}_w = \begin{pmatrix} 0 \\ \mathbb{S} \cdot \boldsymbol{n}_w \\ -\boldsymbol{q} \cdot \boldsymbol{n}_w \end{pmatrix},$$

with $\boldsymbol{n}_w$ the unit vector directed along the normal direction to the wall. In order to have zero heat flux on the wall, a correction flux can be considered. This is exactly the same way how slip boundary conditions are imposed in inviscid simulations. The correction flux on the wall is

$$\mathbf{f}^{\text{wall}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{q}\cdot\boldsymbol{n} \end{pmatrix}.$$

The components of the heat flux can be easily computed, with no extra effort, from the gradient of the conservative variables (already available from the computation). The heat flux for a perfect gas can be written as

$$\boldsymbol{q} = -k\boldsymbol{\nabla}T = -k\frac{\gamma-1}{\mathcal{R}}\boldsymbol{\nabla}e,$$

since for a perfect gas $T = \frac{\gamma-1}{\mathcal{R}}e$, where $\gamma$ is the specif heats ratio, $\mathcal{R}$ is the gas constant and $e$ is the internal energy: $e = \frac{E^t}{\rho} - \frac{1}{2}\|\boldsymbol{v}\|^2$. Considering, for example the $x$ component of a two dimensional flow, it is easy to see that

$$\begin{aligned}
\frac{\partial e}{\partial x} &= \frac{\partial}{\partial x}\left(\frac{E^t}{\rho} - \frac{1}{2}\|\boldsymbol{v}\|^2\right) \\
&= \frac{\partial}{\partial x}\left(\frac{E^t}{\rho}\right) - \frac{1}{2}\left(\frac{\partial}{\partial x}\left(\frac{m_x^2}{\rho^2}\right) + \frac{\partial}{\partial x}\left(\frac{m_y^2}{\rho^2}\right)\right) \\
&= \frac{1}{\rho}\frac{\partial E^t}{\partial x} - \frac{E^t}{\rho^2}\frac{\partial \rho}{\partial x} - \underbrace{\left(\frac{m_x}{\rho^2}\frac{\partial m_x}{\partial x} - \frac{m_x^2}{\rho^3}\frac{\partial \rho}{\partial x} + \frac{m_y}{\rho^2}\frac{\partial m_y}{\partial x} - \frac{m_y^2}{\rho^3}\frac{\partial \rho}{\partial x}\right)}_{=0,\text{wall}} \\
&= \frac{1}{\rho}\frac{\partial E^t}{\partial x} - \frac{E^t}{\rho^2}\frac{\partial \rho}{\partial x},
\end{aligned}$$

with $m_x = \rho v_x$ and $m_y = \rho v_y$, the components of the momentum, which are zero on the wall. The heat flux as function of the conservative variables can be written as

$$\boldsymbol{q} = -\frac{k}{\rho}\frac{\gamma-1}{\mathcal{R}}\left(\boldsymbol{\nabla}E^t - \frac{E^t}{\rho}\boldsymbol{\nabla}\rho\right)$$

Now, it is possible to compute the residual on the wall nodes, as

$$\int_{\partial E_i \cap \partial\Omega} \psi_i\Big(\mathbf{f}^{\text{wall}} - \mathbf{f}(\mathbf{u}^h)\Big)\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega.$$

As usual the residual associated to the momentum equation is explicitly imposed to be zero in order to apply the strong boundary conditions for the velocity on the wall.

## 4.2 Curved elements

The main advantage of high order methods over the low order ones is the capability of the high order methods to achieve an higher level of accuracy given the same computational resources. Obviously, since an high order method introduces more degrees of freedom than a lower one, a fair comparison between high and low order methods must be done with the same number of degrees of freedom. In other words the computational grids used by high order methods should be coarser than those used by low order methods. In generating coarse grids for an high order RD scheme, it is critical that curved boundaries are represented with high fidelity otherwise the error produced by a low-fidelity representation of a curved boundary may nullify the benefits of the high order discretization of the governing equations.

In this work the geometry of the high order elements is defined by the following isoparametric transfor-

mation from the reference space, $\widehat{E}$, to the physical space, $E$

$$\boldsymbol{x}(\boldsymbol{\xi}) = \sum_{i=1}^{N_{\mathrm{DOF}}^E} \boldsymbol{x}_i \widehat{\psi}_i(\boldsymbol{\xi}) \quad \forall\, \boldsymbol{\xi} \in \widehat{E}, \tag{21}$$

where $\boldsymbol{\xi}$ is the independent variable in the reference space and $\boldsymbol{x}_i$ are the know coordinates of the nodes defined in the physical space. Note that the nodes of the element edges belonging to the boundary $\partial\Omega$ are places on the real geometry of the boundary. A curved boundary is therefore approximated by piecewise polynomial curved segments. The use of Lagrangian function guarantees the geometric continuity between neighboring elements, Figure 1.

The integral that defines the total residual is computed as the sum of the integrals on each face of the element

$$\oint_{\partial E} \mathbf{f}(\mathbf{u}^h) \cdot \boldsymbol{n}\, \mathrm{d}\partial\Omega = \sum_{f=1}^{N_f^E} \int_{\partial\Omega_f} \mathbf{f}(\mathbf{u}^h) \cdot \boldsymbol{n}\, \mathrm{d}\partial\Omega,$$

where $N_f^E$ is the number of faces of the element $E$. The integral on each face is approximated with a quadrature formula

$$\int_{\partial\Omega_f} \mathbf{f}(\mathbf{u}^h) \cdot \boldsymbol{n}\, \mathrm{d}\partial\Omega \simeq \sum_{q=1}^{Nq} \omega_q\, \mathbf{f}(\mathbf{u}_q) \cdot \boldsymbol{n}_q\, J_q,$$

where $N_q$ is the number of quadrature points, $\omega_q$ are the quadrature weights, $\boldsymbol{n}_q$ is the normal versor to the face at the quadrature point $q$ and $Jq$ is the determinant of Jacobian of the transformation from the reference to the physical face, evaluated at the quadrature point $q$. The quadrature formula must take into account that $\boldsymbol{n}$, for non-linear elements is not constant on the faces, it is a polynomial one order lower than the transformation polynomials (21).

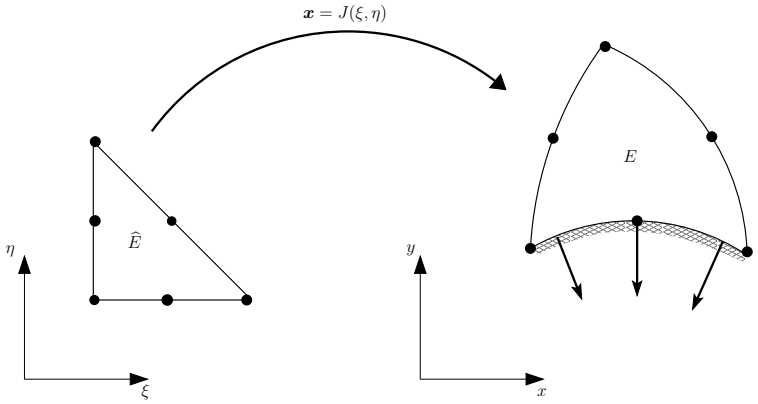



Figure 1: Visual representation of the mapping from the reference element the physical element with curved faces. On the boundary the normal vectors to the face at the quadrature points are also shown.

# 5 Solution of the non linear system

The discrete counterpart of the system of equations (19) or (20) is obtained by assembling for each DOF $\sigma$ the contributions from all the elements $E \ni \sigma$ plus the contribution from the boundary conditions, namely

$$\sum_{E \ni \sigma} \boldsymbol{\Phi}_\sigma^E(\mathbf{u}^h) + \sum_{\Gamma \ni \sigma} \boldsymbol{\Phi}_\sigma^\Gamma(\mathbf{u}^h, \tilde{\mathbf{u}}, \boldsymbol{n}) = \mathbf{0}, \ \forall \sigma. \tag{22}$$

The previous equations represent a system of non linear equations in the form

$$\boldsymbol{F}(\mathbf{u}) = \mathbf{0}, \tag{23}$$

to be solved by the means of an iterative process. Among all kinds of methods for solving a non linear system of equations, the Newton's method is one of the most popular and has a local quadratic convergence. The general form of the Newton's method for solving (23) is

$$\mathbf{u}_{k+1} = \mathbf{u}_k - J(\mathbf{u}_k)^{-1}\boldsymbol{F}(\mathbf{u}_k), \quad k = 0, 1, 2, \dots \tag{24}$$

where $\mathbf{u}_0$ is an initial guess of the solution and $J(\mathbf{u}_k) = \frac{\partial \boldsymbol{F}(\mathbf{u}_k)}{\partial \mathbf{u}_k}$, the Jacobian of $\boldsymbol{F}$, is non singular at each iteration. In practice, the Newton iteration (24) is implemented by the following two steps

$$J(\mathbf{u}_k)\Delta\mathbf{u}_k = -\boldsymbol{F}(\mathbf{u}_k)$$

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta\mathbf{u}_k$$

Usually the problem (22) is replaced by a pseudo-transient one and the steady solution is the limit, for the time variable that tends to infinity, of the pseudo-unsteady problem

$$\sum_{E\ni\sigma} \boldsymbol{\Phi}_\sigma^E(\mathbf{u}^h) + \sum_{\Gamma\ni\sigma} \boldsymbol{\Phi}_\sigma^\Gamma(\mathbf{u}^h, \tilde{\mathbf{u}}, \boldsymbol{n}) = \mathbf{0}, \qquad \longrightarrow \qquad |C_\sigma|\frac{\partial \mathbf{u}_\sigma^h}{\partial t} + \sum_{E\ni\sigma} \boldsymbol{\Phi}_\sigma^E(\mathbf{u}^h) + \sum_{\Gamma\ni\sigma} \boldsymbol{\Phi}_\sigma^\Gamma(\mathbf{u}^h, \tilde{\mathbf{u}}, \boldsymbol{n}) = \mathbf{0},$$

with $|C_\sigma|$ the area of the dual cell associated to the DOF $\sigma$. The presence of the time derivative enables a better convergence of the Newton's method, overcoming the harsh start-up phase when the solution is far from an optimal initial guess. Furthermore, the Jacobian associated to the modified problem is better conditioned than the Jacobian of the original problem during the start-up phase. The pseudo-transient scheme can be written in the following form

$$|C|\frac{\partial \mathbf{u}}{\partial t} = -\boldsymbol{F}(\mathbf{u}).$$

When the Backward Euler formula is used for the discretization of the time derivative, the fully discrete counterpart of the previous system reads

$$|C|\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t^n} = -\boldsymbol{F}(\mathbf{u}^{n+1}), \qquad n = 0, 1, 2, \dots$$

where $n$ is number of the time steps and $\mathbf{u}^0$ is the initial value of the solution. For each time step $n$ a non linear problem should be solved with the Newton's method, namely

$$\left[\frac{|C|}{\Delta t^n} I + J(\mathbf{u}_k^n)\right]\Delta\mathbf{u}_k^n = -\boldsymbol{F}(\mathbf{u}_k^n) \tag{25}$$

$$\mathbf{u}_{k+1}^n = \mathbf{u}_k^n + \Delta\mathbf{u}_k^n, \qquad k = 0, 1, 2, \dots, \tag{26}$$

with $I$ the identity matrix. In practice at each time step only one Newton iteration is performed.

The parameter $\Delta t^n$ is the discrete time step, with $\Delta t^n \to \infty$ as $n \to \infty$. Note that for $\Delta t^n \to \infty$ the iteration of the original Newton's method (24) is retrieved. The evolution of the time step is controlled by the CFL number which is chosen according to the following law [32]

$$\mathrm{CFL}^n = \mathrm{CFL}^{n-1}\frac{\|\boldsymbol{F}(\mathbf{u}^{n-1})\|_\infty}{\|\boldsymbol{F}(\mathbf{u}^n)\|_\infty}, \tag{27}$$

starting from a low CFL number. The iterative process is stopped when the residual of the equations becomes small enough respect to the initial residual.

At each time step the linear system (25) must be solved which is recast for convenience in the following form: $A(\mathbf{u}^n)\Delta\mathbf{u}^n = -\boldsymbol{F}(\mathbf{u}^n)$. The matrix $A$ is non symmetric and has dimension $N \times N$ with $N = N_{\text{DOF}} \times p$, so the number of the non-zero elements can be very high. Krylov methods can be used to solve this class of linear systems, in particular the GMRES [29] is widely used. This method has the property of minimizing the $L_2$-norm of the residual over all vectors in the Krylov subspace. To accelerate the convergence of the iterative linear solver, preconditioning of the matrix $A$ is used. This consists in solving a modified linear system

$$AP^{-1}P\Delta\mathbf{u} = -\boldsymbol{F},$$

with $P$ a preconditioning matrix. When the right preconditioning is used, one first solves

$$AP^{-1}\boldsymbol{w} = -\boldsymbol{F},$$

for $\boldsymbol{w}$, and the solves

$$\Delta\mathbf{u} = P^{-1}\boldsymbol{w},$$

for $\Delta\mathbf{u}$. Only $P^{-1}$ is required.

Complete solving of the linear system is unnecessary for the convergence of the scheme, usually inexact Newton's method is used to reduce the computational effort and avoid over-solving of the system [15]. The linear system is solved until

$$\|\boldsymbol{F}(\mathbf{u}^n) + A(\mathbf{u}^n)\Delta\mathbf{u}^n\| \leq \eta^n \|\boldsymbol{F}(\mathbf{u}^n)\| \tag{28}$$

with a tolerance $\eta^n < 1$. The construction of the matrix $A$ requires to compute the Jacobian of $\boldsymbol{F}$. In order to obtain the quadratic convergence of Newton's method, the linearization of the residuals must be exact. Unfortunately, explicit formation of the Jacobian matrix resulting from the linearization of the high order residuals is extremely expensive, if not impossible. The Jacobian of the low order residual is generally used in the construction of the matrix $A$, but the quadratic convergence of the Newton's method is lost due to the inconsistency between the right hand side of the linear system, constructed with the high order residuals, and the matrix, constructed with the low order residuals.

Since in the GMRES algorithm the matrix $A$ is only required in a matrix-vector product and remembering that the matrix contains the Jacobian of the numerical scheme, one can approximate the product of the matrix $A$ with a generic vector $\boldsymbol{w}$ as follows [31, 11, 18]

$$\begin{aligned} A\boldsymbol{w} &= \left(\frac{|C|}{\Delta t^n} I + J(\mathbf{u}_k^n)\right)\boldsymbol{w} \\ &\simeq \frac{|C|}{\Delta t^n}\boldsymbol{w} + \frac{\boldsymbol{F}(\mathbf{u}^n + \epsilon\boldsymbol{w}) - \boldsymbol{F}(\mathbf{u}^n)}{\epsilon}, \end{aligned}$$

with $\epsilon$ a small step size chosen as [28]

$$\epsilon = \frac{\sqrt{1 + \|\mathbf{u}\|_{L_2}}}{\|\boldsymbol{z}\|_{L_2}}\,\epsilon_{\text{rel}}, \qquad \epsilon_{\text{rel}} = 10^{-8}.$$

Since there is no need to compute explicitly the Jacobian, this approach is called Jacobian-free. However, a rough approximation of the Jacobian is always computed at each step, this matrix is used as a preconditioner in the GMRES algorithm. In other works [20, 21] the Jacobian-free technique is implemented in a full matrix-free version.

The use a Jacobian-free technique without preconditioning usually leads to a lack of convergence on the iterative process. Among all the type of preconditioning developed for CFD applications, we consider in this work the Jacobi and the LU-SGS [17] preconditioning. They represent a good compromise between robustness and memory requirement, and guarantee a good scalability in parallel algorithms. In the Jacobi preconditioning only the diagonal block of the approximated Jacobian are retained, while in the LU-SGS preconditioning the preconditioning matrix is taken as

$$P = (L + D)D^{-1}(D + U),$$

where $D$, is the diagonal block of the approximated Jacobian, $L$ and $U$ are respectively the lower and the

upper parts.

The Jacobian-free approach allows quadratic convergence of Newton's method because the matrix of the linear system is a complete linearization of the residual vector. The price to pay for using this technique is an increment of the computational effort, because at each time step it is necessary to compute several times the residual $\boldsymbol{F}(\mathbf{u})$ on the whole domain. This is largely compensated by a drastic diminution of the iterations number [7].

# 6 Numerical results

In this section, the RD scheme shown in the previous sections, is used first to obtain the discretization of the simple scalar advection-diffusion problem for which an exact solution exists so that the numerical accuracy of the scheme can determined. Subsequently the scheme is also applied to the discretization of the Navier-Stokes equations. In particular two test cases are considered: the standard two dimensional laminar flow over a flat plate and then the more challenging three dimensional flow over a delta wing. All the simulations are considered to be converged when the initial residual is reduced by ten orders of magnitude.

## 6.1 Scalar advection-diffusion

To test the accuracy of the discretization of the advection-diffusion equations, the following two-dimensional scalar problem is solved with the RD scheme:

$$\boldsymbol{a}\cdot\boldsymbol{\nabla}u = \nu\boldsymbol{\nabla}\cdot\big(\boldsymbol{\nabla}u\big), \qquad \text{in} \quad \Omega=[0,1]\times[0,1]$$

with $\boldsymbol{a}=\big(a_x,\,a_y\big)^{\mathrm{T}}$ and the viscosity, $\nu$, constant. The exact solution reads

$$u = -\cos(2\pi\eta)\exp\left(\frac{\xi\left(1-\sqrt{1+16\pi^2\nu^2}\right)}{2\nu}\right),$$

with $\eta = a_y x - a_x y$ and $\xi = a_x x + a_y y$. Here, $a=(0,1)^{\mathrm{T}}$ and different values of $\nu$ are considered. The problem has been discretized by the schemes presented in section 3.1, namely Eq. (14) and Eq. (17), furthermore a modified scheme is also considered which is obtained imposing $\alpha=0$ in (17), this corresponds to use a linear scheme.

We performed an accuracy study on a sequence of unstructured triangular grids and as error estimation the normalized $L_2$ norm of the difference between the exact and the numerical solution has been used

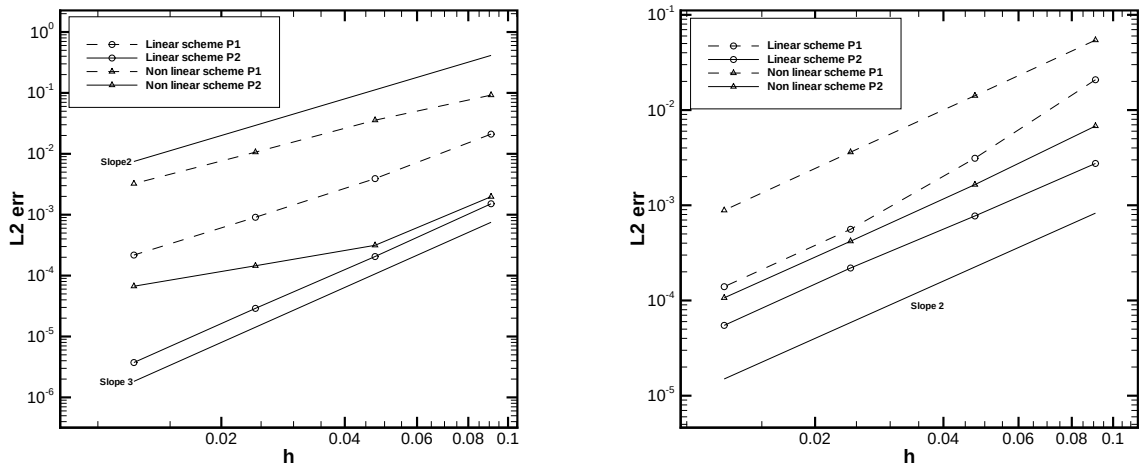
$$e_{L_2(\Omega)} = \sqrt{\frac{\int_\Omega (u-u_e)^2 \mathrm{d}\Omega}{\int_\Omega u_e^2\,\mathrm{d}\Omega}},$$

where $u_e$ is the exact solution. In all the simulations we initialize the solutions with a value zero everywhere except on inflow boundaries where the exact solution is imposed. From the numerical simulations it has been observed that the direct use of the scheme (14) generally does not guarantee that the iterative process reaches the steady state solution and that the scheme is not consistent, for this reason the results for this scheme are not shown.

In the first test case the viscosity coefficient is $\nu = 10^{-6}$ such that an advection dominated problem is considered. In Figure 2-(a) is reported the L2 norm of the error as function of the parameter $h = 1/\sqrt{N_{\mathrm{dof}}}$, with $N_{\mathrm{dof}}$ the total number of the DOF. It can be observed that the linear scheme is capable to preserve the right order of accuracy for both the second and the third order scheme. The non-linear scheme is second order accurate with the $P1$ elements although it has a bigger error constant than the linear scheme. In the case of the $P2$ elements the non-linear scheme loses the third order accuracy with the gird refinement, deteriorating to a first order only accurate scheme.

To investigate the behavior of the scheme when the advection and the diffusion are equally important we consider the advection-diffusion problem with $\nu = 0.01$. The accuracy study is reported in Figure 2-(b). It must be observed that the schemes have at the most second order accuracy with the $P2$ elements, and

the linear scheme has lower error constant than the non-linear one. The second order schemes has the right order of accuracy.




(a) (b)

Figure 2: Normalized L2 solution error for the advection-diffusion scalar problem with $\nu = 10^{-6}$ (left) and $\nu = 0.01$ (right).

In Figure 3 is reported the accuracy study of the same advection-diffusion problem with $\nu = 0.01$, as before, with the linear scheme and the special case in which the exact gradient of the solution is used instead of the reconstructed one. Since the exact solution of the problem is infinitely differentiable, the exact gradient is continuous. In this case the scheme has the right order of accuracy for both the second and third order scheme. Since the problem of loss of accuracy for the third order scheme seems related to the accuracy of the gradient reconstruction, we have performed an accuracy study of gradient reconstruction, on the same type of problem, for different techniques and two values of viscosity coefficient: $\nu = 10^{-6}$ and $\nu = 0.01$. The results are shown in Figure 4, it is worth noticing that the accuracy of gradient reconstruction does not depend on the value of the viscosity coefficient and that in the case of $P1$ element the L2 projection technique has the smallest error while in the $P2$ case the Green-Gauss formula and L2 projection has the same accuracy. The Least-Square technique is the less accurate. It is important to note that in the case $P1$ the accuracy of the gradient reconstruction is much closer to the behavior of second order accurate than to the first order accurate technique, while in the case $P2$ the accuracy of the gradient reconstruction is never more than two.

## 6.2 Navier-Stokes equations

We consider here the discretization of the laminar Navier-Stokes equations in order to verify the accuracy and the robustness of the proposed approach. The first test case that we consider is a laminar flow over a flat plate with a free stream Mach number, M = 0.3 a Reynolds number based on the free stream condition and the flat plate length Re = 5 000. The length of the plate is $L = 1$. The range of computational domain extends in the $x$-direction is $[-1, 1]$ with the leading edge of the flat plate at $x = 0$. The size of the computational domain in the $y$-direction is 1, which is 10 times the boundary layer thickness at $x = 1.0$. At the inlet the inflow boundary condition in imposed, at the top and the exit the static pressure is imposed. Along the plane $y = 0$, the symmetry boundary condition is imposed for $-1 \leq x \leq 0$ and the wall no-slip boundary condition is imposed for $0 \leq x \leq 1$. The coarsest mesh used is shown if Figure 5, if consists of 187 DOF. The medium and the fine grids are obtained successively dividing each triangle in four triangles.

We first consider the effect of the boundary conditions on the solution. In Figure 6 are compared the temperature and the axial velocity profiles at $x = 1$ for the second order schemes using the old boundary

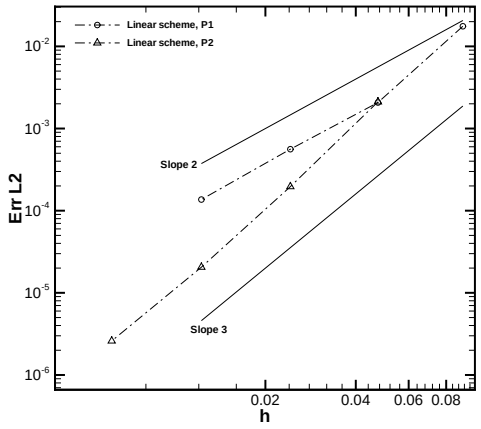



Figure 3: Normalized L2 error for the advection-diffusion scalar problem with $\nu = 0.01$ in which the exact gradient is used instead of the reconstructed one.

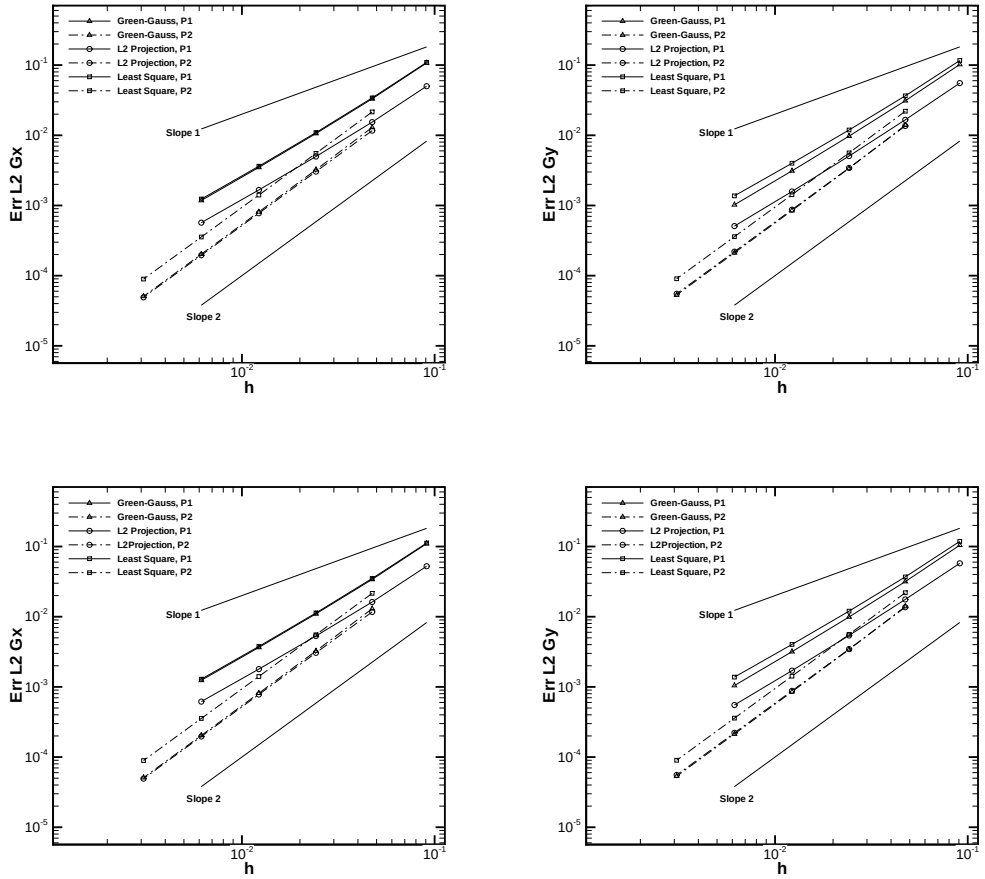



Figure 4: Normalized L2 error of gradient for different techniques of the gradient reconstruction. The advection-diffusion scalar problem is considered. Top row: $\nu = 10^{-6}$, bottom row: $\nu = 0.01$.

condition procedure and the new proposed approach on the medium grid. The results are compared against the exact solution of the thermal boundary layer [30]. Clearly the new approach is able to correctly impose the adiabatic condition, while the old approach completely fail to predict the temperature profile and also the heat flux is not zero on the wall. The velocity profile is instead in good agreement with the exact solution since the no-slip condition is strongly imposed in both the procedures. From the Figure 7-(a) can be also deduced that the new way to handle the adiabatic condition assure a fast convergence to the steady state solution. The third order scheme is much more sensible to the imposition of the boundary conditions as can be seen in Figure 7-(b), only the new way to impose the boundary condition on the wall makes the scheme converge.

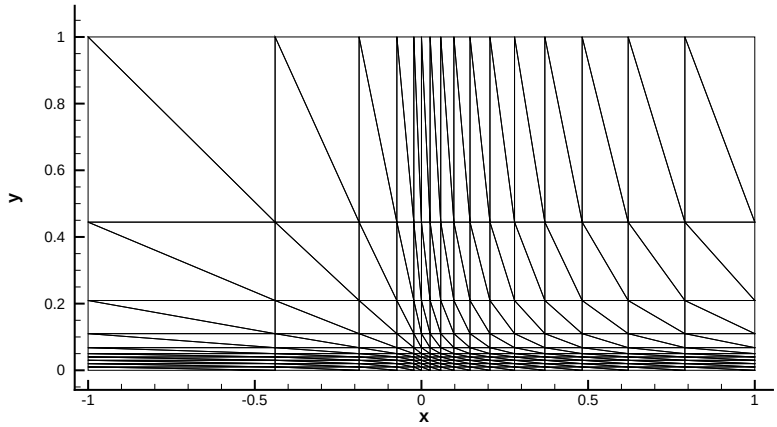



Figure 5: The coarsest mesh used for the flat plate boundary layer case.

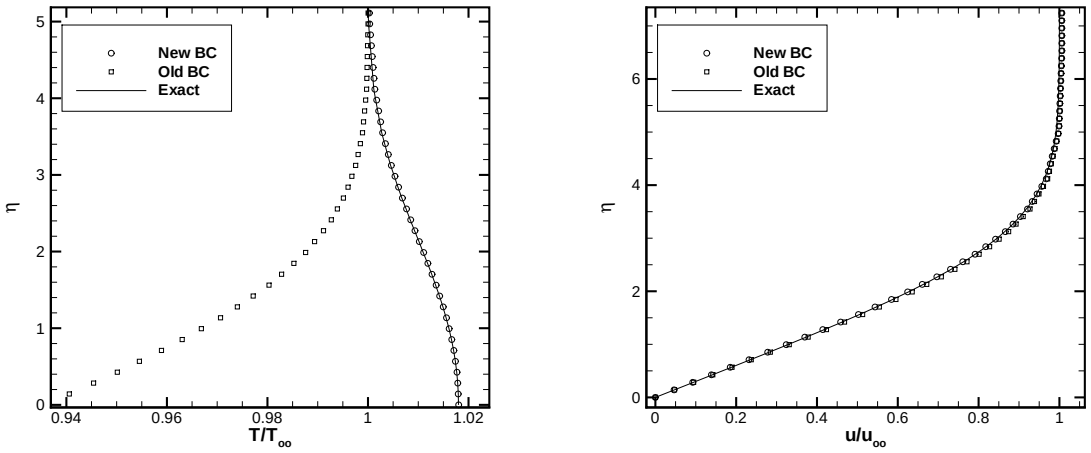
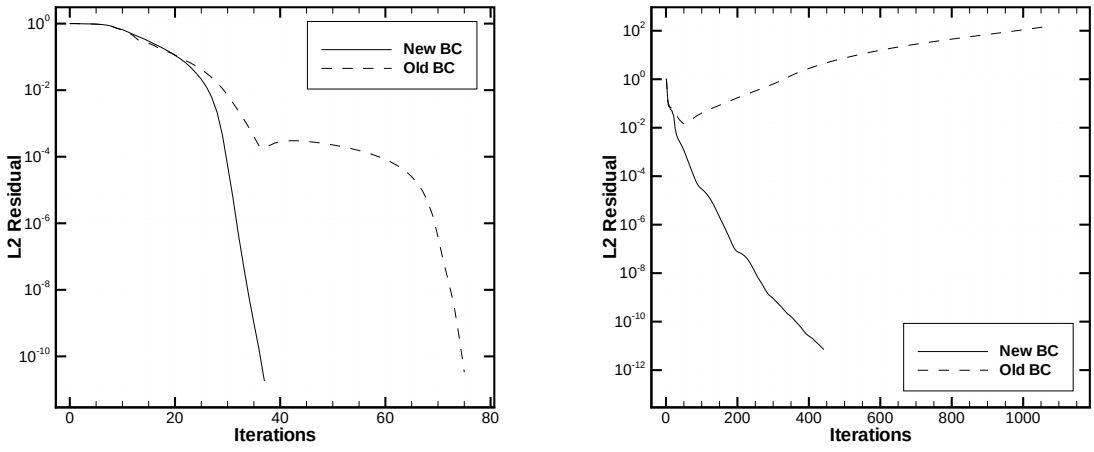



Figure 6: Temperature profile (left) and axial velocity profile (right) at $x = 1$ as function of the dimensionless wall distance ($\eta$) for the old and the improved way to impose the adiabatic boundary conditions. The numerical solutions (second order scheme) are compared against the theoretical solution.

In Figure 8 is shown the axial velocity profile at $x = 1$ for the second and third order schemes on three different grids. The numerical results are in good agreement with the analytical solution even on the coarsest grid. If Figure 9 is shown the skin friction coefficient along the flat plate for the second and third order




(a) (b)

Figure 7: L2 iterative residual of axial momentum equation for the old and the improved way to impose the adiabatic boundary conditions for the second (left) and third (right) order scheme.

schemes on three different grids. Note that the agreements becomes better refining the grid and increasing the order of the scheme.

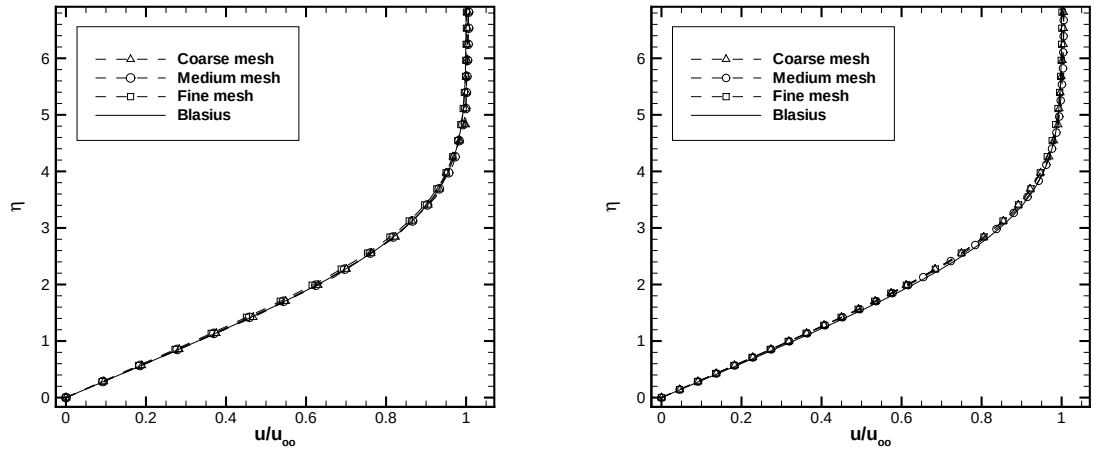



Figure 8: Axial velocity profile for the second (left) and third (right) order schemes at $x = 1$ as function of the dimensionless wall distance ($\eta$) on three different grids. The numerical solutions are compared against the theoretical solution.

The last test case we consider is the steady laminar three dimensional flow around a delta wing, the same test case has been considered in [19]. The delta wing has sharp edges, the geometry of the wing is shown is Figure 10, together with the initial mesh which consist of 21 035 nodes (110 629 elements). The free-stream Mach number is 0.3, the Reynolds number is 4 000 and the angle of attack is 12.5°. As the flow reaches the wing it rolls up creating a main vortex together with a secondary small vortex over the wing. The resulting vortex system remains over long distance behind the wing. Considering the complex flow patter, the numerical solution of the problem has been combined with a sequence of mesh adaptations. The grids are adapted using Mmg3d library [14] with the vorticity magnitude as criterion for the adaptation.

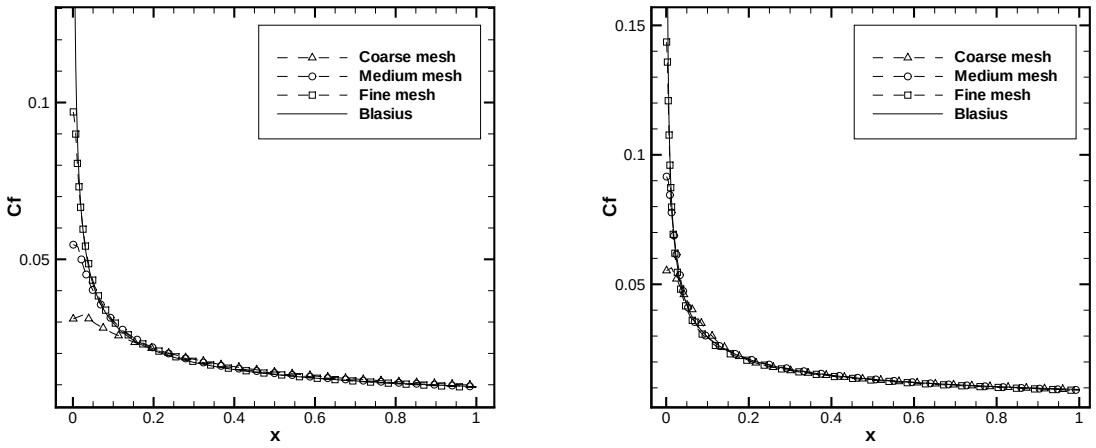



Figure 9: Skin friction coefficient along the flat plate fort he second (left) and third (right) order schemes on three different grids. The numerical solutions are compared against the theoretical solution.

In Figure 11 are compared the contours of Mach number and of the vorticity magnitude together with the mesh, at the end of the wing, for the initial grid and the adapted grid after one cycle. In Figure 12 are compared the contours of the Mach number and of the vorticity magnitude together with the mesh, at the end of the wing and on the wake, for the adapted grid after one cycle and two cycles of adaptation. The final grid consist of 102 413 nodes (594 595 elements). The grid adaptation is able to refine the mesh in the zone that contains the main vortex, resulting in a better resolution of the flow patter behind the wing. This can be observed in Figure 13, where are shown the slices of the Mach number and of the vorticity magnitude for the original grid and the adapted grids, at different positions along the wing and the wake. It is worth noticing how adapting mesh allows to capture better the vortex structure behind the wing, as it can be also seen in Figure 14 where the Mach iso-surface for the original and final adapted grid is shown.

On the final grid we performed also a third order simulation, the results of the second and third order scheme are compared in Figure 6.2. Finally, in Table 6.2 are reported the values of the aerodynamic coefficients for the different simulations. The values obtained on the final grid are not far from the reference values, the difference is probably due to the fact that during the adaptation process the surface discretization of the wing has not be modified respect to the original coarse grid, since the library used is able to adapt only on the volume. Nevertheless, it is worth noticing that the value of the drag coefficient is very close to the reference value, because in this case the main contribution to the drag is due to the flow separation over and behind the wing, and this feature is well captured by the grid adaptation.

| | $C_L$ | $C_D$ |
|---|---|---|
| Original Grid (Order 2) | 0.32737699 | 0.144857313 |
| Adapted grid 1 (Order 2) | 0.33456865 | 0.145790923 |
| Adapted grid 2 (Order 2) | 0.33923147 | 0.147307554 |
| Adapted grid 2 (Order 3) | 0.33667129 | 0.161839115 |
| Reference [19] | 0.347 | 0.165 |

Table 1: Values of the lift and drag coefficients for the different simulations of the delta wing test case.

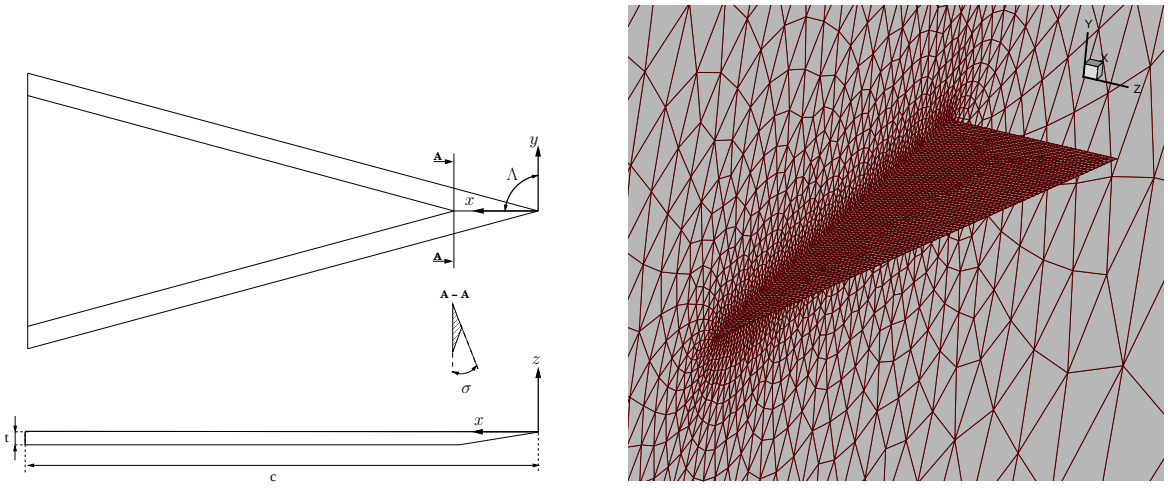



Figure 10: Geometry of the delta wing. The following parameter has been used: $\Lambda = 75°$, $\sigma = 60°$ and $t/c = 0.024$. The initial coarse mesh is also shown.

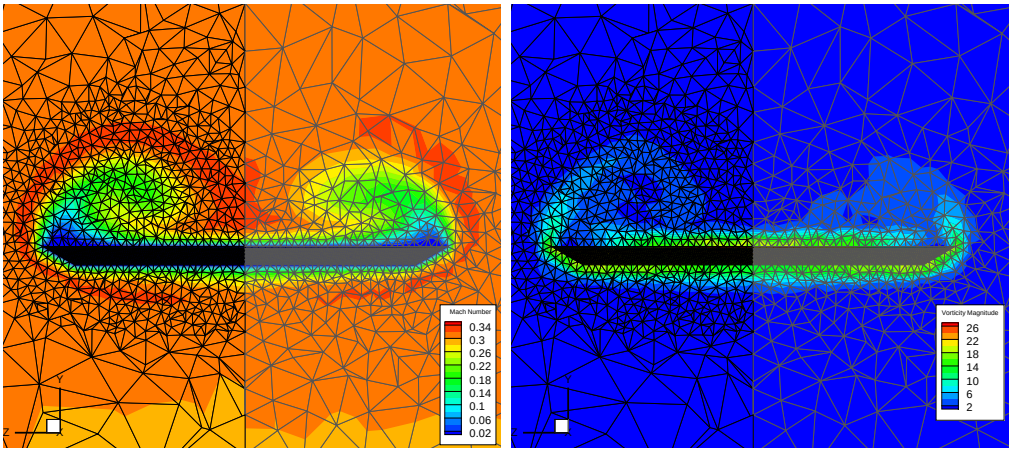



Figure 11: Contours of the Mach number (left) and of the vorticity magnitude (right) at the end of the wing. In each figure are shown the solution and the grid before and after the first grid adaptation. Second order scheme.

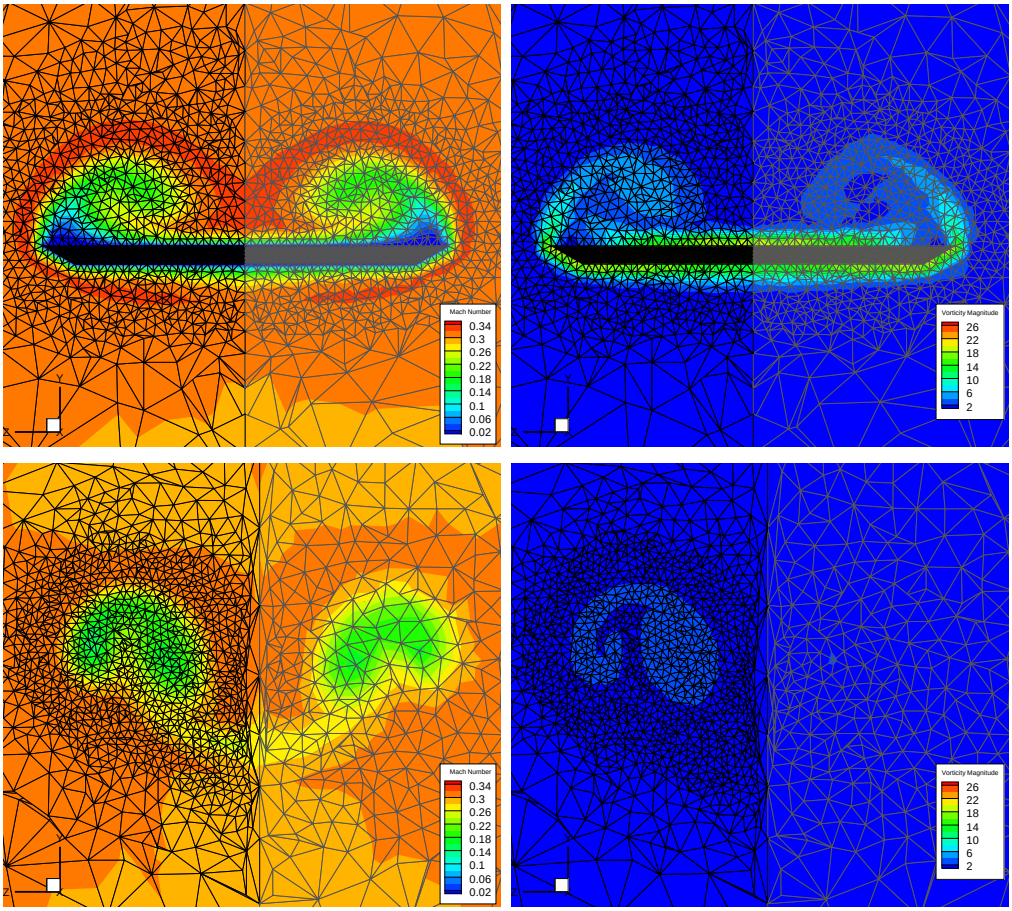



Figure 12: Contours of the Mach number and of the vorticity magnitude at the end of the wing (first row) and behind the wing (second row). In each figure are shown the solution and the grid before and after the second grid adaptation. Second order scheme.

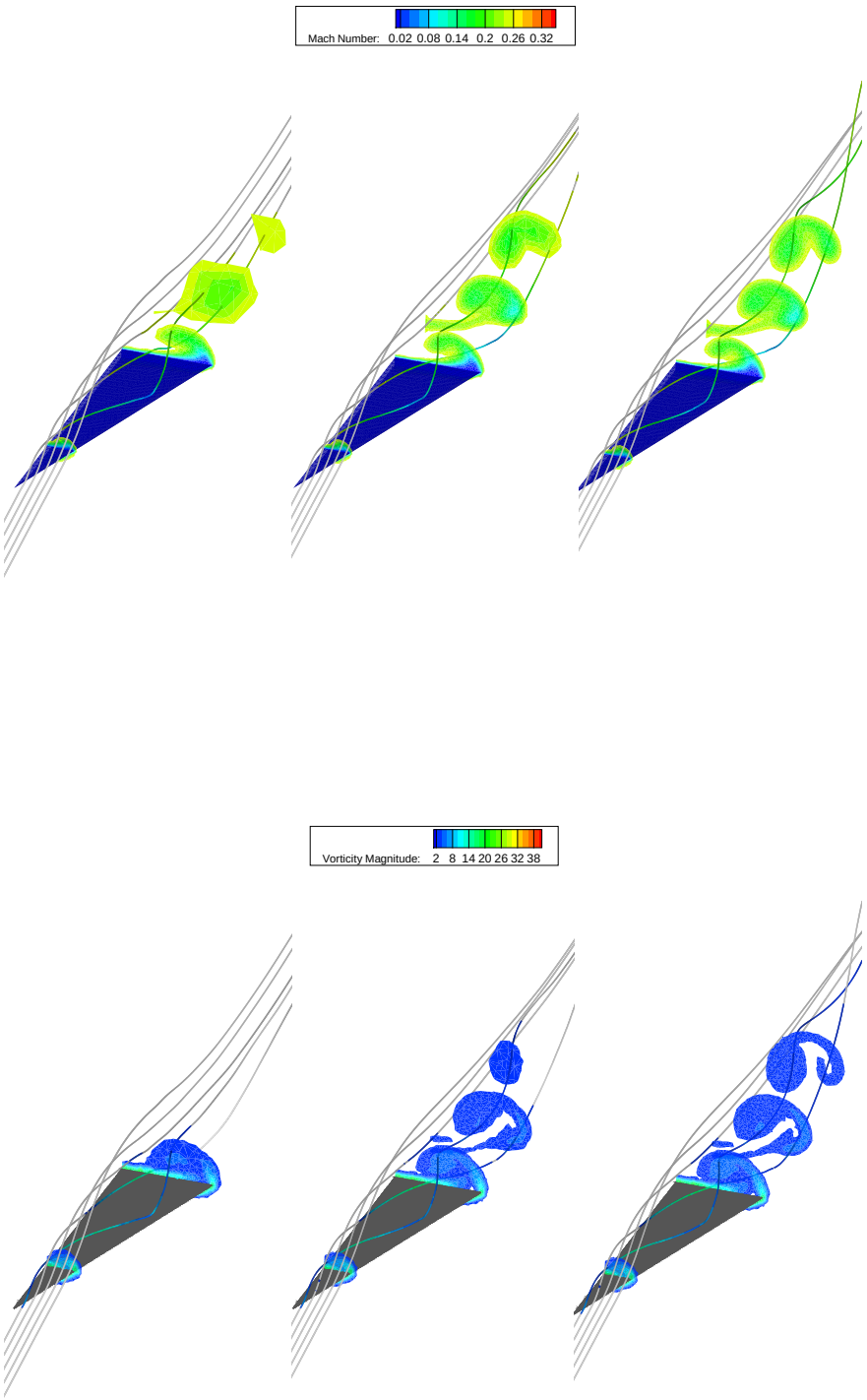



Figure 13: Slices of the Mach number and the vorticity magnitude at different positions along the wing and the wake. Second order scheme.

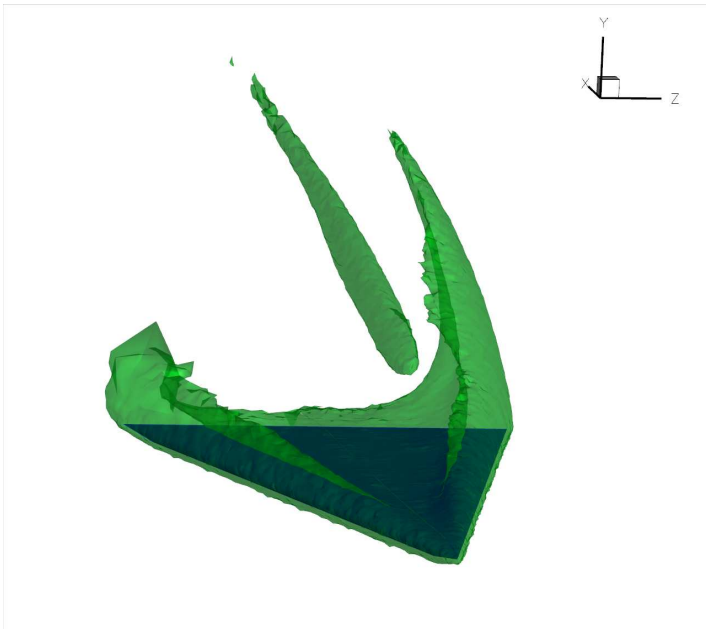



Figure 14: Mach iso-surface; on the left part of the wing solution on the original grid on the right part of the wing solution on the final adapted grid. Second order scheme.

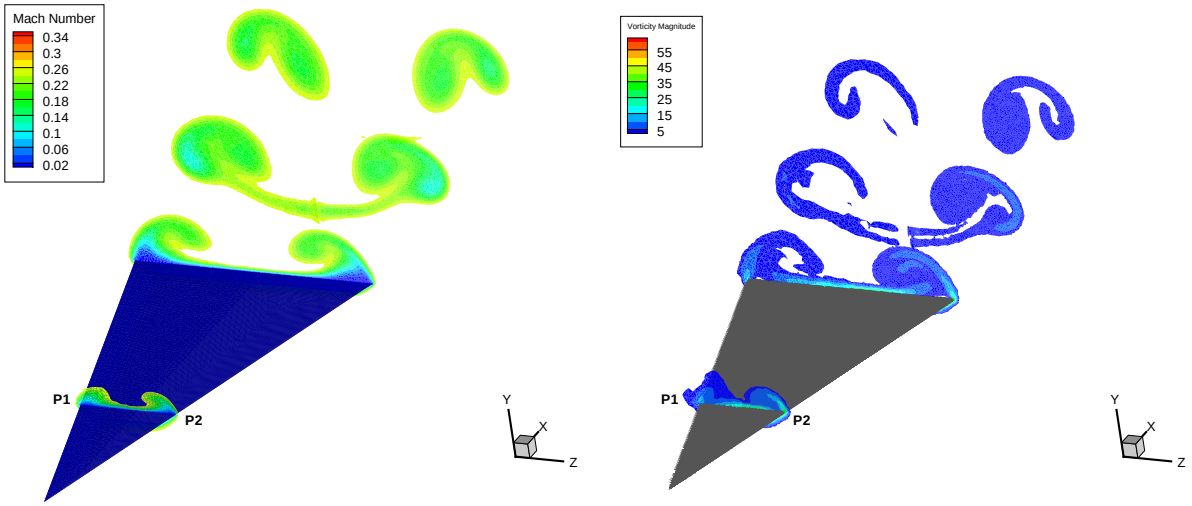



Figure 15: Comparison of the Mach number (left) and vorticity magnitude (right) for the second and third order scheme.

# 7 Conclusion

A high order accurate Residual Distribution scheme for the solution of the compressible Navier-Stokes equations has been presented. The scheme avoid the use of the combined discretization with the RD scheme for the advection and the Galerkin scheme for the diffusion. The contribution of the advection and of the diffusion are handled with the same scheme. It has been also proposed a Jacobian-free technique to accelerate the convergence of the implicit iterative scheme to the steady state solution. The proposed scheme, in combination with an improvement of the adiabatic no-slip boundary conditions, has shown to perform the standard test cases used for laminar compressible flows as well as complex three dimensional aerodynamic flows.

# References


[1] R. Abgrall. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *Journal of Computational Physics*, 214:773–808, 2006.

[2] R. Abgrall. A residual distribution method using discontinuous elements for the computation of possibly non smooth flows. *Advances in Applied Mathematics and Mechanics*, 2:32–44, 2010.

[3] R. Abgrall, G. Baurin, A. Krust, D. De Santis, and M. Ricchiuto. Numerical approximation of parabolic problems by means of residual distribution schemes. *International Journal for Numerical Methods in Fluids (accepted for publication)*, 2012.

[4] R. Abgrall, A. Larat, and M. Ricchiuto. Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes. *Journal of Computational Physics*, 230:4103–4136, 2011.

[5] R. Abgrall and M Mezine. Construction of second-order accurate monotone and stable residual distribution schemes for unsteady problem. *Journal of Computational Physics*, 195:474–507, 2004.

[6] R. Abgrall and P. L. Roe. High-order fluctuation schemes on triangular meshes. *Journal of Scientific Computing*, 19:3–36, 2003.

[7] R. Abgrall and D. De Santis. High order residual distribution scheme for Navier-Stokes equations. In *20th AIAA Computational Fluid Dynamics Conference*. AIAA 2011-3231, 2011.

[8] R. Abgrall and C. W. Shu. Development of residual distribution schemes for the discontinuous Galerkin method: the scalar case with linear elements. *Communications in Computational Physics*, 5:376–390, 2009.

[9] Francesco Bassi, Andrea Crivellini, Stefano Rebay, and Marco Savini. Discontinuous Galerkin solution of the reynolds-averaged Navier–Stokes and $k-\omega$ turbulence model equations. *Computers & Fluids*, 34:507 – 540, 2005.

[10] A.N. Brooks and T. J. R Hughes. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32:199–259, 1982.

[11] P.N. Brown and Y. Saad. Hybrid krylov methods for nonlinear systems of equations. *SIAM Journal on Scientific and Statistical Computing*, 11:450–481, 1990.

[12] D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind scheme for Navier–Stokes simulations. *Theoretical and Computational Fluid Dynamics*, 15:373–401, 2002.

[13] B. Cockburn, G.E. Karniadakis, and C.W. Shu. *Discontinuous Galerkin methods: theory, computation and application*. Lecture notes in computational science and engineering. Springer, Berlin, 2000.

[14] Cécile Dobrzynski. Mmg3d: User Guide. Technical Report 422, Inria, March 2012.

[15] S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact newton method. *SIAM Journal on Scientific and Statistical Computing*, 17:16–32, 1996.

[16] M.E. Hubbard. A framework for discontinuous fluctuation distribution. *International Journal for Numerical Methods in Fluid*, 56:1305–1311, 2008.

[17] A. Jameson and S. Yoon. Lower-upper implicit schemes with multiple grids for the Euler equation. *AIAA Journal*, 25:929–935, 1987.

[18] D. A. Knoll and D. Keyes. Jacobian-free newton method: a survey of approaches and application. *Journal of Computational Physics*, 193:357–397, 2004.

[19] T. Leicht and R. Hartmann. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations. *Journal of Computational Physics*, 229(19):7344 – 7360, 2010.

[20] H. Luo, J.B. Baum, and R. Lohener. A fast, matrix-free implicit method for compressible flows on unstructured grids. *Journal of Computational Physics*, 146:664–690, 1998.

[21] N.P.C Marques and J.C.F Pereira. Comparison of matrix-free acceleration techniques in compressible Navier–Stokes calculations. *International Journal for Numerical Methods in Engineering*, 61:455–474, 2007.

[22] H. Nishikawa. A first-order system approach for diffusion equation. I: Second-order residual-distribution scheme. *Journal of Computational Physics*, 227:315–352, 2007.

[23] H. Nishikawa. Beyond interface gradient: A general principle for constructing diffusion scheme. In *40th Fluid Dynamics Conference and Exhibit*. AIAA Paper 2010-5093, 2010.

[24] H. Nishikawa. A first-order system approach for diffusion equation. II: Unification of advection and diffusion. *Journal of Computational Physics*, 229:3889–4016, 2010.

[25] H. Nishikawa. Robust and accurate viscous discretization via upwind scheme I: Basic principle. *Computers & Fluids*, 49:62–86, 2011.

[26] H. Nishikawa and P. L. Roe. On high-order fluctuation-splitting schemes for Navier–Stokes equations. In *Computational Fluid Dynamics 2004: Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD, Toronto, 12-16 July 2004*. Springer 2006, 2004.

[27] H. Paillère, J. Boxho, G. Degrez, and H. Deconinck. Multidimensional upwind residual distribution schemes for the convection-diffusion equation. *International Journal for Numerical Methods in Engineering*, 23:923–936, 1996.

[28] M. Pernice and H.F. Walker. Nitsol: A newton iterative solver for nonlinear systems. *SIAM Journal on Scientific and Statistical Computing*, 19:302–318, 1998.

[29] T. Saad and M.H. Schultz. Gmres: a generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869, 1986.

[30] Hermann Schlichting and Klaus Gersten. *Boundary-layer theory*. Springer, 2000.

[31] K. R. Jackson T. F. Chan. Nonlinearly preconditioned krylov subspace methods for discrete newton algorithms. *SIAM Journal on Scientific and Statistical Computing*, 5:533–4542, 1984.

[32] M.D. Tidiriri. Preconditioning techniques for the newton-krylov solution of compressible flows. *Journal of Computational Physics*, 132:51–61, 1997.

[33] N. Villedieu, T. Quintino, and H. Deconinck M. Ricchiuto. Third order residual distribution schemes for the Navier–Stokes equations. *Journal of Computational Physics*, 230:4301–4315, 2011.