

# A NURBS-based Discontinuous Galerkin method for conservation laws with high-order moving meshes

Stefano Pezzano, Régis Duvigneau

► **To cite this version:**

Stefano Pezzano, Régis Duvigneau. A NURBS-based Discontinuous Galerkin method for conservation laws with high-order moving meshes. *Journal of Computational Physics*, Elsevier, 2021, 434 (1). hal-02887312v2

**HAL Id: hal-02887312**

**<https://hal.inria.fr/hal-02887312v2>**

Submitted on 6 Jan 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A NURBS-based Discontinuous Galerkin method for conservation laws with high-order moving meshes

Stefano Pezzano, Régis Duvigneau

Université Côte d'Azur, INRIA, CNRS, LJAD  
INRIA Sophia-Antipolis, 2004 route des Lucioles - B.P. 93  
06902 Sophia-Antipolis, France

## Abstract

The objective of the present work is to develop a new numerical framework for simulations involving deformable domains, in the specific context of high-order meshes consistent with Computer-Aided Design (CAD) representations. Thus, the proposed approach combines ideas from isogeometric analysis, able to handle exactly CAD-based geometries, and Discontinuous Galerkin (DG) methods with an Arbitrary Lagrangian-Eulerian (ALE) formulation, able to solve complex problems with moving grids. The resulting approach is a DG method based on rational Bézier elements, that can be easily constructed from Non-Uniform Rational B-Splines (NURBS), formulated in a general ALE setting. We focus here on applications in two-dimensional compressible flows, but the method could be applied to other models as well. Two verification exercises are conducted, to assess rigorously the properties of the method and the convergence rates for representations up to sixth order. Finally, three problems are analysed in depth, involving compressible Euler and Navier-Stokes equations, for an oscillating cylinder and a pitching airfoil. In particular, the convergence of flow characteristics is investigated, as well as the impact of using curved boundaries in the context of deformable domains.

## 1 Introduction

In the last 20 years, Computational Fluid Dynamics (CFD) has become a standard tool for analysis and design of fluid systems in industry. In this context, the issue of a better integration of CFD solvers in complex multidisciplinary analysis chains has emerged. In particular, the lack of integration between Computer Aided Design (CAD) and CFD software regarding the treatment of the geometry has been reported. This difficulty is mainly due to the fact that, in most industrial cases, the grid generation process is not fully automatized yet and requires several geometrical transformations that are still carried out by hand. In order to facilitate the integration of design and simulation, Hughes et al. (1) introduced the concept of Iso-Geometric Analysis (IGA), whose formulation relies on Non-Uniform Rational B-Splines (NURBS), the standard mathematical representation of modern CAD (2). Essentially, IGA consists in a finite element method in which both the geometry and the solution are described by NURBS functions. Even though IGA aims at a seamless integration of CAD and finite-element methods, it is still limited by the difficulty of generating NURBS-based grids for complex topologies. However, new approaches are emerging to construct NURBS-based parameterizations of computational domains, inspired by both structured (3) and unstructured (4; 5; 6) mesh generation techniques.

At the same time, the CFD community has developed a growing interest in high-order schemes for conservation laws (7). In particular, promising results have been obtained with Discontinuous Galerkin (DG) methods in a number of applications (8; 9; 10). However, as shown by several authors, the use of piecewise-linear geometry descriptions in such very high-order CFD solvers may lead to non-physical phenomena in numerical solutions (8; 11; 12) and generally limits the accuracy of such schemes to second order. This observation justified the development of a new generation of high-order solvers, capable of handling CAD geometries natively. One can refer for instance to the NURBS-Enhanced Finite Element Method (NEFEM) (13) and its Discontinuous Galerkin counterpart (11), in which the NURBS representations are only employed for the geometry of the boundaries. Similarly, an extension of the residual distribution scheme for NURBS geometries has been proposed (14). More recently, two DG schemes inspired by IGA have been developed: the Blended-Isogeometric Discontinuous Galerkin (BIDG) method (15) which relies on nodal elements whose geometry is defined by NURBS in the whole computational domain, and its fully isogeometric counterpart which uses NURBS for both the geometry and the solution (16). All these approaches have demonstrated the interest of using high-order boundary representations in terms of accuracy for some selected test-cases governed by compressible or incompressible flow models. Besides, the DG framework may alleviate the mentioned difficulties of generating NURBS-based grids, thanks to the higher flexibility of the DG method in handling non-conformities and local refinement (17).

Complex flows, however, often involve time dependent geometries or moving interfaces. As illustrated in (18), the use of high-order curvilinear grids with ALE formulation is an advantage in the context of multi-material flows. Bazilevs et al. (19) demonstrated the benefits of IGA for Lagrangian hydrodynamics, and, in the same field, a Finite Volume method based on conical cells has been presented in (20). More generally, improving the description of the geometry is certainly beneficial in applications with deformable domains, in particular because the displacement of the boundary may amplify the spurious phenomena generated by a low-order geometry representation. Therefore, simulating flows while accounting exactly for CAD geometries should be an advantage over traditional approaches where the moving boundaries are piecewise linear. This has been illustrated for fluid-structure interaction problems in the context of classic IGA (21; 22). However, several issues arise when one attempts to combine Arbitrary Lagrangian-Eulerian (ALE) formulations and NURBS representations, such as the development of a robust technique to deform high-order grids or the preservation of constant solutions on moving meshes. As a consequence, the present study aims at proposing a new framework for simulating compressible flows with arbitrary displacement of boundaries defined by NURBS. The proposed approach relies on a DG formulation based on rational Bézier elements (16), that can be easily constructed from a set of NURBS boundaries using local refinement (17), preserving the CAD geometry exactly.

ALE algorithms can be classified into two categories: direct methods (23; 9), in which the equations in ALE form are solved, and indirect methods (18; 24), which are characterized by a pure Lagrangian computation, followed by a rezoning and a remapping phase. The present work is focused on direct ALE approaches. In particular, several ALE formulations have been proposed in the context of DG methods. A review of the possible direct ALE techniques is therefore carried out in section 2 and two categories of approaches are identified. In the first family of methods, the conservation laws are effectively solved in the moving domain (9; 25), whereas, in the second class of schemes, the equations are solved in a fixed reference domain thanks to the introduction of a map (26; 27). The main characteristics

of the various approaches are discussed and, in particular, the respect or the violation of the Discrete Geometric Conservation Law (DGCL) is examined in each case. For the sake of simplicity, the proposed analysis is limited to method of lines DG discretizations. It is however worth mentioning that interesting results have also been obtained with space-time ALE-DG schemes, either in fully implicit form (28) or, more recently, in the ADER framework (Arbitrary high-order schemes using DERivatives) (23; 29).

The most suitable ALE formulation for NURBS representation is selected, and its extension to rational Bézier elements is detailed in section 3. Moreover, we explain how the regularity and hierarchy properties of NURBS can be exploited to define a very smooth mesh deformation algorithm, even in presence of large boundary displacements. In order to focus on the validation of the ALE formulation, we consider test-cases where the mesh movement is known a priori. Problems involving physics-based couplings, such as fluid-structure interaction or multi-material flows, are out of the scope of the present manuscript. In section 4, a rigorous verification of the methods is presented using two problems with analytic solutions governed by linear advection and compressible Euler equations. Then, in section 5, three more complex test-cases are considered to assess the capabilities of the proposed methodologies. The compressible Navier-Stokes equations are first solved for an oscillating cylinder, for different representations ranging from degree 3 to degree 6. A systematic grid refinement study is achieved to assess the accuracy of the computations and compare the results obtained with reference data from the literature. As a second problem, we consider the compressible flow around a pitching airfoil, governed by Euler and compressible Navier-Stokes equations. Here, we focus on quantifying the impact of using curved grids, in comparison with classical piecewise-linear meshes, in order to highlight the gain obtained by improving the geometry description in the context of moving bodies. The last test case investigates the pitching airfoil flow in transonic conditions, with the goal of assessing the robustness of the developed scheme in presence of shocks. Finally, the main points learned in this study are summarized in the conclusion.

## 2 ALE formulations for Discontinuous Galerkin

We start by considering, for the sake of generality, a system of conservation laws of the following form:

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathbf{F} = 0, \quad (1)$$

with  $\mathbf{W}$  being the vector of conservative variables and  $\mathbf{F}$  the physical flux. The Arbitrary Lagrangian-Eulerian (ALE) approach represents a generalization of the Eulerian and Lagrangian viewpoints. Using the ALE framework, conservation laws can be solved in a referential domain, which is moving with an arbitrary velocity  $\mathbf{V}_g$ . As explained by Donea et al. (30) and Venkatasubban (31), it is possible to write the conservation law (1) in its respective ALE form:

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathbf{F} - \mathbf{V}_g \cdot \nabla \mathbf{W} = 0, \quad (2)$$

where  $\partial/\partial t$  is the time derivative in the referential domain.

In a first family of ALE-DG approaches, the weak form of the conservation law is solved in the referential domain. In order to discretize eq. (2) using a DG scheme, the computational domain  $\Omega$  is discretized using a set of elements  $\Omega_j$ ,  $j = \{1, 2, \dots, N_{el}\}$ . The numerical solution

$\mathbf{w}_h$  is then expressed in each element of index  $j$  as:

$$\mathbf{w}_h^j = \sum_{i=1}^{N_p} \varphi_i^j(\mathbf{x}) \mathbf{w}_i^j, \quad (3)$$

with  $\varphi_i^j$  being a generic set of basis functions and  $\mathbf{w}_i^j$  the  $N_p$  degrees of freedom, defined on the  $j$ -th element. The weak formulation is obtained multiplying eq. (2) by a basis function  $\varphi_k$  and integrating over the elemental domain  $\Omega_j$ :

$$\int_{\Omega_j} \varphi_k \frac{\partial \mathbf{w}_h}{\partial t} d\Omega + \int_{\Omega_j} \varphi_k \nabla \cdot \mathbf{F} d\Omega - \int_{\Omega_j} \varphi_k \mathbf{V}_g \cdot \nabla \mathbf{w}_h d\Omega = 0. \quad (4)$$

Note that the index  $j$  on the basis functions and the discrete solution has been dropped to ease the notation. After integration by parts, one obtains:

$$\begin{aligned} \int_{\Omega_j} \varphi_k \frac{\partial \mathbf{w}_h}{\partial t} d\Omega + \int_{\Omega_j} \nabla \varphi_k \cdot \mathbf{F} d\Omega - \int_{\Omega_j} \nabla \cdot (\varphi_k \mathbf{V}_g) \mathbf{w}_h d\Omega \\ + \oint_{\partial\Omega_j} \varphi_k (\mathbf{F}^* - \mathbf{V}_g \mathbf{w}^*) \cdot \mathbf{n} d\Gamma = 0, \end{aligned} \quad (5)$$

where  $\mathbf{F}^*(\mathbf{w}_h^+, \mathbf{w}_h^-, \mathbf{n})$  is a consistent Riemann solver and  $\mathbf{w}^*(\mathbf{w}_h^+, \mathbf{w}_h^-)$  is the solution of the associated Riemann problem at the element interface. Manipulating equation (5) it is possible to derive several ALE-DG schemes, that are equivalent from a continuous point of view, but not at a discrete level. Indeed, constant solutions may not be preserved by the discrete scheme due to the mesh movement, as discussed in (9). In order to ensure an exact preservation of constant solutions, an ALE method has to satisfy the so-called Discrete Geometric Conservation Law (DGCL), which is specific to each numerical scheme, as explained by Guillard et al. (32).

Applying again integration by parts to the second and third terms of eq. (5), the ALE-DG scheme of Lomtev et al. (9) is found:

$$\begin{aligned} \int_{\Omega_j} \varphi_k \frac{\partial \mathbf{w}_h}{\partial t} d\Omega + \int_{\Omega_j} \varphi_k (\nabla \cdot \mathbf{F} - \mathbf{V}_g \cdot \nabla \mathbf{w}_h) d\Omega \\ + \oint_{\partial\Omega_j} \varphi_k [\mathbf{F}^* - \mathbf{F} - \mathbf{V}_g (\mathbf{w}^* - \mathbf{w}_h)] \cdot \mathbf{n} d\Gamma = 0. \end{aligned} \quad (6)$$

This formulation can be interpreted as the ALE extension of the strong form DG, presented in (33). Provided that the numerical fluxes  $\mathbf{F}^*$  and  $\mathbf{w}^*$  are consistent, eq. (6) always preserves constant solutions exactly. However, the implementation is substantially different with respect to the more commonly used DG method in weak form. In order to find an alternative formulation, we rewrite eq. (5) as:

$$\begin{aligned} \int_{\Omega_j} \varphi_k \frac{\partial \mathbf{w}_h}{\partial t} d\Omega + \int_{\Omega_j} \varphi_k \mathbf{w}_h \nabla \cdot \mathbf{V}_g d\Omega - \int_{\Omega_j} \nabla \varphi_k \cdot (\mathbf{F} - \mathbf{V}_g \mathbf{w}_h) d\Omega \\ + \oint_{\partial\Omega_j} \varphi_k (\mathbf{F}^* - \mathbf{V}_g \mathbf{w}^*) \cdot \mathbf{n} d\Gamma = 0. \end{aligned} \quad (7)$$

Introducing  $\mathbf{F}_{ale}^* = \mathbf{F}^* - \mathbf{V}_g \mathbf{w}^*$  and applying the Reynolds transport theorem for moving control volumes, the following ALE-DG scheme is found:

$$\frac{d}{dt} \int_{\Omega_j} \varphi_k \mathbf{w}_h \, d\Omega - \int_{\Omega_j} \nabla \varphi_k \cdot (\mathbf{F} - \mathbf{V}_g \mathbf{w}_h) \, d\Omega + \oint_{\partial\Omega_j} \varphi_k \mathbf{F}_{ale}^* \, d\Gamma = 0. \quad (8)$$

From a practical point of view,  $\mathbf{F}_{ale}^*(w_h^+, w_h^-, \mathbf{V}_g, \mathbf{n})$  is computed with a numerical flux function that satisfies the modified consistency property:

$$\mathbf{F}_{ale}^*(\mathbf{w}_0, \mathbf{w}_0, \mathbf{V}_g, \mathbf{n}) = \mathbf{F}(\mathbf{w}_0) \cdot \mathbf{n} - (\mathbf{V}_g \cdot \mathbf{n}) \mathbf{w}_0. \quad (9)$$

The formulation (8) was first introduced by Nguyen (25) and represents an elegant extension to moving domains of the classic DG method in weak form. One can show that this scheme exactly preserves constant solutions if the following identity is verified at the discrete level:

$$\oint_{\partial\Omega_j} \varphi_k \mathbf{V}_g \cdot \mathbf{n} \, d\Gamma - \int_{\Omega_j} \nabla \varphi_k \cdot \mathbf{V}_g \, d\Omega = 0, \quad (10)$$

which means that the numerical quadrature has to be sufficiently accurate to exactly compute the terms of the identity. As a consequence, constant solutions may not be exactly preserved for elements of general shape, such as curvilinear elements, and for arbitrary grid velocity fields.

In a second family of ALE approaches the conservation law (1) is mapped in a fixed reference domain by means of a function  $\mathcal{G}$  that transforms a point  $\hat{\mathbf{x}}$  in the reference domain into the corresponding point  $\mathbf{x}$  in the physical space:  $\mathbf{x}(t) = \mathcal{G}(\hat{\mathbf{x}}, t)$ . It has been shown in (34; 26) that the equivalent of equations (1) in the reference domain can be written as:

$$\frac{\partial \widehat{\mathbf{W}}}{\partial t} + \widehat{\nabla} \cdot \widehat{\mathbf{F}} = 0, \quad (11)$$

where  $\widehat{\nabla}$  is the vector of partial derivatives with respect to the reference coordinate frame, and the transformed conservative variables  $\widehat{\mathbf{W}}$  and corresponding flux vector  $\widehat{\mathbf{F}}$  are defined as:

$$\widehat{\mathbf{W}} = |J_G| \mathbf{W}, \quad \widehat{\mathbf{F}} = |J_G| J_G^{-1} (\mathbf{F} - \mathbf{V}_M \mathbf{W}), \quad (12)$$

where  $J_G$  is the Jacobian matrix of the map and  $\mathbf{V}_M$  is the map velocity:

$$\mathbf{V}_M = \frac{\partial \mathcal{G}}{\partial t}. \quad (13)$$

The mapped ALE technique consists in solving the transformed system of conservation laws (11) in the reference domain. The mapped ALE-DG method has been introduced by Persson et al. (26) in the context of compressible fluid mechanics. The reference domain is discretized and the basis functions are defined in the reference space on each element  $\widehat{\Omega}_j$ . Then, a DG discretization is applied to (11), obtaining:

$$\begin{aligned} \frac{d}{dt} \int_{\widehat{\Omega}_j} \varphi_k \mathbf{w}_h |J_G| \, d\widehat{\Omega} - \int_{\widehat{\Omega}_j} \widehat{\nabla} \varphi_k \cdot J_G^{-1} (\mathbf{F} - \mathbf{V}_M \mathbf{w}_h) |J_G| \, d\widehat{\Omega} \\ + \oint_{\partial\widehat{\Omega}_j} \varphi_k J_G^{-1} \mathbf{F}_{ale}^* |J_G| \, d\widehat{\Gamma} = 0, \end{aligned} \quad (14)$$

where the map  $\mathcal{G}$  is assumed to be  $\mathcal{C}^1$ . It is worthwhile noting that, even though the mapped ALE approach is significantly different, many similarities can be observed comparing equations (14) and (8). Due to the presence of the metric terms of the map within the discrete equations, this method violates the DGCL in general. However, as explained in (26), the preservation of constant solutions can be enforced by slightly modifying eq. (11) and adding a scalar conservation equation that has to be solved with the same DG scheme, see (26) for more details. Since arbitrary maps can be employed, the mapped ALE approach yields an extremely flexible method and allows high-order domain deformations. On the other hand, computing the map function may not be straightforward when the movement law is not known *a priori*. After this synthesis on ALE methods for DG, we describe in the following section how to adapt one of the analyzed schemes to CAD-consistent representations.

### 3 NURBS-based ALE-DG

#### 3.1 Basis functions

The construction of a CAD-consistent scheme necessarily relies on bases used in CAD. The first approach to interactive geometric design consisted in the use of Bézier surfaces, defined as (2):

$$\mathbf{S}(\xi, \eta) = \sum_{i_1=1}^{p+1} \sum_{i_2=1}^{p+1} B_{i_1}^p(\xi) B_{i_2}^p(\eta) \mathbf{x}_{i_1 i_2}, \quad (15)$$

where  $\mathbf{x}_{i_1 i_2}$  are the control points, and  $B_i^p$  are the Bernstein polynomials of degree  $p$ :

$$B_i^p(\varphi) = \binom{p}{i-1} \varphi^{i-1} (1-\varphi)^{p-i+1}, \quad (16)$$

with  $\varphi \in [0, 1]$ . Polynomial curves are not able to exactly represent conic sections, therefore, in order to overcome this limitation, rational Bernstein functions were introduced (35):

$$R_{i_1 i_2}^p(\xi, \eta) = \frac{B_{i_1}^p(\xi) B_{i_2}^p(\eta) \omega_{i_1 i_2}}{\sum_{j_1=1}^{p+1} \sum_{j_2=1}^{p+1} B_{j_1}^p(\xi) B_{j_2}^p(\eta) \omega_{j_1 j_2}}, \quad (17)$$

with the coefficients  $\omega_{i_1 i_2}$  being positive real numbers called weights. Since Bernstein polynomials are a partition of unity, when the weights are uniform,  $R_i^p(\xi)$  coincides with  $B_i^p(\xi)$ . Rational Bézier surfaces are defined as:

$$\mathbf{S}(\xi, \eta) = \sum_{i_1=1}^{p+1} \sum_{i_2=1}^{p+1} R_{i_1 i_2}^p(\xi, \eta) \mathbf{x}_{i_1 i_2} \quad (18)$$

Complex geometries, however, require a high-degree basis when represented using a single polynomial patch. This may present several drawbacks from the numerical point of view (35). Therefore, as explained in (1), modern CAD software representations are commonly based on B-Splines and NURBS, rather than on Bézier curves. These bases are thus used in classical Isogeometric Analysis (1) to construct Continuous Galerkin (CG) approximations.

However, due to the piecewise nature of the B-Spline basis functions, NURBS cannot be directly used in a DG discretization (16). Hopefully, starting from each NURBS patch, a set of rational Bézier patches can be extracted, without altering the CAD geometry (35). This

procedure is called Bézier extraction, and, as explained in (16), allows the generation of a DG-compliant representation, starting from CAD data. In fact, Rational Bernstein functions (17) allow to generate discontinuous solutions at element interfaces. In the next section we thus present a DG approach based on rational Bézier patches.

### 3.2 Formulation

The rational Bernstein functions (17) are defined on a parametric domain, therefore, as in Isogeometric Analysis, the elements are mapped from the physical space to the parametric domain. In this context, a mapped ALE approach would require an additional map function to simulate the moving domain, resulting in a complex and cumbersome implementation. For this reason we opt to solve the equations in the moving referential domain. In particular, the formulation (8) is chosen, as the implementation of the strong form ALE-DG scheme (6) would represent a major redesign of the framework we proposed in (16).

We assume that, in general, the physical flux can be expressed as:

$$\mathbf{F} = \mathbf{F}_c(\mathbf{W}) - \mathbf{F}_v(\mathbf{W}, \nabla \mathbf{W}) \quad (19)$$

where  $\mathbf{F}_c$  is the convective flux and  $\mathbf{F}_v$  is the viscous flux.  $\mathbf{G} = \nabla \mathbf{W}$  is the gradient of the conservative variables. The second order derivatives are discretized with the Local Discontinuous Galerkin (LDG) approach (36). We thus rewrite eq. (2) as a system of first order equations:

$$\begin{cases} \frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathbf{F}_c(\mathbf{W}) - \nabla \cdot \mathbf{F}_v(\mathbf{W}, \mathbf{G}) - \mathbf{V}_g \cdot \nabla \mathbf{W} = 0, \\ \mathbf{G} - \nabla \mathbf{W} = 0. \end{cases} \quad (20)$$

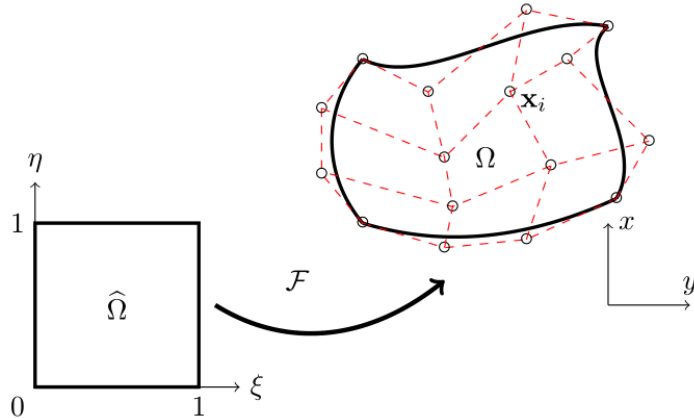


Figure 1: example of Bézier patch

In the NURBS-based DG framework, each element is a rational Bézier patch, as represented in figure 1. The geometry  $\mathbf{x}$  and the local solution fields  $\mathbf{w}_h$  and  $\mathbf{g}_h$  are expressed using the same basis functions:

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{w}_h \\ \mathbf{g}_h \end{pmatrix} = \sum_{i=1}^{(p+1)^2} R_i(\xi, \eta) \begin{pmatrix} \mathbf{x}_i \\ \mathbf{w}_i \\ \mathbf{g}_i \end{pmatrix}, \quad (21)$$



where  $R_i(\xi, \eta)$  are the rational Bernstein functions  $R_{i_1, i_2}^p(\xi, \eta)$  with a trivial index change and omission of degree  $p$ . The isogeometric paradigm is adopted to extend the scheme (8) and discretize eq. (20). Using the map defined by the rational Bézier functions on each element  $\Omega_j$ , the integrals are transposed from the physical space to the parametric unit square  $\widehat{\Omega} = [0, 1]^2$ , obtaining the NURBS-based ALE-DG formulation:

$$\left\{ \begin{array}{l} \frac{d}{dt} \left( \mathbf{w}_i \int_{\widehat{\Omega}} R_k R_i |J_{\Omega_j}| d\widehat{\Omega} \right) = \int_{\widehat{\Omega}} \nabla R_k \cdot (\mathbf{F}_c - \mathbf{F}_v - \mathbf{V}_g \mathbf{w}_h) |J_{\Omega_j}| d\widehat{\Omega} \\ \quad - \oint_{\partial \widehat{\Omega}} R_k (\mathbf{F}_{ale}^* - \mathbf{F}_v^*) |J_{\Gamma_j}| d\widehat{\Gamma}, \\ \mathbf{g}_i \int_{\widehat{\Omega}} R_k R_i |J_{\Omega_j}| d\widehat{\Omega} = \int_{\widehat{\Omega}} \nabla R_k \mathbf{w}_h |J_{\Omega_j}| d\widehat{\Omega} - \oint_{\partial \widehat{\Omega}} R_k \mathbf{W}^* |J_{\Gamma_j}| d\widehat{\Gamma}. \end{array} \right. \quad (22a)$$

$$\left\{ \begin{array}{l} \frac{d}{dt} \left( \mathbf{w}_i \int_{\widehat{\Omega}} R_k R_i |J_{\Omega_j}| d\widehat{\Omega} \right) = \int_{\widehat{\Omega}} \nabla R_k \cdot (\mathbf{F}_c - \mathbf{F}_v - \mathbf{V}_g \mathbf{w}_h) |J_{\Omega_j}| d\widehat{\Omega} \\ \quad - \oint_{\partial \widehat{\Omega}} R_k (\mathbf{F}_{ale}^* - \mathbf{F}_v^*) |J_{\Gamma_j}| d\widehat{\Gamma}, \\ \mathbf{g}_i \int_{\widehat{\Omega}} R_k R_i |J_{\Omega_j}| d\widehat{\Omega} = \int_{\widehat{\Omega}} \nabla R_k \mathbf{w}_h |J_{\Omega_j}| d\widehat{\Omega} - \oint_{\partial \widehat{\Omega}} R_k \mathbf{W}^* |J_{\Gamma_j}| d\widehat{\Gamma}. \end{array} \right. \quad (22b)$$

With respect to the original method, the integrals in (22) contain the additional metric terms  $|J_{\Omega_j}|$  and  $|J_{\Gamma_j}|$ , which are the Jacobians of the coordinate transformation defined by equation (21). Furthermore, the gradient of the basis functions on each element  $\Omega_j$ , in the physical space, is computed by means of the Jacobian matrix of the isogeometric map  $J_{\Omega_j}$ :

$$\nabla R_k = J_{\Omega_j}^{-T} \widehat{\nabla} R_k, \quad (23)$$

where  $\widehat{\nabla} R_k$  is the gradient of the basis function in the parametric domain. The Gauss-Legendre quadrature rule is employed to approximate the integrals. The viscous numerical fluxes  $\mathbf{F}_v^*$  and  $\mathbf{W}^*$  are computed with the LDG approach (36). For the numerical flux function  $\mathbf{F}_{ale}^*$ , any existing Riemann solver can be adapted. It is indeed possible to write the Jacobian matrix of the convective physical flux in the direction  $\boldsymbol{\tau}$  as:

$$\mathbf{J}_{ale} \cdot \boldsymbol{\tau} = \mathbf{J}_F \cdot \boldsymbol{\tau} - (\mathbf{V}_g \cdot \boldsymbol{\tau}) \mathbf{I}, \quad (24)$$

with  $\mathbf{J}_F$  the Jacobian of the convective flux on the fixed mesh and  $\mathbf{I}$  the identity matrix. It is thus trivial to show that the eigenvectors of  $\mathbf{J}_{ale}$  are invariant with respect to the grid velocity, whereas for the eigenvalues the following relation holds:

$$S_{ale} = S_0 - \mathbf{V}_g \cdot \boldsymbol{\tau}, \quad (25)$$

where  $S_0$  are the eigenvalues computed without mesh movement. It is thus possible to obtain a consistent numerical flux for moving meshes using the modified wave speeds computed with eq. (25) and adding the flux contribution generated by mesh movement to the fixed grid Riemann solver. Two examples will be provided in section 4.

Equation (22b) does not contain a time derivative, therefore it can be solved separately from eq. (22a) within each time iteration. The system of equations (22) can be thus rewritten in a compact form:

$$\frac{d}{dt} (\mathcal{M} \mathbf{w}) = \mathcal{R}(\mathbf{w}_h, \mathbf{V}_g), \quad (26)$$

where the residual  $\mathcal{R}$  is the right-hand side of eq. (22a) and  $\mathcal{M}$  is the block-diagonal mass matrix, which is generally not constant in time, except for rigid mesh motion. In this paper we consider explicit Runge-Kutta (RK) time integration to solve the system of ordinary differential equations (26). In particular, we either employ the classic four-stage 4th-order Runge-Kutta (RK) scheme or the Strong Stability Preserving (SSP) 3rd-order RK method of Gottlieb et al. (37).

One can observe that, due to the introduction of the isogeometric map, the mathematical structure of the proposed formulation (22) is nearly identical to that of the mapped ALE-DG scheme (14). However, in our approach the map is local to each element and it is only used to transform the integrals of the weak formulation. On the other hand, in the technique proposed by Persson et al. (26), the map is global and it is applied to modify the conservation laws at the continuous level. Similarly to eq. (14), the proposed scheme (22) violates the DGCL in the general case. Indeed, the equivalent of the identity (10) for equation (22a) is:

$$\oint_{\partial\hat{\Omega}} R_k \mathbf{V}_g \cdot \mathbf{n} |J_{\Gamma_j}| d\hat{\Gamma} - \int_{\hat{\Omega}} J_{\Omega_j}^{-T} \hat{\nabla} R_k \cdot \mathbf{V}_g |J_{\Omega_j}| d\hat{\Omega} = 0. \quad (27)$$

When the basis functions  $R_k$  are rational, an exact integration is not achievable by means of Gauss-Legendre quadrature. In the case of uniform weights,  $R_k$  is a polynomial, and the products  $|J_{\Gamma_j}| \mathbf{n}$  and  $|J_{\Omega_j}| J_{\Omega_j}^{-T}$  are polynomials as well. Therefore, the integrals in (27) can be exactly computed by means of numerical quadrature. As a consequence the DGCL is satisfied when the weights are uniform. We decide to not enforce the DGCL for rational functions as, in the majority of applications, uniform weights are employed. When it is not the case, the use of rational functions is usually limited to very small regions around the obstacles, and polynomial Bézier elements are used in the rest of the computational domain. Moreover, as demonstrated in (26), the preservation of constant solutions is not a necessary condition for stability and high-order accuracy.

### 3.3 Mesh movement technique

The second key ingredient of an ALE method is the mesh movement algorithm. Traditional ALE approaches for aerodynamics applications are based on rectilinear grids. Therefore, the grid velocity in each element is defined by the velocities of its vertices, usually computed via elastic analogy (38) or graph theory based methods (9). On the contrary, our formulation is based on a high-order geometric representation, enabling more control over the shape of the elements. Therefore, a wider range of mesh velocities can be employed with respect to traditional approaches. We decide to adopt the isogeometric paradigm to define the grid velocity field  $\mathbf{V}_g$  in each element:

$$\mathbf{V}_g = \sum_{i=1}^{(p+1)^2} R_i(\xi, \eta) \mathbf{v}_{g,i}, \quad (28)$$

with  $\mathbf{v}_{g,i}$  being the velocity of the control point  $\mathbf{x}_i$ . Obviously, we impose the additional constraint that the velocities match exactly at the interface between two elements. This choice leads to a completely unified description of all the variables appearing in the ALE formulation: the solution fields, the geometry and the mesh velocity. Thus, the time evolution of the control point net is determined by:

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_{g,i}. \quad (29)$$

Given the distribution of control point velocities, the eq. (29) is integrated with the same Runge-Kutta method employed for the DG formulation. In the present work, we analyse cases where the mesh movement is imposed in the whole computational domain. We also assume

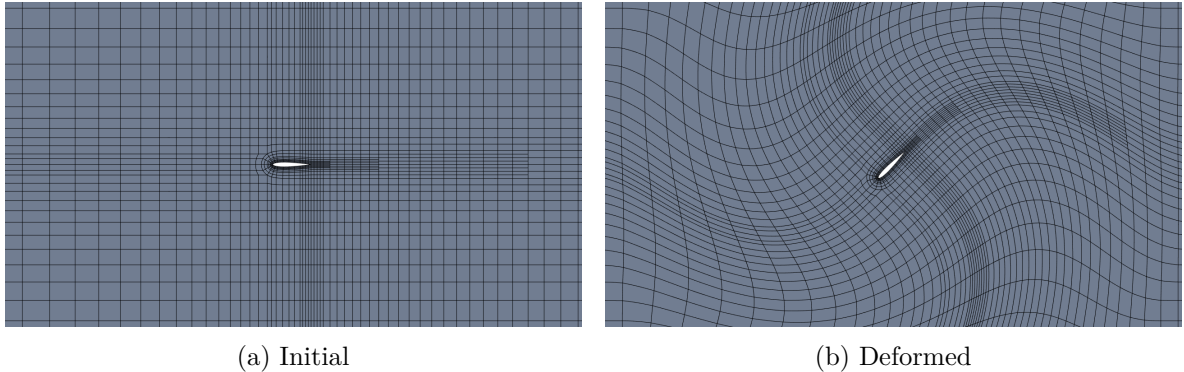


Figure 2: Mesh movement example

that no topology change occurs during the movement. The control point velocities are thus explicitly assigned in each time step.

The proposed technique is capable of generating arbitrarily high-order mesh deformations. Moreover, we can take advantage of regularity properties of Bézier surfaces, such as the convex hull property (35), to preserve the mesh quality over time and avoid tangling. The velocity field  $\mathbf{V}_g$  inside each element is always  $\mathcal{C}^\infty$  thanks to the definition (28), whereas, across the element boundaries, the derivatives can be discontinuous. In order to enhance the regularity at element interfaces, the hierarchical construction of the computational domain can be exploited. As previously explained, the elements are obtained by recursive splitting of a few baseline NURBS patches. When internal knots multiplicities are equal to 1, the inner geometric regularity of these patches is  $\mathcal{C}^{p-1}$ . The control points velocities  $\mathbf{v}_g^0$  of the baseline patches are computed by means of an explicit function of space and time. Thus, the resulting baseline velocity field exhibits also a  $\mathcal{C}^{p-1}$  regularity. The Bézier extraction is then applied to compute the control point velocities  $\mathbf{v}_g$  of each DG element:

$$\mathbf{v}_g = \mathbf{C} \mathbf{v}_g^0, \quad (30)$$

where  $\mathbf{C}$  is the linear Bézier extraction operator. As result, the mesh velocity field  $\mathbf{V}_g$  computed with this algorithm possess a  $\mathcal{C}^{p-1}$  regularity. This hierarchical splitting of the geometrical elements is also an important tool for moving non-conformal meshes. Indeed, a naive application of any non-linear deformations would generate holes between elements in the proximity of hanging nodes.

Furthermore, the hierarchical definition of the mesh velocity facilitates the avoidance of tangling: thanks to the convex hull property (35), an element remains admissible as long as its control points are not overlaid. By applying the velocity function at the coarsest level, this admissibility criterion can be easily imposed because the distance between the control points at this level is usually far larger than the displacement. Fig. 2 illustrates the mesh deformation for a large amplitude movement. The potential and the flexibility of the proposed mesh movement algorithm are also demonstrated in sections 4 and 5.

### 3.4 Shock capturing for compressible flows

The applications presented in this work concern flows governed by the two-dimensional Navier-Stokes equations. Using the formalism of equation (20), the conservative variables and the

physical fluxes are:

$$\mathbf{W} = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho e \end{pmatrix}, \quad \mathbf{F}_{c,i} = \begin{pmatrix} \rho u_i \\ \rho u_1 u_i + p \delta_{1i} \\ \rho u_2 u_i + p \delta_{2i} \\ \rho u_i (e + \frac{p}{\rho}) \end{pmatrix}, \quad \mathbf{F}_{v,i} = \begin{pmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ u_k \tau_{ki} - q_i \end{pmatrix}, \quad (31)$$

where  $\tau_{ij}$  is the viscous stress tensor and  $q_i$  is the thermal conduction flux, defined as:

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_k}{\partial x_k} \delta_{ij}, \quad (32)$$

$$q_i = -\gamma \frac{\mu}{Pr} \frac{\partial e}{\partial x_i}, \quad (33)$$

where  $\gamma = 1.4$ ,  $Pr = 0.72$  and  $\mu$  is determined by the Reynolds number. When Euler equations are considered, the viscous flux  $\mathbf{F}_v$  is set equal to 0 and the numerical algorithm just solves eq. (22a).

Compressible flows are characterized by the presence of shocks. It is therefore critical for a robust numerical algorithm to be able to accurately capture discontinuities. Two approaches are commonly used for DG schemes. The first method consists in introducing an artificial viscosity term in the conservation law (39; 40), whereas the second technique is inspired by Finite Volume slope limiting (41; 42). The approach employed here adapts the original idea presented in (39) to rational elements, and it was tested in the context of the proposed method in (16), on a simple problem with a fixed geometry. A constant by element artificial viscosity  $\varepsilon$  is added to  $\mu$  in eq. (32), when a shock is detected in the element. Its value is computed with the following smooth function:

$$\varepsilon = \begin{cases} 0 & \text{if } s_j \leq s_0 - \kappa, \\ \frac{\varepsilon_0}{2} \left( 1 + \sin \frac{\pi(s_j - s_0)}{2\kappa} \right) & \text{if } s_0 - \kappa < s_j < s_0 + \kappa, \\ \varepsilon_0 & \text{if } s_j \geq s_0 + \kappa, \end{cases} \quad (34)$$

where  $\varepsilon_0 = d/(p+1)$ ,  $\kappa = s_0$  and  $s_0 = \tilde{s}_0/(p+1)^2$ . The shock sensor  $s_j$  gives a measure of the oscillatory nature of the solution in each element, and it is based on the total variation of the control points of the density field in the two directions (16):

$$s_\xi^{i_2} = \left( \sum_{i_1=1}^p |\rho_{i_1+1, i_2} - \rho_{i_1, i_2}| \right) - |\rho_{p+1, i_2} - \rho_{1, i_2}|, \quad (35)$$

$$s_\eta^{i_1} = \left( \sum_{i_2=1}^p |\rho_{i_1, i_2+1} - \rho_{i_1, i_2}| \right) - |\rho_{i_1, p+1} - \rho_{i_1, 1}|. \quad (36)$$

The sensor for a given element is finally defined as:

$$s_e = \frac{1}{2(p+1)} \left( \sum_{i_2=1}^{p+1} s_\xi^{i_2} + \sum_{i_1=1}^{p+1} s_\eta^{i_1} \right) \frac{1}{\bar{\rho}}, \quad (37)$$

with  $\bar{\rho}$  being the average density.  $\tilde{s}_0$  is a user-defined parameter to adjust the sensitivity of the shock capturing algorithm. In this study,  $\tilde{s}_0 = 10$ .

## 4 Verification

### 4.1 Advection equation

The proposed approach is firstly tested on a scalar advection problem. As numerical flux function we adopt a modified Local Lax-Friedrichs flux that suits the Arbitrary Lagrangian-Eulerian formulation:

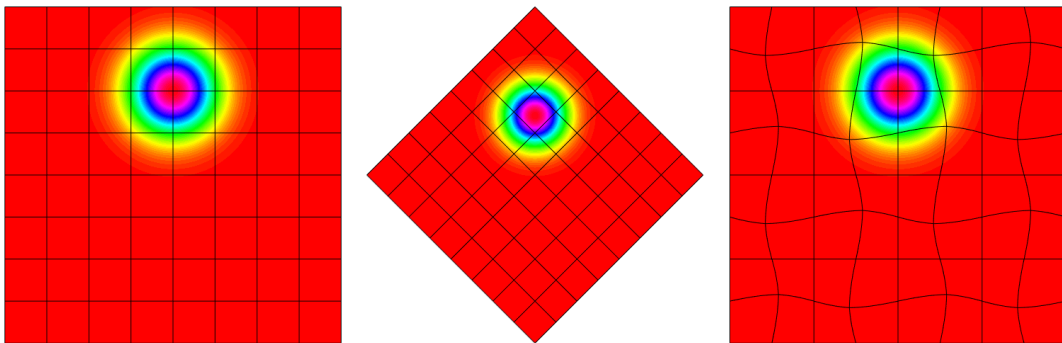
$$\mathbf{F}_{ale}^*(w^+, w^-, \mathbf{V}_g, \mathbf{n}) = \frac{1}{2}(w^+ + w^-)(\mathbf{V} - \mathbf{V}_g) \cdot \mathbf{n} + \frac{1}{2}|(\mathbf{V} - \mathbf{V}_g) \cdot \mathbf{n}|(w^+ - w^-), \quad (38)$$

where  $w$  is the scalar variable and  $\mathbf{V}$  is the advection velocity, that we assume constant in space. It is straightforward to verify that the proposed numerical flux complies with the consistency condition (9). A convergence analysis is performed on the following test case:

$$\begin{cases} w_0(\mathbf{x}) = \exp(-(x-2)^2 - y^2), \\ u(t) = -4\pi \sin(2\pi t), \\ v(t) = 4\pi \cos(2\pi t). \end{cases} \quad (39)$$

The initial solution is advected along an anticlockwise circle of radius 2, thus, the analytic solution is:

$$w(\mathbf{x}, t) = \exp\left(-[x - 2\cos(2\pi t)]^2 - [y - 2\sin(2\pi t)]^2\right). \quad (40)$$



(a) Fixed mesh

(b) Rotating mesh

(c) Deforming mesh

Figure 3: Advection test case, solution at  $t=0.25$ ,  $p=5$

The computational domain is  $[-4, 4] \times [-4, 4]$ , discretized with a Cartesian grid, and the boundary numerical fluxes are computed by means of the exact solution. The error in  $L^2$ -norm is evaluated using numerical quadrature after one rotation period. In order to assess the accuracy of the numerical scheme, two mesh velocity laws are tested, as illustrated in Fig. 3, a rigid clockwise rotation about the origin and a sinusoidal deformation where the control points move with the following velocity:

$$u_g(\mathbf{x}, t) = v_g(\mathbf{x}, t) = \sin\left(\frac{N_x\pi}{L_x}x\right) \sin\left(\frac{N_y\pi}{L_y}y\right) \sin(2\pi t). \quad (41)$$

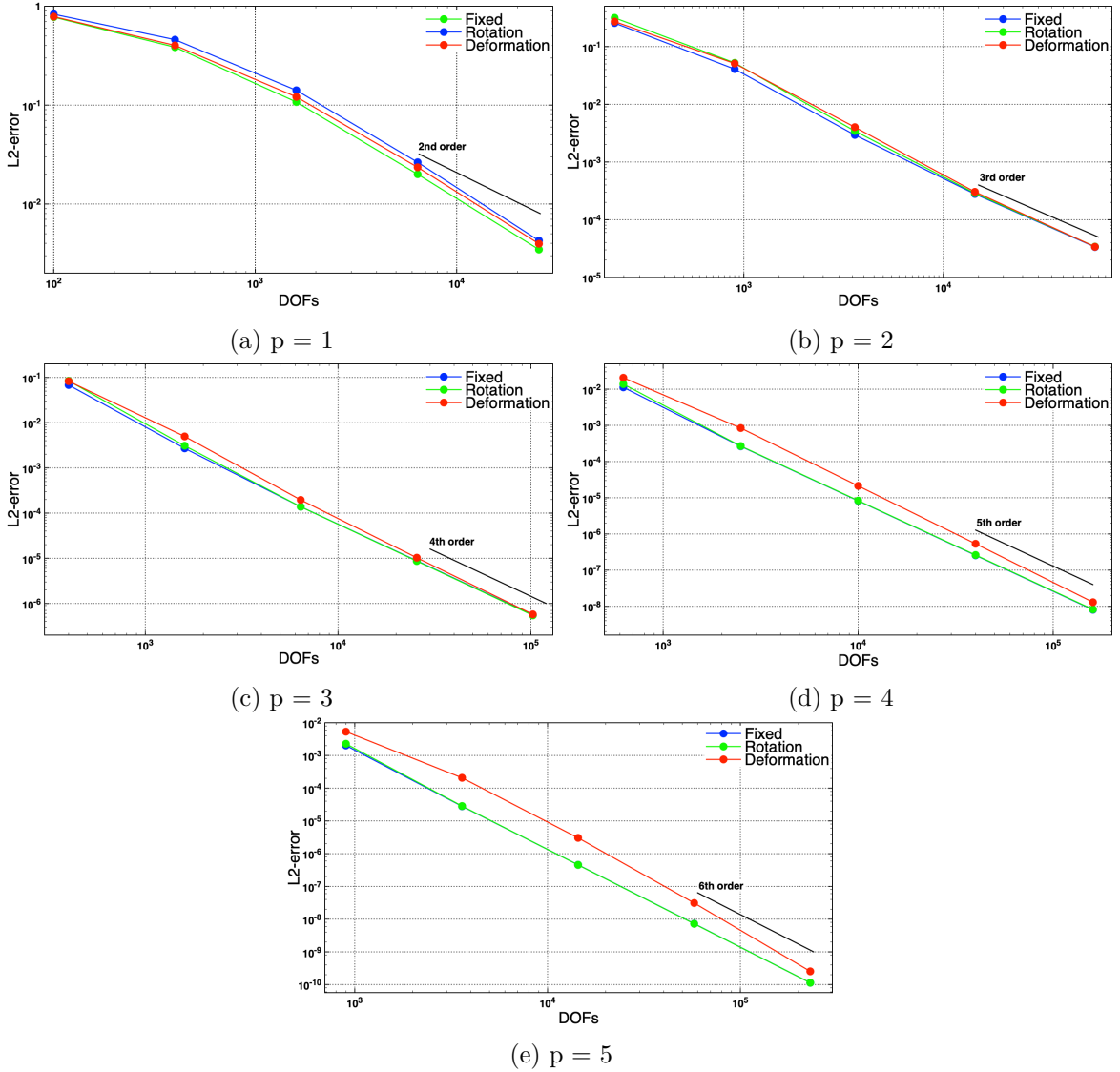


Figure 4: Convergence analysis, 2D advection equation

We compare the results obtained with the 3 configurations. Polynomials up to fifth degree are tested and optimal convergence rates are verified for each degree, as shown in figure 4. It is also possible to observe that results on rigidly moving mesh do not substantially differ from the fixed domain simulation, whereas the error increases when the mesh deforms over time, especially for high polynomial degrees. Defining  $r_d$  as the ratio between the errors computed respectively on the deforming and the fixed mesh:

$$r_d = \frac{\epsilon_{deforming}}{\epsilon_{fixed}}, \quad (42)$$

the maximum value of  $r_d$  for cubic functions is around 1.8, whereas for quartic polynomials a factor 3.2 is found in the worst case and for the quintic basis a value of 7.4 is reached. This steep increase is caused by the loss of accuracy of the Bézier interpolation on distorted elements. Indeed, as proved by Bazilevs et al. (43), the norm of the interpolation error

depends on the Jacobian matrix of the isogeometric map.

## 4.2 Euler equations

In order to validate the proposed approach for non-linear problems, we consider the compressible Euler equations for the second test case. The numerical flux is computed using the HLL Riemann solver (44), which has been adapted to take into account the mesh movement flux:

$$\mathbf{F}_{ale}^* = \begin{cases} \mathbf{F}_{ale}^- = \mathbf{F}(\mathbf{w}_h^-) - (\mathbf{V}_g \cdot \mathbf{n})\mathbf{w}_h^- & \text{if } S_{ale}^- \geq 0 \\ \frac{S_{ale}^+ \mathbf{F}_{ale}^- - S_{ale}^- \mathbf{F}_{ale}^+ + S_{ale}^+ S_{ale}^- (\mathbf{w}_h^+ - \mathbf{w}_h^-)}{S_{ale}^+ - S_{ale}^-} & \text{if } S_{ale}^- \leq 0 \leq S_{ale}^+ \\ \mathbf{F}_{ale}^+ = \mathbf{F}(\mathbf{w}_h^+) - (\mathbf{V}_g \cdot \mathbf{n})\mathbf{w}_h^+ & \text{if } S_{ale}^+ \leq 0 \end{cases} \quad (43)$$

where  $S_{ale}^-$  and  $S_{ale}^+$  are respectively the minimum and the maximum wave speeds computed with the two states  $\mathbf{w}_h^-$  and  $\mathbf{w}_h^+$ . The proposed numerical flux function (43) respects the consistency condition (9). The advection of an isentropic vortex (33) is considered, with the following analytic solution:

$$\begin{cases} \rho = \left(1 - \frac{\gamma - 1}{16\gamma\pi^2} \beta^2 e^{2(1-r^2)}\right)^{\frac{1}{\gamma-1}} \\ u = 1 - \beta \frac{y - y_0}{2\pi} e^{1-r^2} \\ v = \beta \frac{x - t - x_0}{2\pi} e^{1-r^2} \\ p = \rho^\gamma \end{cases} \quad (44)$$

where  $r = \sqrt{(x - t - x_0)^2 + (y - y_0)^2}$ , with  $x_0 = 5$ ,  $y_0 = 0$  and  $\beta = 5$ . The computational domain is  $[0, 10] \times [-5, 5]$  and the boundary conditions are weakly imposed using the analytic solution. The simulation is run on a fixed Cartesian grid and on a deforming mesh, whose control points follow the sinusoidal law (41) used for the advection test case.

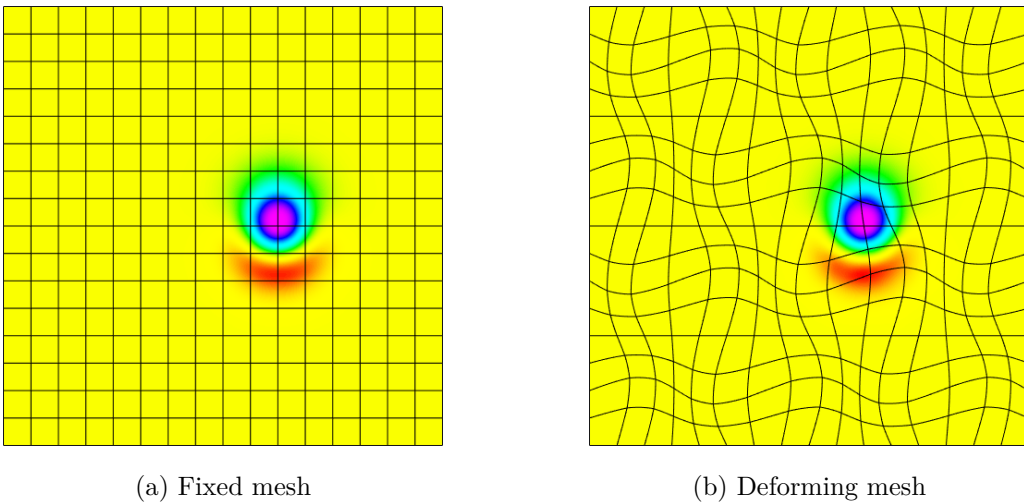


Figure 5: Isentropic vortex, energy,  $t=1.25$ ,  $p=3$

We compute the  $L^2$ -norm of the error of the total energy at time  $t = 2$ . Polynomials up to fifth degree are tested. Optimal convergence rates are verified in both cases for each degree as it is shown in Fig. 6, where it is also possible to compare the error levels between the two setups. The evolution of the ratio  $r_d$  (42) confirms the results obtained for the advection equation: for cubic polynomials the maximum ratio is around 1.8, whereas for the quartic basis a factor 3.5 is found in the worst case and for quintic functions a value of 6 is reached. The sensitivity with respect to mesh deformation is a natural consequence of the interpolation error, as already evidenced for the advection problem. The proposed numerical scheme is hence capable of accurately solving non-linear conservation equations on moving meshes, preserving the high-order convergence rate of the DG discretization.

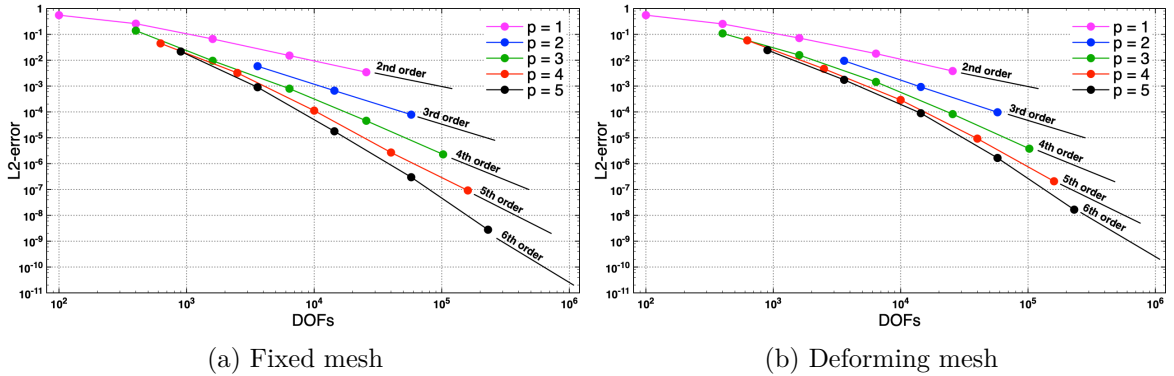


Figure 6: Convergence analysis, Euler equations,  $L^2$ -error of total energy

## 5 Case study

### 5.1 Oscillating cylinder

In the next example we exploit the capability of rational Bézier curves of exactly representing conic shapes. We simulate the bidimensional viscous flow around a circular cylinder oscillating in the crossflow direction. Thus, the Navier-Stokes equations, defined by Eq.(31-33), are solved. Using 4 rational Bézier curves, the circular boundary can be exactly described and a very coarse baseline mesh can be generated. The external boundary of the computational domain is the rectangle  $[-25D, 100D] \times [-25D, 25D]$ , where  $D$  is the diameter, and the center of cylinder is the origin  $(0,0)$ . The actual computational mesh is obtained by locally refining the initial patches, without the need of refitting, since the coarse grid already provides an exact representation of the domain. In order to perform a mesh convergence study, 3 different refinement levels, shown in Fig. 7, are used for the simulations:

- coarse: 1065 elements, with 16 rational Bézier arcs on the cylinder,
- intermediate: 2145 elements, with 32 rational Bézier arcs on the cylinder,
- fine: 4455 elements, with 64 rational Bézier arcs on the cylinder.

The oscillating cylinder flow has been widely investigated in the literature, using both experimental and numerical techniques (25; 45; 46). It is characterised by complex non-linear physics, such as the lock-in phenomenon, that consists in the synchronisation of the vortex



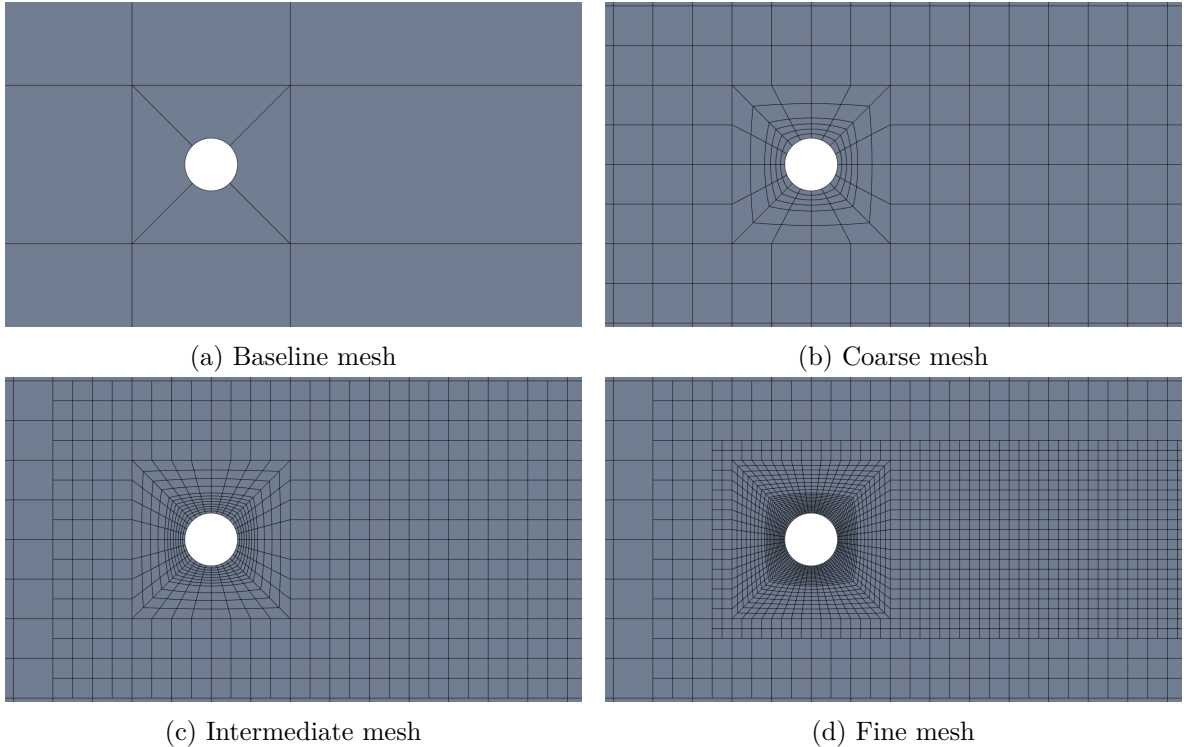


Figure 7: Different refinement levels for the cylinder simulation

shedding with the oscillation of the body. Due to its non-linear nature, the problem is extremely sensitive to acoustic perturbations, therefore, it is important to avoid the interference between the acoustic waves possibly reflected at the boundaries and the flow in the vicinity of the cylinder. This is achieved by choosing a sufficiently large computational domain and a layer of coarse cells in the proximity of the external boundaries. Moreover, far-field boundary conditions are implemented in a weak sense using Riemann invariants to avoid reflection effects. The cylinder surface is modelled as an adiabatic no-slip wall and a weakly prescribed boundary condition is employed (47). The initial state corresponds to uniform fields, whose values are determined according to the freestream Mach number.

The convergence study is conducted using rational functions built from polynomials of degrees from 3 up to 6. We consider a freestream Mach number equal to 0.2, in order to avoid strong compressibility effects, and a Reynolds number of 500 with respect to the diameter. The cylinder motion is given by:

$$y = A \sin(2\pi ft), \quad (45)$$

with  $A = 0.25D$  and  $f = 0.875f_{sh}$ , with  $f_{sh}$  being the vortex shedding frequency of the fixed cylinder at equal Mach and Reynolds numbers. The chosen configuration of  $A$  and  $f$  lies in the lock-in range. For each combination of degree and refinement level the oscillation frequency  $f$  is determined from the value of  $f_{sh}$  obtained with the corresponding fixed mesh simulation.

A first set of results, shown in Fig. 8, is obtained by running the test case with a rigid mesh movement law, in which the entire grid moves at the velocity of the cylinder. Once a periodic solution is established, we examine more precisely the peak lift coefficient  $\hat{C}_l$ , the average

	$St$
Present work	0.225
Blackburn et al. (46)	0.228
Lu et al. (48)	0.222
Nguyen (25)	0.218
Roshko (Experimental) (49)	0.20 - 0.22

Table 1: Comparison of Strouhal number with reference data

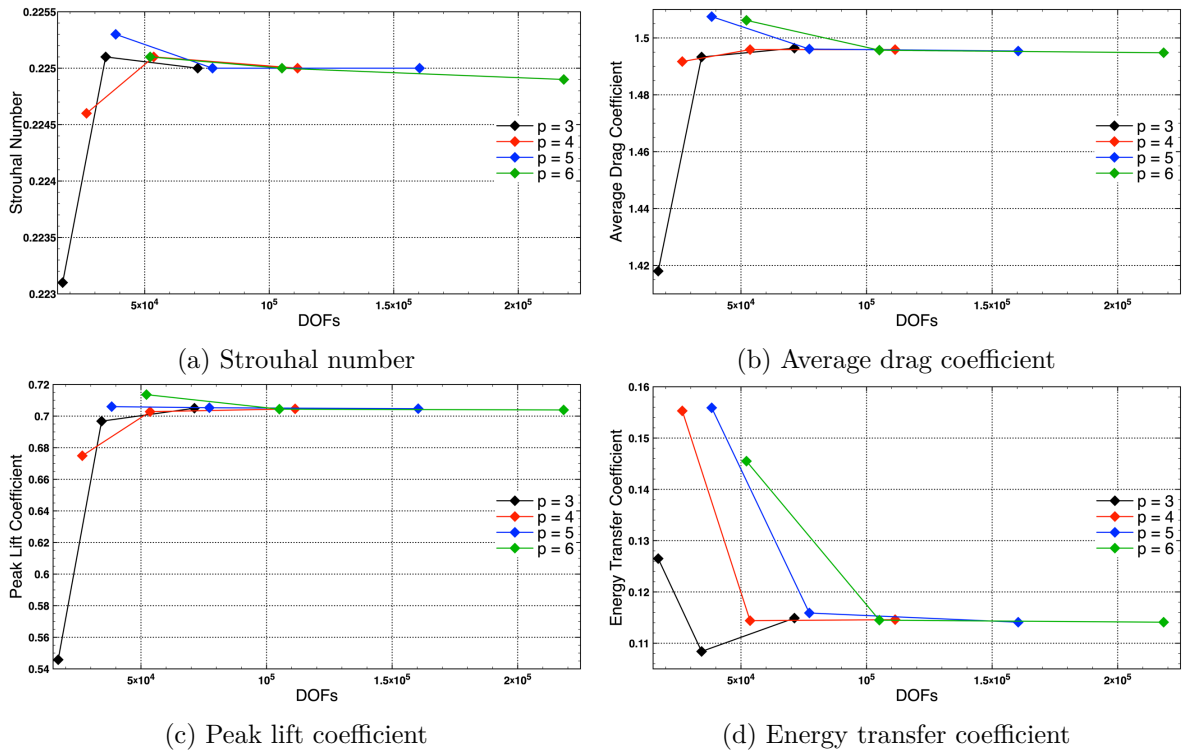


Figure 8: Oscillating cylinder flow, convergence study

drag coefficient  $\bar{C}_d$ , and the energy transfer coefficient  $E$ , which quantifies the mechanical work done by the fluid on the cylinder over a lock-in cycle:

$$E = \frac{1}{\frac{1}{2}\rho_\infty u_\infty^2 D^2} \oint \mathbf{f} \cdot d\mathbf{x} = \oint \frac{L}{\frac{1}{2}\rho_\infty u_\infty^2 D} d\left(\frac{y}{D}\right) = \oint C_l d\alpha, \quad (46)$$

where  $\mathbf{f}$  is the force vector,  $\mathbf{x}$  the displacement,  $L$  the lift force and  $\alpha$  the dimensionless displacement in the  $y$  direction. The convergence of the Strouhal number of the fixed cylinder is illustrated for the sake of completeness, and a comparison of the converged value with reference data is reported in table 1. The lock-in limit cycle is well reproduced with all the basis degrees and a faster convergence is observed with functions of degree 4,5 and 6. Note that using high-order bases on the coarsest mesh does not lead to significant improvements. In order to estimate  $E$  with sufficient precision, it is necessary to use at least the intermediate refinement level. Hence, increasing the basis degree on coarse grids becomes inefficient, indeed, the severe stability restrictions due to explicit time stepping significantly raises the

computational costs without benefits in terms of accuracy. Therefore, a combination of  $h$  and  $p$  refinements has to be used in order to optimise the computational cost, for a given level of accuracy. The numerical solution of the lock-in limit cycle is compared to the results provided by Blackburn et al. (46). The comparison is presented in Fig. 9b, the lift-displacement curve obtained with the proposed scheme is in close comparison with the reference cycle.

The proposed test case is also used to evaluate the influence of the grid movement law in the presence of moving walls. Therefore, the results obtained with rigid mesh displacements are compared to the simulations performed with a smooth deformation velocity field, where the control point net moves with the following law:

$$v_g(\mathbf{x}, t) = v_c(t) e^{-\frac{\psi^2(\mathbf{x}, t)}{d^2}} \quad (47)$$

where  $v_c(t)$  is the velocity of the cylinder,  $\psi(\mathbf{x}, t)$  is the distance of the point  $\mathbf{x}$  from the cylinder surface at time  $t$  and  $d$  is a characteristic length, that controls the propagation of the boundary displacement within the domain. As shown in Fig. 10b, this profile allows the elements close to the wall to move almost rigidly, preserving the initial mesh quality. At the same time, thanks to the exponential decay far from the cylinder, only a small portion of the domain is deformed.

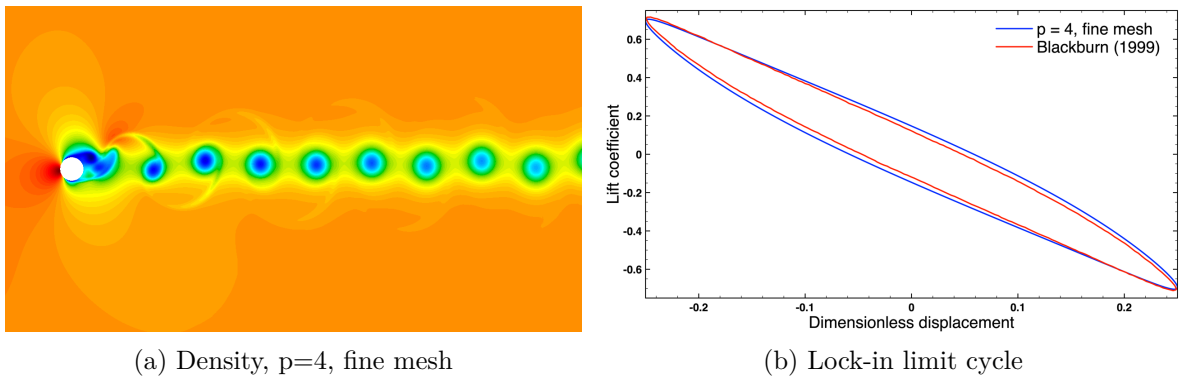


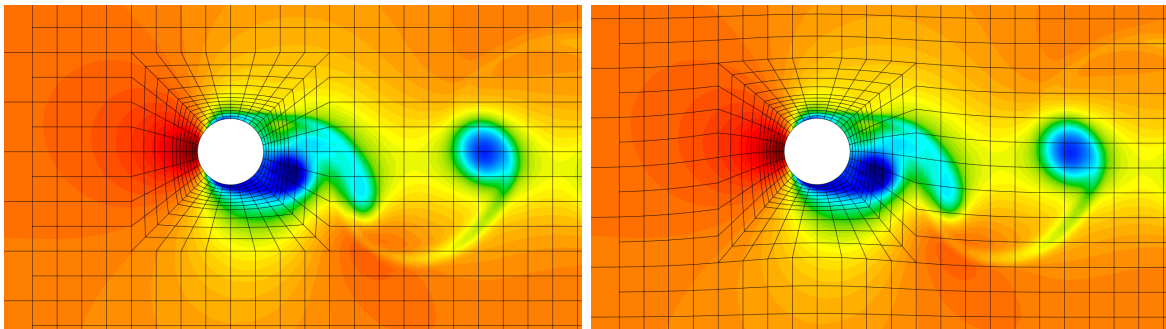
Figure 9: Numerical solution of the oscillating cylinder test case

In Fig. 10a and 10b the comparison of the density field in the periodic regime for the rigid and deforming mesh movement laws is presented for the intermediate mesh level and quintic basis. The two solutions are nearly identical and, for the deforming mesh, no spurious effects generated by the lack of freestream preservation can be observed, even if the numerical quadrature of rational functions is less accurate. The same conclusion is obtained by comparing the evolution of the aerodynamic coefficients, reported in Fig. 10c for the same mesh and basis configuration.

An extensive comparison of the two mesh movement techniques has been carried out, for all the combination of meshes and basis degrees. We report in table 2 the energy transfer coefficient, as this quantity is more sensitive than the force coefficients. For a given basis degree, a discrepancy in the third significant figure can be observed for coarse meshes. However, both values are significantly far from the converged result. Refining the grid, the difference between the two movement techniques becomes smaller and the predicted values are equally accurate.

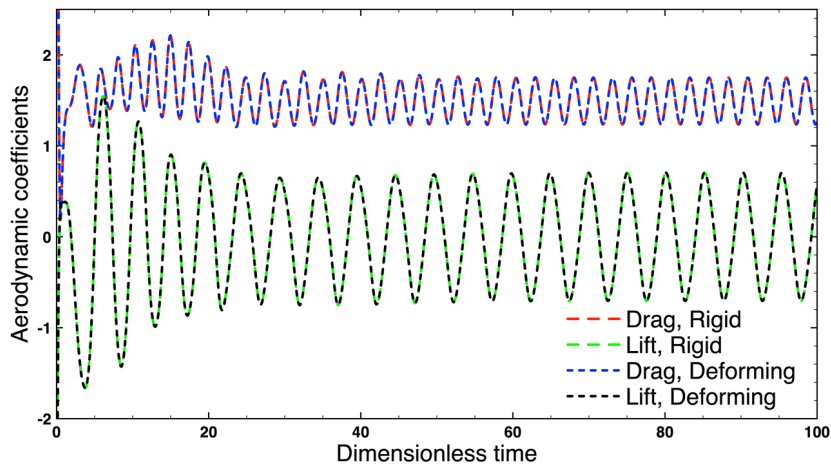
degree	Rigid			Deformation		
	coarse	intermediate	fine	coarse	intermediate	fine
3	0.1265	0.1084	0.1149	0.1298	0.1086	0.1151
4	0.1553	0.1144	0.1146	0.1566	0.1150	0.1148
5	0.1559	0.1159	0.1141	0.1590	0.1154	0.1141
6	0.1455	0.1145	0.1141	0.1422	0.1145	0.1141

Table 2: Energy transfer coefficient for different mesh movement laws



(a) Density field, rigid mesh

(b) Density field, deforming mesh



(c) Force coefficients

Figure 10: Comparison of movement laws,  $p=5$ , intermediate mesh

## 5.2 Pitching airfoil in subsonic flow

In the second case study, the subsonic flow around a pitching NACA 0012 airfoil is investigated. Contrarily to the cylinder case, the geometry cannot be exactly represented with NURBS functions, therefore, the boundary of the computational domain is approximated using polynomial curve fitting. We employ a cubic Bernstein basis for both the geometry and the solution fields. The goal of the proposed test case is to assess the impact of using curved grids. This is achieved via a double convergence study: the first is performed using the high-order boundary representation, whereas a piecewise linear approximation is used for the second. For each refinement level, the low-order grid is obtained by linearizing the respective

curved mesh. Both inviscid and viscous flow models are considered, as shown in Fig. 11.

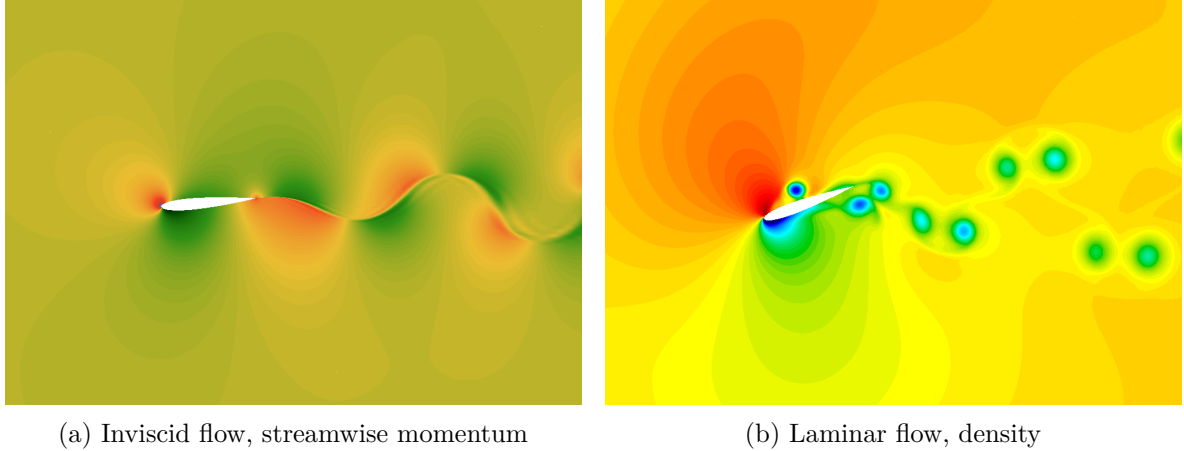


Figure 11: Pitching airfoil, solution fields

The airfoil is subject to a pure pitching motion about mid-chord. The angle of attack oscillates in time with a sinusoidal law:

$$\alpha(t) = -A \sin(2\pi ft), \quad (48)$$

with  $A$  and  $f$  being respectively the pitch amplitude and frequency. The rotation of the airfoil is transferred to the mesh through a piecewise defined velocity field:

$$\begin{cases} u_g(\mathbf{x}, t) = -\tilde{\omega} (y - y_c) \\ v_g(\mathbf{x}, t) = \tilde{\omega} (x - x_c) \end{cases} \quad (49)$$

where  $(x_c, y_c)$  is the mid-chord position and  $\tilde{\omega} = \dot{\alpha} \sigma(\mathbf{x}, t)$ , with  $\sigma$  being a blending function of the following form:

$$\sigma(\mathbf{x}, t) = \begin{cases} 1, & \text{if } R \leq R_{int} \\ \frac{1}{2} \left[ 1 + \cos \left( \pi \frac{R - R_{int}}{R_{ext} - R_{int}} \right) \right], & \text{if } R_{int} < R < R_{ext} \\ 0, & \text{if } R \geq R_{ext} \end{cases} \quad (50)$$

with  $R$  the distance with respect to the mid-chord position. The blending function divides the domain into 3 regions, the internal one, delimited by a circle of radius  $R_{int}$ , moves rigidly with the airfoil, while the area outside the circle of radius  $R_{ext}$  is fixed. The 2 regions are connected by a deforming transition ring. The proposed function (50) is  $C^1$ , in order to have a smoother deformation profile, however a  $C^0$  class function could be employed as well.

The proposed problem is first studied using an inviscid flow model, based on the compressible Euler equations, and then with a viscous model, based on compressible Navier-Stokes equations. In both cases, the initial state corresponds to uniform fields, defined according to the freestream Mach number. As for the cylinder test-case, the far-field conditions are weakly imposed using Riemann invariants. For the inviscid example, a slip wall boundary condition is imposed on the airfoil surface using a weak Riemann approach (47), whereas, the adiabatic no-slip wall is employed in the viscous case.

### 5.2.1 Inviscid flow

We first consider an inviscid flow configuration, the freestream Mach number  $M_\infty$  is set to 0.2, a pitch amplitude of  $5^\circ$  is considered, with a reduced frequency  $k = \pi f c / U_\infty$  of 0.25, where  $c$  is the chord length and  $U_\infty$  the freestream flow velocity. The streamwise momentum field  $\rho u$  is illustrated in Fig. 11a. The main physical feature is the slip line that develops from the trailing edge of the airfoil due to the pitching motion. Once the transitory effects have vanished, the solution converges to a periodic flow regime, it is hence possible to compute an average drag coefficient  $\bar{C}_d$  and a peak lift coefficient  $\hat{C}_l$ . The meshes adopted for the inviscid test case are reported in Fig. 12.

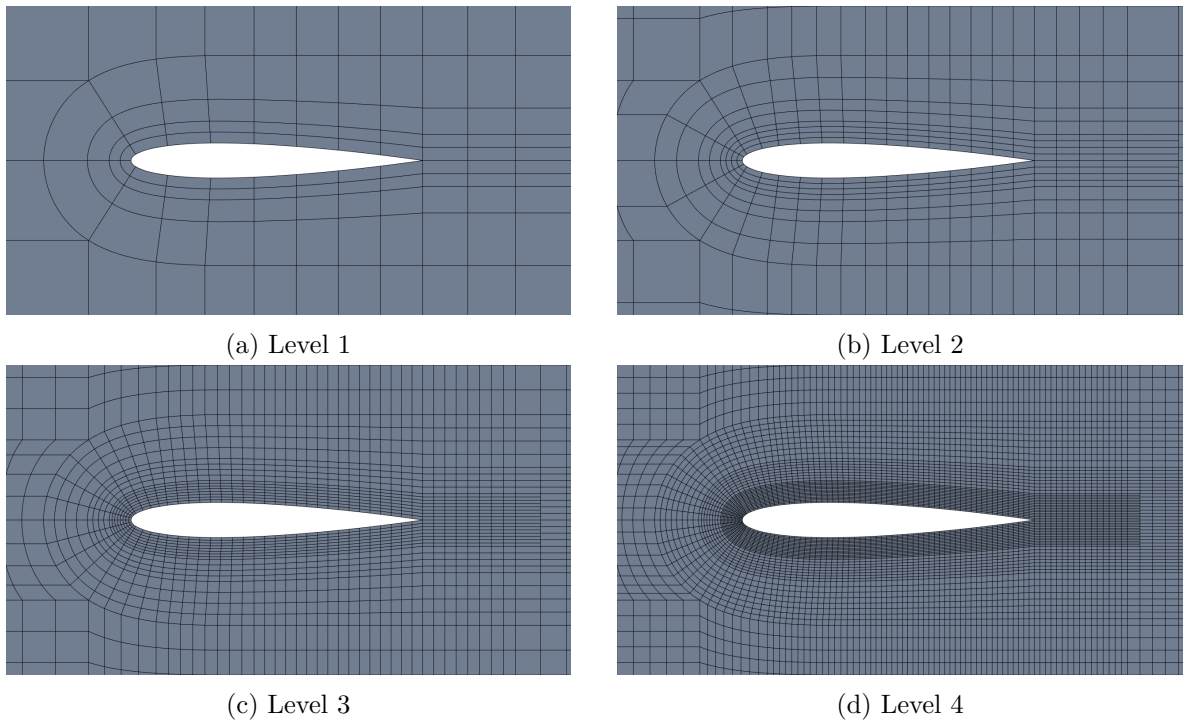


Figure 12: Meshes for the inviscid test case

In Fig. 13 we compare the results obtained using the high-order and piecewise linear grids. In particular, the density fields are presented in Fig. 13a and 13b for the coarsest mesh. The spurious effects introduced by the piecewise linear mesh can be easily noticed, non-physical expansion fans develop at each boundary vertex, generating further numerical oscillations. On the contrary, the high-order mesh guarantees a physically coherent solution even when the mesh is very coarse. The strong influence of the geometry description is caused by the flow-tangency boundary condition, which depends on the normal vector. For piecewise linear approximations, the normal vector is piecewise constant, therefore, across each element the boundary condition presents a discontinuity that creates the non-physical oscillations. This behaviour has already been evidenced in (11; 16). The observations are confirmed by the convergence study, illustrated in Fig. 13c and 13d for  $\bar{C}_d$  and  $\hat{C}_l$ . The aerodynamic coefficients converge considerably faster when the curved boundary representation is adopted. Moreover, the error decreases monotonically for the high-order mesh.

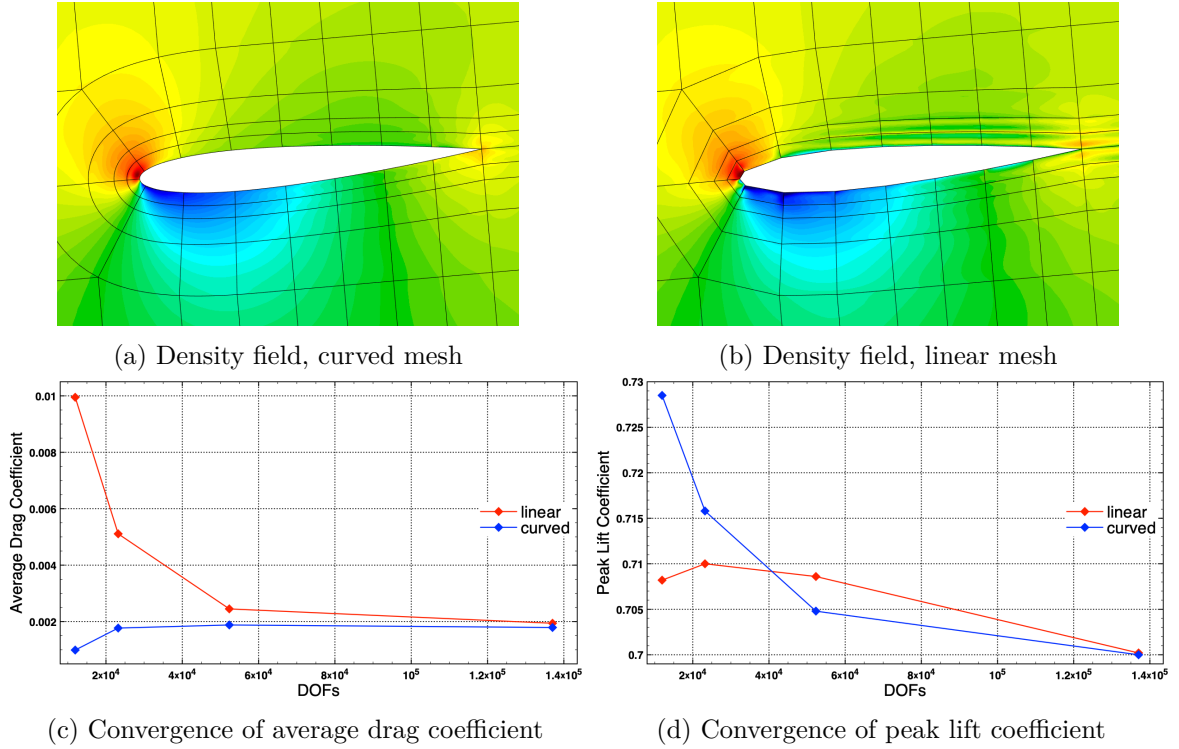


Figure 13: Influence of the geometry, inviscid flow

### 5.2.2 Laminar flow

In the second part of the case study, we repeat the convergence analysis for a viscous fluid. The Reynolds number with respect to the chord length is equal to 1000. The freestream Mach number and the pitching frequency are unchanged, whereas the pitch amplitude is increased to  $20^\circ$ , in order to generate a massively separated flow. Indeed, the considered setup leads to a dynamic stall of the airfoil with a periodic shedding of vortex pairs, as shown in Fig. 11b, where the density field is represented. The meshes adopted for the laminar test case are reported in Fig. 14.

In Fig. 15a and 15b it is possible to compare the density fields computed with the high-order and the linearized mesh. Despite the low grid resolution, only very small discrepancies between the two solutions can be spotted, on the suction side of the airfoil close to the leading edge, but, overall, the two fields are very similar. The aerodynamic forces obtained with the linearized geometry are comparable to those computed using the curved boundary too. Even for the coarsest mesh, the values of the aerodynamic coefficients are nearly identical, as reported in Fig. 15c and 15d.

With respect to the inviscid configuration, the high-order boundary representation seems to play a smaller role. This effect is caused by the different nature of the no-slip boundary condition, which is solely a function of the position. In contrast to the flow tangency condition used for Euler equations, the no-slip boundary condition is continuous across each element, therefore it introduces less numerical error. Furthermore, the physical viscosity regularizes the solution field, damping the artificial oscillations. It is thus possible that the influence of the boundary representation increases for higher Reynolds numbers, also due to the development

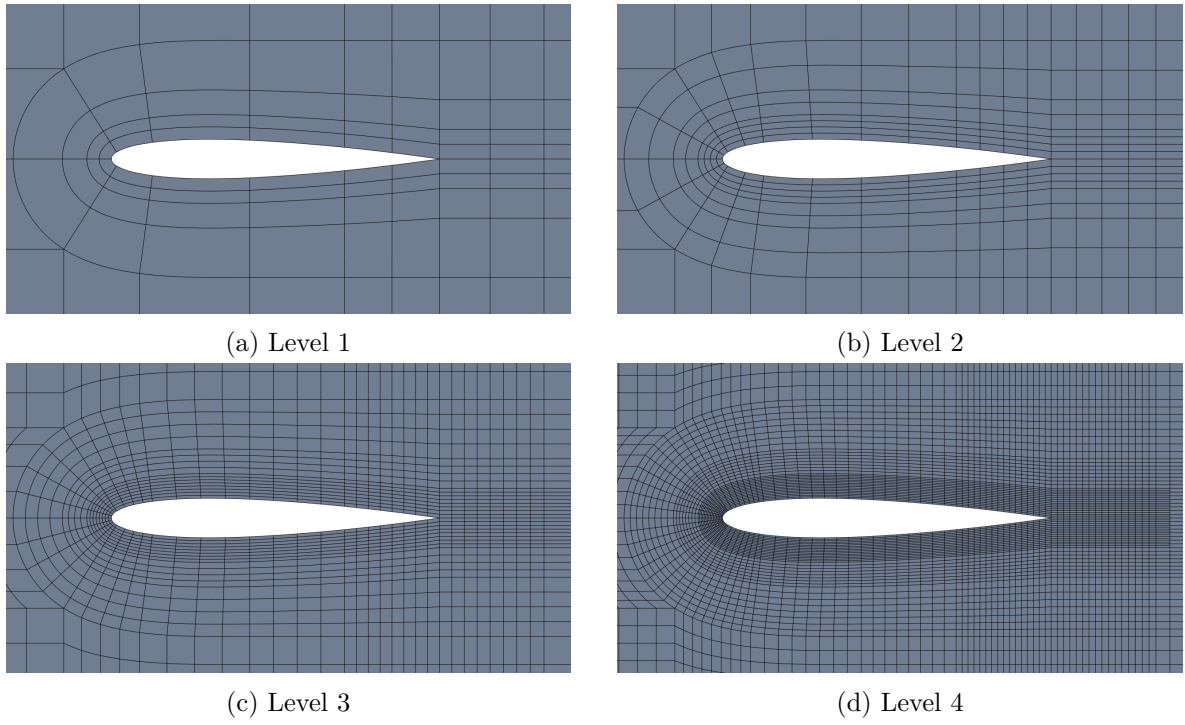


Figure 14: Meshes for the laminar test case

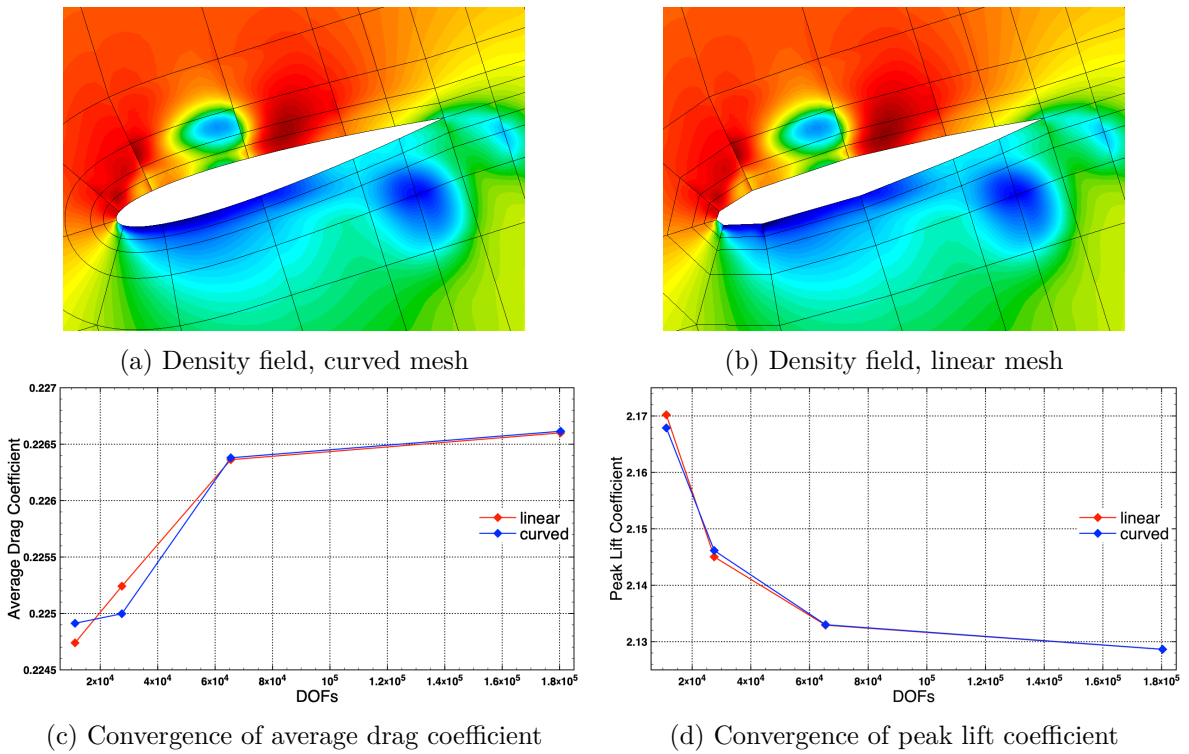


Figure 15: Influence of the geometry, laminar flow



of thin boundary layers and turbulent phenomena. However, such an investigation is beyond the scope of the present paper.

### 5.3 Pitching airfoil in transonic flow

In the third case study, we validate the shock capturing capability of the proposed scheme by investigating the transonic flow around a pitching NACA 0012 airfoil. The angle of attack evolves with the following law:

$$\alpha(t) = \alpha_0 + \Delta\alpha \sin(2\pi ft), \quad (51)$$

where  $\alpha_0 = 0.016^\circ$ ,  $\Delta\alpha = 2.51^\circ$  and the reduced frequency  $k = 0.0814$ . The airfoil oscillates about the quarter-chord and the mesh movement technique is the same used in the previous case study. The freestream Mach number is equal to 0.755, in order to create a transonic flow over the airfoil. An inviscid fluid is considered, therefore  $\mu = 0$ . However, due to the presence of the artificial viscosity, the Navier-Stokes equations are solved. A uniform flow field is used as a initial condition and a slip wall model is used for the airfoil surface. The results presented here are obtained using a quartic polynomial approximation with 3790 elements.

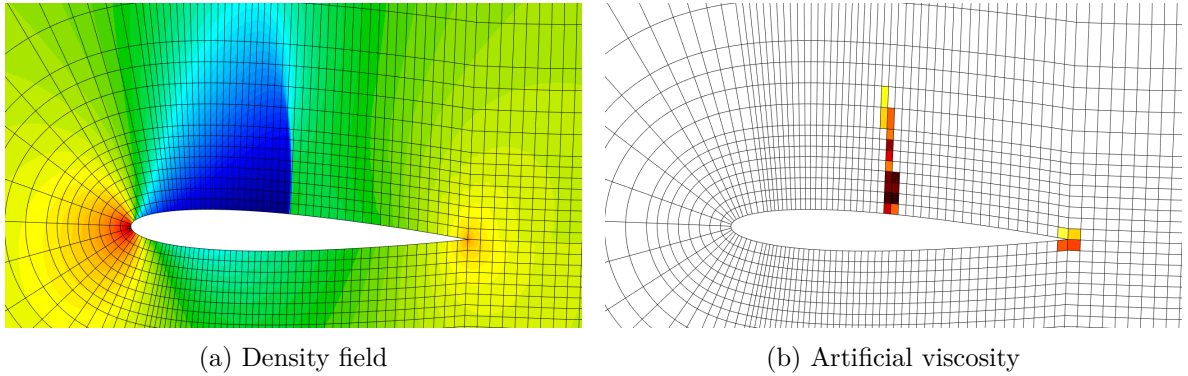


Figure 16: Transonic pitching airfoil, solution at  $\alpha = 2.25^\circ$ , descending phase

Transonic flows around pitching airfoils are a challenging benchmark for shock capturing algorithms, as the discontinuity position and intensity vary with the angle of attack. The density field and the artificial viscosity at  $\alpha = 2.25^\circ$  are reported in Fig. 16. The shock is well detected by the sensor, thus, the artificial viscosity is added only in the cells that require stabilization. It can also be observed that, due to a slight under-resolution, some artificial viscosity is added around the trailing edge.

In Fig. 17 we propose a comparison with the numerical results of Ren et al. (50) and the experimental data of Landon (51). The curve of the pressure coefficient  $C_p$ , presented in Fig. 17a, correlates very well with both the experimental and numerical references. Moreover, it is possible to appreciate the precision of the developed shock capturing technique. In Fig. 17b the evolution of the pitching moment coefficient over an oscillation cycle is presented. Even though a small discrepancy can be observed, the obtained curve is comparable to the numerical reference and it is in line with the majority of the experimental measurements. The deviations with respect to the wind tunnel data can be imputed to the impossibility to exactly reproduce the experimental conditions, characterized by a very high Reynolds number ( $5.5 \cdot 10^6$ ).

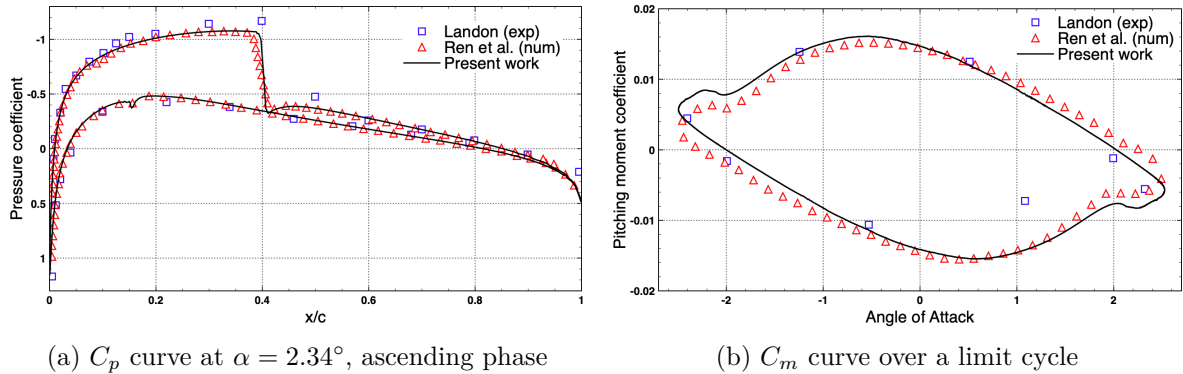


Figure 17: Transonic pitching airfoil, comparison with reference data

## 6 Conclusion

In this work, we investigated the use of NURBS-based grids in the context of moving computational domains. Different possible approaches to account for moving meshes in a DG formulation were analysed, in the perspective of implementing a scheme for moving rational Bézier elements. The proposed formulation combines an isogeometric approach for DG with the ALE description, leading to a unified representation of the geometry, the grid velocity and the solution variables.

Firstly, the accuracy of the method was verified on two problems with analytical solutions, exhibiting optimal convergence rates for rigidly moving or deforming grids. As expected, large grid deformations cause an increase of the error, due to the loss of the accuracy of the interpolation on distorted elements. The discrete geometry conservation law is satisfied when polynomial Bézier patches are considered. On the contrary, constant solutions are not exactly preserved when rational elements are used, due to the approximation introduced by the numerical quadrature.

We then considered three more demanding test-cases, involving inviscid and viscous flows around an oscillating cylinder and a pitching airfoil. The results confirmed the robustness of the proposed scheme in presence of high-order mesh deformations. Moreover, a faster convergence of the flow characteristics thanks to the use of curved grids was clearly established, in the context of moving bodies. It was also noted that inviscid flows are particularly sensitive to the lack of regularity in the geometry, due to the slip boundary condition at wall. Finally, we showed the robustness of the present methodology in presence of shocks, using a transonic benchmark problem.

Several prospects can be drawn, starting from these results. In particular, one could envisage the extension to physics-based movements, like in fluid-structure interaction problems. The possibility to define a unique high-order interface with different discretizations on each side would be appealing to account for the different characteristics of the fluid and structural models. Because of the capability to exactly represent circular shapes, the proposed approach would also be interesting to accurately compute fluxes at sliding interfaces, allowing the development of a fully conservative scheme for sliding grids.

## Code Repository

The developed methodology is implemented in the **Igloo** software suite, which has been employed to perform all the presented computations. The source code is available, under the GNU General Public Licence v3, at the following repository: <https://gitlab.inria.fr/igloo/igloo/-/wikis/home>.

## Acknowledgements

The authors are grateful to the OPAL infrastructure from Université Côte d’Azur for providing resources and support.

## References

- [1] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering* 194 (2005) 4135 – 4195.
- [2] G. Farin, *Curves and Surfaces for CAGD*, 5th ed., Morgan Kaufmann, 2002.
- [3] G. Xu, B. Mourrain, A. Galligo, R. Duvigneau, Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method, *J. Comput. Physics* 252 (2013).
- [4] S. Xia, X. Qian, Isogeometric analysis with Bézier tetrahedra, *Computer Methods in Applied Mechanics and Engineering* (2017) 782–816.
- [5] N. Jaxon, X. Qian, Isogeometric analysis on triangulations, *Computer Aided Design* (2014) 45–57.
- [6] L. Engvall, J. Evans, Isogeometric triangular Bernstein-Bézier discretizations: automatic mesh generation and geometrically exact finite-element analysis, *Computer Methods in Applied Mechanics and Engineering* (2016) 378–407.
- [7] Z. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H. Huynh, N. Kroll, G. May, P.-O. Persson, B. van Leer, M. Visbal, High-order CFD methods: current status and perspective, *International Journal for Numerical Methods in Fluids* 72 (2013) 811–845.
- [8] F. Bassi, S. Rebay, A High-Order Accurate Discontinuous Finite Element Method for the Numerical solution of the compressible navier–stokes equations, *Journal of Computational Physics* 131 (1997) 267 – 279.
- [9] I. Lomtev, R. Kirby, G. Karniadakis, A Discontinuous Galerkin ALE Method for Compressible Viscous Flows in Moving Domains, *Journal of Computational Physics* 155 (1999) 128 – 159.
- [10] J.-B. Chapelier, M. de la Llave Plata, F. Renac, E. Lamballais, Evaluation of a high-order discontinuous Galerkin method for the DNS of turbulent flows, *Computers & Fluids* 95 (2014) 210 – 226.

- [11] A. S. Silveira, R. C. Moura, A. F. C. Silva, M. A. Ortega, Higher-order surface treatment for discontinuous Galerkin methods with applications to aerodynamics, *International Journal for Numerical Methods in Fluids* 79 (2015) 323–342.
- [12] R. Costa, S. Clain, R. Loubère, G. J. Machado, High-order accurate finite volume scheme on curved boundaries for the two-dimensional steady-state convection-diffusion equation with Dirichlet condition, *Applied Mathematical Modelling* 54 (2018).
- [13] R. Sevilla, S. Fernandez-Mendez, A. Huerta, NURBS-Enhanced Finite Element Method for Euler equations, *Int. J. for Numerical Methods in Fluids* 57 (2008).
- [14] R. Abgrall, C. Dobrzynski, A. Froehly, An example of high order residual distribution scheme using non Lagrange elements : example of Bézier and NURBS., in: *WCCM 2010 - the 9th World Congress on Computational Mechanics and 4th Asian Pacific Congress on Computational Mechanics*, Sydney, Australia, 2010.
- [15] C. Michoski, J. Chan, L. Engvall, J. Evans, Foundations of the blended isogeometric discontinuous Galerkin (BIDG) method, *Computer Methods in Applied Mechanics and Engineering* 305 (2016) 658–681.
- [16] R. Duvigneau, Isogeometric analysis for compressible flows using a Discontinuous Galerkin method, *Computer Methods in Applied Mechanics and Engineering* 333 (2018) 443 – 461.
- [17] R. Duvigneau, CAD-consistent adaptive refinement using a NURBS-based discontinuous Galerkin method, *Int. J. for Numerical Methods in Fluids* (2020).
- [18] W. Anderson, V. Dobrev, T. Kolev, R. Rieben, V. Tomov, High-order multi-material ALE hydrodynamics, *SIAM J. Sci. Comput.* 40 (2018).
- [19] Y. Bazilevs, I. Akkerman, D. Benson, G. Scovazzi, M. Shashkov, Isogeometric analysis of Lagrangian hydrodynamics, *Journal of Computational Physics* 243 (2013) 224 – 243.
- [20] B. Boutin, E. Deriaz, P. Hoch, P. Navaro, Extension of ALE methodology to unstructured conical meshes, *ESAIM Proceedings* 32 (2011).
- [21] Y. Bazilevs, V. Calo, T. Hughes, Y. Zhang, Isogeometric fluid-structure interaction: theory, algorithms, and computations, *Comput. Mech.* 43 (2008) 3 – 37.
- [22] Y. Bazilevs, K. Takizawa, T. Tezduyar, *Computational Fluid–Structure Interaction: Methods and Applications*, John Wiley & Sons, 2013.
- [23] W. Boscheri, M. Dumbser, Arbitrary-Lagrangian–Eulerian Discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes, *Journal of Computational Physics* 346 (2017) 449 – 479.
- [24] R. Loubère, P.-H. Maire, M. Shashkov, J. Breil, S. Galera, ReALE: A reconnection-based arbitrary-Lagrangian–Eulerian method, *Journal of Computational Physics* 229 (2010) 4724 – 4761.
- [25] V.-T. Nguyen, An arbitrary Lagrangian-Eulerian discontinuous Galerkin method for simulations of flows over variable geometries, *Journal of Fluids and Structures* 26 (2010) 312, 329.

- [26] P.-O. Persson, J. Bonet, J. Peraire, Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains, *Computer Methods in Applied Mechanics and Engineering* 198 (2009) 1585 – 1595.
- [27] F. Vilar, P.-H. Maire, R. Abgrall, A discontinuous Galerkin discretization for solving the two-dimensional gas dynamics equations written under total Lagrangian formulation on general unstructured grids, *Journal of Computational Physics* 276 (2014) 188 – 234.
- [28] J. van der Vegt, H. van der Ven, Space–Time Discontinuous Galerkin Finite Element Method with Dynamic Grid Motion for Inviscid Compressible Flows: I. General Formulation, *Journal of Computational Physics* 182 (2002) 546 – 585.
- [29] E. Gaburro, W. Boscheri, S. Chiochetti, C. Klingenberg, V. Springel, M. Dumbser, High order direct Arbitrary-Lagrangian-Eulerian schemes on moving Voronoi meshes with topology changes, *Journal of Computational Physics* 407 (2020) 109 – 167.
- [30] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferran, *Arbitrary Lagrangian–Eulerian Methods*, 2004.
- [31] C. S. Venkatasubban, A new finite element formulation for ALE (Arbitrary Lagrangian Eulerian) compressible fluid mechanics, *International Journal of Engineering Science* 33 (1995) 1743 – 1762.
- [32] H. Guillard, C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, *Computer Methods in Applied Mechanics and Engineering* 190 (2000) 1467 – 1482.
- [33] J. S. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, 1st ed., Springer, 2007.
- [34] M. R. Visbal, D. V. Gaitonde, On the Use of Higher-Order Finite-Difference Schemes on Curvilinear and Deforming Meshes, *Journal of Computational Physics* 181 (2002) 155 – 185.
- [35] L. Piegl, W. Tiller, *The NURBS Book*, second ed., Springer-Verlag, New York, NY, USA, 1996.
- [36] B. Cockburn, C.-W. Shu, The Local Discontinuous Galerkin Method for Time-Dependent Convection-Diffusion Systems, *SIAM Journal on Numerical Analysis* 35 (1998) 2440–2463.
- [37] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong Stability-Preserving High-Order Time Discretization Methods, *SIAM Review* 43 (2001) 89, 112.
- [38] R. Löhner, C. Yang, Improved ALE mesh velocities for moving bodies, *Communications in Numerical Methods in Engineering* 12 (1996) 599–608.
- [39] P.-O. Persson, J. Peraire, *Sub-Cell Shock Capturing for Discontinuous Galerkin Methods*, 2006.
- [40] G. E. Barter, D. L. Darmofal, Shock capturing with PDE-based artificial viscosity for DGFEM: Part I. Formulation, *Journal of Computational Physics* 229 (2010) 1810 – 1827.

- [41] J. Qiu, C.-W. Shu, Runge-Kutta Discontinuous Galerkin Method Using WENO Limiters, *SIAM Journal on Scientific Computing* 26 (2005) 907–929.
- [42] M. Dumbser, R. Loubère, A simple robust and accurate a posteriori sub-cell finite volume limiter for the discontinuous Galerkin method on unstructured meshes, *Journal of Computational Physics* 319 (2016) 163 – 199.
- [43] Y. Bazilevs, L. Beirão Da Veiga, J. A. Cottrell, T. J. R. Hughes, G. Sangalli, Iso-geometric Analysis: approximation, stability and error estimates for h-refined meshes, *Mathematical Models and Methods in Applied Sciences* 16 (2006) 1031–1090.
- [44] A. Harten, P. D. Lax, B. van Leer, On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws, *SIAM Review* 25 (1983) 35–61.
- [45] C. Williamson, A. Roshko, Vortex formation in the wake of an oscillating cylinder, *Journal of Fluids and Structures* 2 (1988) 355 – 381.
- [46] H. M. Blackburn, R. D. Henderson, A study of two-dimensional flow past an oscillating cylinder, *Journal of Fluid Mechanics* 385 (1999) 255–286.
- [47] G. Mengaldo, D. D. Grazia, F. Witherden, A. Farrington, P. Vincent, S. Sherwin, J. Peiro, *A Guide to the Implementation of Boundary Conditions in Compact High-Order Methods for Compressible Aerodynamics*, 2014.
- [48] X.-Y. Lu, C. Dalton, Calculation of the timing of vortex formation from an oscillating cylinder, *Journal of Fluids and Structures* 10 (1996) 527 – 541.
- [49] A. Roshko, *On the Development of Turbulent Wakes from Vortex Streets*, National Advisory Committee for Aeronautics (NACA), 1954.
- [50] X. Ren, K. Xu, W. Shyy, A multi-dimensional high-order DG-ALE method based on gas-kinetic theory with application to oscillating bodies, *Journal of Computational Physics* 316 (2016) 700 – 720.
- [51] R. H. Landon, *NACA 0012 oscillatory and transient pitching*, Advisory Group for Aerospace Research and Development (AGARD), 1982.