mathematics

MDPI

Article

Comparative Analysis of NURBS and Finite Element Method in Computational Fluid Dynamics Applications: Case Study on NACA 2412 Airfoil Aerodynamics

Sohaib Guendaoui [1], Lahcen El Ouadefli [1], Abdeslam El Akkad [1,2], Ahmed Elkhalfi [1], Sorin Vlase [3] and Maria Luminița Scutaru [3,*]

[1] Faculty of Sciences and Techniques, Sidi Mohammed Ben Abdellah University, B.P. 2202 Route Imouzzer, Fes 30003, Morocco; sohaib.guendaoui@usmba.ac.ma (S.G.); lahcen.elouadefli@usmba.ac.ma (L.E.O.); elakkadabdeslam1@gmail.com (A.E.A.); ahmed.elkhalfi@usmba.ac.ma (A.E.)

[2] Département de Mathématiques, Centre Régional des Métiers d'Education et de Formation de Fès Meknès (CRMEF Fès-Meknès), Rue de Koweit 49, Ville Nouvelle, Fez 30050, Morocco

[3] Department of Mechanical Engineering, Faculty of Mechanical Engineering, Transylvania University of Brasov, B-dul Eroilor 29, 500036 Brasov, Romania; svlase@unitbv.ro

* Correspondence: lscutaru@unitbv.ro

Abstract: In this research, an attempt was made to employ the Non-Uniform Rational B-Splines (NURBS) method for a challenging computational fluid dynamics (CFD) problem of aerodynamics around NACA 2412 airfoils. The comparison was carried out thoroughly by using the same boundary conditions and geometry, comparing NURBS to standard FEM implementations. Our study was interested in demonstrating the foreseeable functionalities of NURBS for solving complex CFD problems and conducting a comparative effectiveness performance evaluation between them with traditional FEM methodologies.

Keywords: B-spline; NURBS; CFD; aerodynamics; NACA2412; FEM

MSC: 76D06; 60H30; 76M35; 94B27; 39B99

Citation: Guendaoui, S.; Ouadefli, L.E.; El Akkad, A.; Elkhalfi, A.; Vlase, S.; Scutaru, M.L. Comparative Analysis of NURBS and Finite Element Method in Computational Fluid Dynamics Applications: Case Study on NACA 2412 Airfoil Aerodynamics. *Mathematics* **2024**, *12*, 3211. https://doi.org/10.3390/math12203211

Academic Editors: Gang Sun and Liyue Wang

Received: 11 September 2024
Revised: 7 October 2024
Accepted: 10 October 2024
Published: 14 October 2024

1. Introduction

In the world of aerodynamics, airfoils are well worth investigating. Airfoils, which are profiles designed to generate lift in aircraft wings, are exposed to nonlinear fluid dynamics with direct repercussions on their functioning. The most important among these are the lift and drag coefficients, which characterize how much lift or drag is produced by the airfoil; hence, they determine its aerodynamic performance [1].

In the field of linear elasticity and stress distribution in structures, including the Navier–Lamé equations [2], which govern the deformation and stress–strain relationships in elastic materials, computational fluid dynamics (CFD) has afforded substantial contributions in modeling and analyzing fluidic dynamics with respect to flow velocity distribution patterns, pressure gradients, and turbulent viscosity. CFD, combined with the stress analysis from the Navier–Lamé equations, plays a crucial role in understanding the stress distribution in pipelines [3], where internal pressures and fluid flow cause complex stress patterns that need to be evaluated for structural integrity. CFD also helps calculate the lift and drag coefficients for different flow conditions, which can be used to optimize designs, including airfoils and piping systems.

CFD analysis uses the Navier–Stokes equations, which characterize the movement of fluid substances, as a basis for its technology. These equations are used with respect to the characteristics of fluid [4], in particular viscosity and density, and they govern how momentum is balanced within a given fluid. Nonetheless, calculating these equations for complex geometries and flow conditions can be time-consuming.

Here is where Non-Uniform Rational B-Splines (NURBS) and isogeometric analysis (IGA) are useful. While NURBS are a general mathematical representation widely used in computer graphics and CAD systems, they provide a rich solution for airfoil geometry definition. The use of NURBS together with CFD analyses enables more precise mapping airfoil representation and calculation process optimization.

Isogeometric analysis was pioneered by Cottrell and al. [5] as an effort to use the same mathematical representations (NURBS) employed in CAD systems for analysis, thus realizing a fully integrated simulation and design process.

With high-order CAD-consistent geometry used directly in modeling, mesh optimization and refinement processes can be simplified.

This high regularity of NURBS functions makes this technique ideally suited to elliptic and parabolic problems [6], such as those in structural mechanics [7]; however, applications in CFD are currently quite rare.

This paper attempts to present the Non-Uniform Rational B-Splines methodology and describe the governing equations used in computational fluid dynamics (CFD). Then, a discussion is provided on aerodynamic terminologies, which primarily involve velocity and pressure related to aerospace.

For airfoil analysis using the NURBS methodology, a complex B-spline Matlab code was developed, and the NURBS methodology was intended to be applied for CFD analysis on a standardized airfoil model (NACA 2412). This simulation is an important step in understanding the complex behaviors of moving objects in a variety of airfoil environments.

For the last part of our study, these results are compared to those obtained using a finite element method (FEM) simulation. This work is intended to compare the different methods and the biggest public impact that it can have on concealed applications, by using the NACA 2412 airfoil as the model of study.

Having examined the a posteriori error curve and detailed pressure/velocity vector fields, our analysis substantiates that the use of NURBS gives superior results concerning both accuracy and efforts.

The posteriori error results provide information on the convergence behavior of NURBS vs. FEM simulations, as shown in [8], revealing that they converge with a moderate rate, which is to be expected for this problem given the complex nature of airflow around airfoil profiles like NACA 2412. This critical analysis is essential in determining the usability of the NURBS method and procuring accurate results from it.

Our results, therefore, demonstrate an increased level of accuracy that can be achieved when using NURBS simulations, and these simulations offer a more efficient approach for obtaining these results with much lower computational time. This performance is essential given the use cases in aerospace engineering where fast response time with exact simulation results is too important to miss.

This paper is organized into multiple sections. In Section 2, basic introductions regarding Non-Uniform Rational B-Splines (NURBS) are given. The governing equations are introduced in Section 3. The process of discretizing the Navier–Stokes equations using the NURBS IGA is presented in Section 4. Section 5 shows the numerical resolution process and the comparative results, and finally, a conclusion is provided in Section 6.

2. Governing Preliminaries of Non-Uniform B-Spline Functions

Non-Uniform B-Spline (NURBS) functions have long been the industry standard for CAD (Computer-Aided Design) systems. Recently, their prominence has significantly increased within the Computational Mechanics community, primarily because of the contributions by Hughes et al. [9]. In their research, NURBS were utilized as shape functions within the finite element method (FEM), leading to the development of a specialized version known as isogeometric analysis. This innovative approach has since been applied to a wide range of problems.

Let $a = (a_1, a_2, ..., a_{n+p+1})$ be a set of non-decreasing numbers representing coordinates in the parametric space, where $i = 1, ..., n + p + 1$. A knot vector is termed uniform

when all its knot values are evenly spaced; otherwise, it is considered non-uniform. Additionally, a knot vector is described as open if its first and last knots are repeated p + 1 times.

A B-spline basis function of order 0 is then defined as follows:

$$B_i^0(x) := \begin{cases} 1, & \text{if } a_i \le x \le a_{i+1} \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

B-spline basis functions of order p > 1 are defined recursively as follows (Figure 1):

$$B_i^d(x) := \frac{x - a_i}{a_{i+d} - a_i} B_i^{d-1}(x) + \frac{a_{i+d+1} - x}{a_{i+d+1} - a_{i+1}} B_{i+1}^{d-1}(x), \quad d = 1, \ldots; p. \tag{2}$$

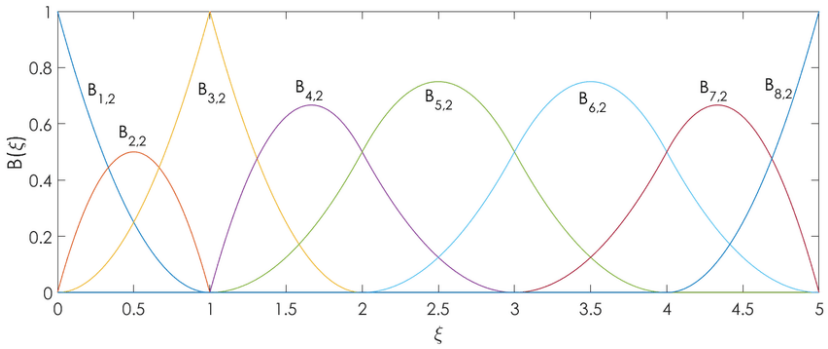


Figure 1. B-spline basis functions at knot vector Ξ = {0,0,0,1,1,2,3,4,5,5,5}.

NURBS basis functions are established by integrating a set of positive weights that shape the B-spline $\omega_i, i = 1, ..., n$.

$$R_i^p(x) := \frac{\omega_i B_i^p(x)}{\sum_{j=1}^{n} \omega_j B_j^d(x)} \tag{3}$$

NURBS basis functions C^{p-1} preserve continuity when there are no repeated internal knots.

For a repeated knot of order k, the basis C^{p-k} maintains continuity at that knot. A NURBS curve is characterized by the integration of NURBS basis functions of order p and n control points. Therefore, it can be expressed as $\underline{C}(x) = \sum_{i}^{n} R_i^p \underline{P_i}(x)$, and the control points determine the shape and configuration of the polygon:

Bivariate NURBS (2D) are defined as follows:

$$N_{i,j}^{p,q}(x,y) = \frac{B_i^p(x) B_j^p(y) \omega_{ij}}{\sum_k^n \sum_j^m B_l^p(x) B_j^q(y) \omega_{kl}} \tag{4}$$

NURBS curves can be expanded into NURBS surfaces and volumes by utilizing the tensor products of B-spline basis functions. The characteristics of NURBS surfaces are as follows:

$$S : [0,1]^2 \to \mathbb{R}^2 \tag{5}$$

$$S(\xi_1, \xi_2), = (x_1(\xi_1, \xi_2), x_2(\xi_1, \xi_2)), \tag{6}$$

$$S(\xi_1, \xi_2), = \sum_{i}^{n} \sum_{j}^{m} N_{i,j}^{p,q}(\xi_1, \xi_2) P_{ij}. \tag{7}$$

These are based on two coordinates (ξ_1, ξ_2).

The B-spline basis functions (Figure 2) are defined for polynomial orders p and q, with the control net consisting of n·m control points. Unlike the control points P_{ij}, the associated weights ω_{ij} used in defining the NURBS basis do not adhere to a tensor product structure

and can be selected freely. NURBS volumes can be formulated in a similar manner. The derivatives of NURBS parametrizations with respect to the parameter coordinates can be readily computed for any order of the NURBS basis functions (Figure 3).

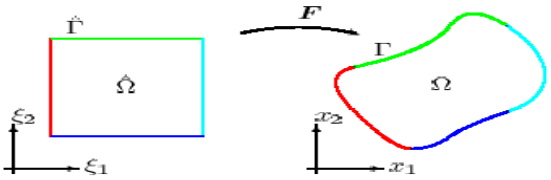


Figure 2. Physical domain on the right and reference domain on the left.

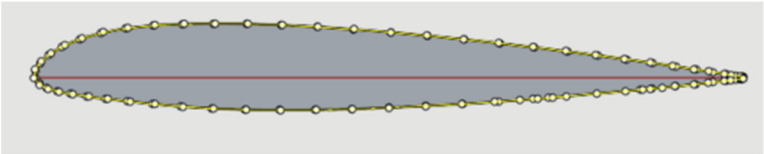


Figure 3. A representation of a blade's cross-section using B-spline modeling for the control point coordinates.

The NURBS space in the parametric domain $[0,1] \times [0,1]$ is $\hat{\mathcal{N}}^{p,q} := \text{span}\left\{N_{i,j}^{p,q}\right\}_{i,j=1'}^{n,m}$ and for the space $\mathcal{R}$ on Ω, it represents a push-forward of the space $\hat{\mathcal{N}}$:

$$\mathcal{R}^{p,q} := \text{span}\left\{N_{i,j}^{p,q} \circ s^{-1} \mid i = 1, \ldots, n-1; j = 1, \ldots, m-1\right\} \tag{8}$$

To implement a complex geometry, such as an airfoil cross-section, using NURBS functions, it is essential to divide the domain into multiple sections. This approach is referred to as a multi-patch domain.

Assume that the physical airfoil domain Ω is formed by the union of $N_{ptc} = 4$ patches (see Figure 4), in such a way that

$$\overline{\Omega} = \bigcup_{i=1}^{N_{ptc}} \overline{\Omega}_i, \quad \text{with } \Omega_i \cap \Omega_j = \emptyset \text{ for } i \neq j \tag{9}$$

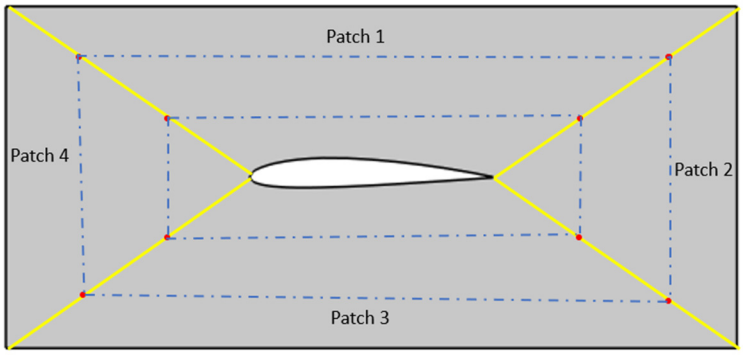


Figure 4. Multi-patch domain of airfoil NACA2412.

Each patch is defined as the image of the same parametric domain under various parametrizations, that is

$$\Omega_i = \mathbf{F}^{(i)}(\hat{\Omega}). \tag{10}$$

Including multi-patch domains, such a methodology is similar to former methodologies enforcing C_0 continuity strongly based on conforming patch alignment. The residual error at interfaces is particularly important in this study as the robust continuity of the

yellow interfaces between patches is desired, as shown in the figure below which contains three control points for each one, which are associated with basis functions that align precisely on the interfaces.

3. Governing Equations

The two-dimensional case of fluid flow around an airfoil can be described by the Navier–Stokes equations that govern both pressure gradient and viscous effects dictating laminar flow behavior. These can be written as follows:

$$\begin{cases} -\nu \operatorname{div}(\nabla u) + u.\nabla u + \nabla p = f & \text{in} \quad \Omega, \\ \operatorname{div}(u) = 0 & \text{in} \quad \Omega, \end{cases} \tag{11}$$

By the $C_{\alpha,\beta}$ boundary condition,

$$\alpha u + \beta(\nabla u - pI)n = t \ \text{ on } \ \Gamma := \partial\Omega. \tag{12}$$

It is additionally hypothesized that Ω has a polygonal boundary $\Gamma{:=}\partial\Omega$, so n is the generic outward-pointing normal, where

$u : \Omega \to \mathbb{R}^n, n = 2 \text{ or } 3,$ and $p : \Omega \to \mathbb{R}$ represent the unknown velocity and pressure, respectively. The symbol ν denotes the kinematic viscosity.

To derive the weak form of the Navier–Stokes Equations (11) and (12), which describe the laminar flow behavior over an airfoil by considering both the pressure gradient and viscous effects, the equations are multiplied by the test functions and then integrated over the domain Ω.

Our mathematical model is the Navier–Stokes system, accompanied by a specified boundary condition $C_{\alpha,\beta}$. ∇ is the gradient, $\nabla.$ is the divergence, and ∇^2 is the Laplacian operator. f, t, α, and β are polynomials such that f is defined in Ω, and α and β are nonzero defined on $\partial\Omega$ and verify the following:

There are two strictly positive constants a_1 and b_1 such that

$$a_1 \leq \frac{\alpha(x)}{\beta(y)} \leq b_1 \text{ for all } x \in \Gamma. \tag{13}$$

Therefore, α is called the Dirichlet coefficient, and β is the Neumann coefficient.

Our objective is to find a weak solution to the Navier–Stokes equations in the following spaces:

$$X = \left[H^1(\Omega)\right]^n, \tag{14}$$

$$Q = \left\{ q \in L^2(\Omega) : \int_\Omega q dx = 0 \right\}, \tag{15}$$

$$X_0 := \left[H_0^1(\Omega)\right]^n, \tag{16}$$

$$M := L_0^2(\Omega) = Q, \tag{17}$$

$$V := X_0 \times M. \tag{18}$$

An additional condition on Q is necessary to ensure the uniqueness of the pressure. To address the issue of non-unique pressure solutions due to a constant term, the variational formulation is derived by taking the inner product of the momentum equation with a test function $v \in X_0$ and by taking the inner product of the continuity equation with a test function $q \in Q$.

The product of the momentum and mass balance equations in (11) with test functions $v \in X_0$ and $q \in M$, respectively, is applied.

Applying integration by parts results in the weak formulation. Utilizing Green's theorem and the homogeneous Dirichlet boundary condition, it can be expressed as follows:

For all $v \in X_0$,

$$-\nu \int_\Omega \mathrm{div}(\nabla u).v d\delta + \int_\Omega (u.\nabla u).v d\delta + \int_\Omega \nabla p.v d\delta = \int_\Omega f.v d\delta, \tag{19}$$

$$\int_\Omega \mathrm{div}(u) q = 0, \quad \forall q \in Q, \tag{20}$$

For all cases, it is obtained that $v \in X_0$:

$$\begin{cases} -\nu \int_\Omega \nabla u : \nabla v d\delta + \int_\Gamma \frac{\alpha}{\beta}.u.v d\Gamma + \int_\Omega (u.\nabla)u.v d\delta - \int_\Omega p \mathrm{div}(v) d\delta = \int_\Omega f.v d\delta + \int_\Gamma \frac{1}{\beta}.t.v d\Gamma, \\ \qquad\qquad \forall q \in Q, \\ \int_\Omega \mathrm{div}(u) q d\delta = 0, \end{cases} \tag{21}$$

The trilinear, bilinear, linear, and semilinear forms will now be introduced, starting with the bilinear forms:

$$A : X \times X \to R, \quad A(u,v) = \nu(\nabla u, \nabla v) + \int_\Gamma \frac{\alpha}{\beta}.u.v d\Gamma, \tag{22}$$

$$B : X \times Q \to R, \quad B(u,q) = (\mathrm{div}(u), q), \tag{23}$$

$$L : Q \to R, \quad L(v) = \int_\Omega f.v d\delta + \int_\Gamma \frac{1}{\beta}.t.v d\Gamma. \tag{24}$$

Then, the non- linear forms are as follows:

$$N : X \times X \times X \to R, N(w,u,v) = ((w.\nabla)u, v). \tag{25}$$

The following is then obtained.
Find $(u,p) \in X \times M$ such that

$$\begin{cases} A(u,v) + N(u,u,v) - B(v,p) = L(v) & \forall\, v \in X_0, \\ B(u,q) = 0, & \forall q \in Q. \end{cases} \tag{26}$$

A similar trajectory as that in [10] is followed by introducing the extension $G \in X$ of g which verifies $(\varepsilon G) = 0$. From this, $u = w + G$, where $w \in X_0$.

Therefore, problems (19) and (20) allow us to conclude that w is the solution to the following problem:

$$\begin{cases} A(w,v) + N(w+G, w+G, v) - B(v,p) = l(v) - A(G,v), & \forall\, v \in X_0, \\ B(w,q) = 0, & \forall q \in Q. \end{cases} \tag{27}$$

4. Isogeometric Discretization

In isogeometric analysis, the computational domain is directly obtained from a CAD model of the embedded structure [11]. Although this method reduces some of the effort involved in volume mesh generation, it does not eliminate this step. Based on [12], the key features of all mesh outputs are highlighted. The mesh (i.e., the tessellation T^h) is isotopically graded away from aerodynamic surfaces and possesses a tensorial structure. Mesh lines are extended normal to the wall to maintain mesh quality in the boundary layer. This approach, as noted in [13], is one of the advantages of the isogeometric method.

The approximation spaces are derived from the resulting geometric description of the volume. Since airfoil sections are typically represented by polynomials, B-spline discretization is also considered here. A brief introduction to this technology is provided, with further details available in [14] for an introduction to isogeometric analysis and [15] for details on their implementation.

The initial form of isogeometric analysis (IGA) employs a Galerkin formulation within a finite-dimensional subspace of X_0; this subspace is formed using the same set of B-spline/NURBS basis functions that describe the geometry, aiming to estimate the solution

to Equation (27). Consequently, the functional spaces $X_h = \left(\mathcal{R}^{p+1,p+1} \times \mathcal{R}^{p+1,p+1}\right) \cap X_0$, and $M_h = \mathcal{R}^{p,p} \cap M$, where the spaces $\mathcal{R}^{p,p}$ are pre-defined in Equation (8). Withing this framework, X_h and M_h are defined using the multivariate NURBS functions discussed in Section 2.

The IGA formulation can be expressed as follows. Find $(u_h, p_h) \in X_h \times M_h$ such that

$$\begin{cases} A\left(w_h^{j+1}, v_h\right) + N\left(w_h^j + G, w_h^{j+1}, v_h\right) - B\left(v_h, p_h^{j+1}\right) = L(v_h) - A(G, v_h) - 6N\left(w_h^j + G, G, v_h\right), \\ B\left(w_h^{j+1}, q_h\right) = 0, \qquad \forall\, v_h \in X_h,\ \forall q_h \in M_h, \end{cases} \tag{28}$$

At iteration j, the goal is to find the velocity $u_h^j : [0,1]^2 \to \mathbb{R}^2$ and pressure $p_h^j : [0,1]^2 \to \mathbb{R}$ as linear combinations of the basis functions mentioned above. While NURBS are mainly used for representing geometry, their effectiveness will be evaluated by employing them to approximate state variables, specifically the velocity u and the pressure p.

$$u_{hk}^j(\xi_1, \xi_2) = (w_{hk}^j + G)(\xi_1, \xi_2) = \sum_i^n N_i^{p,q}(\xi_1, \xi_2) u_{ik}^j, \tag{29}$$

In Equation (29), (k = 1, 2) refers to the two components of the velocity field.

$$p_h^j(\xi_1, \xi_2) = \sum_i^m N_i^{r,s}(\xi_1, \xi_2) p_{h_i}^j, \tag{30}$$

In this case, n and m in Equations (29) and (30) denote the number of basis functions for velocity and pressure, respectively. Correspondingly, u_{ik}^j and p_{hi}^j are the unknown control variables of the velocity and pressure at iteration j that need to be determined.

The velocity and pressure fields in Equation (28) are represented in parameter space, whereas those governing Equation (11) are defined in physical space.

To compute the fields in physical space, the inverse of the geometric parameterizations function defined in Equation (7) is used for the pressure: $\Omega \to R$ on the physical domain which is computed as $p \circ s^{-1}$ and the velocity $u : \Omega \to \mathbb{R}^2$ on the physical domain as $u \circ s^{-1}$.

The Jacobian matrix of the geometry parameterization is denoted by J and is defined as follows:

$$J = \begin{bmatrix} \frac{\partial x_1}{\partial \xi_1} & \frac{\partial x_1}{\partial \xi_2} \\ \frac{\partial x_2}{\partial \xi_1} & \frac{\partial x_2}{\partial \xi_2} \end{bmatrix}, \tag{31}$$

The pressure gradient in parameter space is as follows:

$$\hat{\nabla} p = J \nabla p = \begin{pmatrix} \left(\frac{\partial p}{\partial \xi_1}\right) \\ \left(\frac{\partial p}{\partial \xi_2}\right) \end{pmatrix} \tag{32}$$

$\nabla p = \left(\begin{smallmatrix} \left(\frac{\partial p}{\partial x_1}\right) \\ \left(\frac{\partial p}{\partial x_2}\right) \end{smallmatrix}\right)$ represents the pressure gradient in the physical domain.

Similarly, the velocity gradient in parameter space is defined. Since B-splines have limited support by construction, only a few fundamental velocity functions have support on Γ.

By organizing the functions according to their support, n_{ext} is the number of basis functions with a support intersecting the boundary Γ, of which $n\_{int} = n - n\_{ext}$ is the number of basis functions having a support inside the domain. So, Formula (29) would be in the following form:

$$u_{hk}^j(\xi_1, \xi_2) = \sum_{i=1}^{n_{int}} N_i^{p,q}(\xi_1, \xi_2) u_{ik}^j + \sum_{i=n_{int}+1}^{n} N_i^{p,q}(\xi_1, \xi_2) u_{ik}^j. \tag{33}$$

The strong form is applied only to n_{ext} values such that $u^j_{hk} = t^j_{hk}$ by setting the velocity control variables u^j_{ki}, so the total in Equation (28) is the value $t^j = \left(t^j_{h1}, t^j_{h2}\right)$ in (12). If t^j at iteration j lies in the functional space by $N^{p,q}_i$, the conditions are met precisely but only satisfied in the least-squares sense. For pressure, only the pressure gradient is present in Equation (11). Since no pressure control variables need to be fixed, $m = n_{int}$.

The approximation of the velocity and pressure fields (29) and (30) in the iterative approximate form (28) of the governing equations is implemented by following the approach of decomposing u into components with support on the fixed boundary and components without boundaries, as illustrated in Equation (33). This decomposition allows us to interchange summation and integration, facilitating the rearrangement of terms. Consequently, unknown terms can be isolated on the left side of System (28), while known terms are placed on the right side. By rearranging these terms, the unknown terms are obtained on the left side of system (33) and the known terms on the right side. System (28) can then be transformed into a matrix problem, which will be solved using appropriate MATLAB code. At iteration J, the matrix system is as follows:

$$\begin{bmatrix} A_1 + N_1^{j+1} & 0 & -B_1^T \\ 0 & A_2 + N_2^{j+1} & -B_2^T \\ B_1 & B_2 & 0 \end{bmatrix} \begin{bmatrix} w_1^{j+1} \\ w_2^{j+1} \\ p^{j+1} \end{bmatrix} = \begin{bmatrix} f_1^j \\ f_2^j \\ 0 \end{bmatrix} + \frac{1}{\beta} \begin{bmatrix} t_1^j \\ t_2^j \\ 0 \end{bmatrix} - \begin{bmatrix} A_1 + N_1^j & 0 \\ 0 & A_2 + N_2^j \\ B_1 & B_2 \end{bmatrix} \begin{bmatrix} w_1^j \\ w_2^j \end{bmatrix}, \tag{34}$$

With

$$(A_k)_{st} = \int_0^1 \int \nu \left(\varepsilon J^{-T} \nabla N^{p,q}_{ks} . J^{-T} \nabla N^{p,q}_{kt} \det(J) \right) d\xi_1 d\xi_1 + \int_\Gamma \frac{\alpha}{\beta} . \left(N^{p,q}_{ks} . N^{p,q}_{kt} \det(J) \right) d\Gamma, \tag{35}$$

For all iterations $j = 1, 2 \ldots$

$$\left(N^j_k\right)_{st} = \int_0^1 \int \left(w^{j-1}_{kh} + G^{j-1}_{kh} \right) \frac{\partial S}{\partial \xi_k} \frac{\partial N^{p,q}_{ks}}{\partial x_k} N^{p,q}_{kt}(S) \det(J) d\xi_1 d\xi_1, \tag{36}$$

$$(B_k)_{nm} = \int_0^1 \int \frac{\partial S}{\partial \xi_k} \frac{\partial N^{p,q}_{kn}}{\partial x_k} N^{r,s}_{km}(S) \det(J) d\xi_1 d\xi_1, \tag{37}$$

$$(f_k)_m = \int_0^1 \int f_k N^{p,q}_{km}(S) \det(J) d\xi_1 d\xi_1, \tag{38}$$

$$(t_k)_m = \int_\Gamma \frac{1}{\beta} . t_k . N^{p,q}_{km} \det(J) d\Gamma. \tag{39}$$

The integrals in Equations (35)–(39) are evaluated using Gaussian quadrature [15]. Following this, the performance of the method will be assessed by comparing the approximate solutions for velocity and pressure with those obtained using the finite element method (FEM).

5. Numerical Results and Discussions

Before integrating Non-Uniform Rational B-Splines (NURBS) within computational fluid dynamics (CFD) to enhance the design and aerodynamic performance of the NACA 2412 airfoil, the primary objective of this research was to evaluate the efficacy of NURBS in improving the precision and flexibility of airfoil geometry representation, thereby optimizing aerodynamic characteristics in this section, and the meshing process used in our analysis is detailed. The NACA 2412 airfoil domain is discretized using NURBS, which are highly effective for creating smooth and accurate representations of complex geometries. Regarding the NURBS meshing setup, the domain is divided into four patches, as shown in

Figure 4, each with a polynomial degree of 2 in the u and v directions, ensuring consistent meshing across all boundaries, as shown in Figure 5.

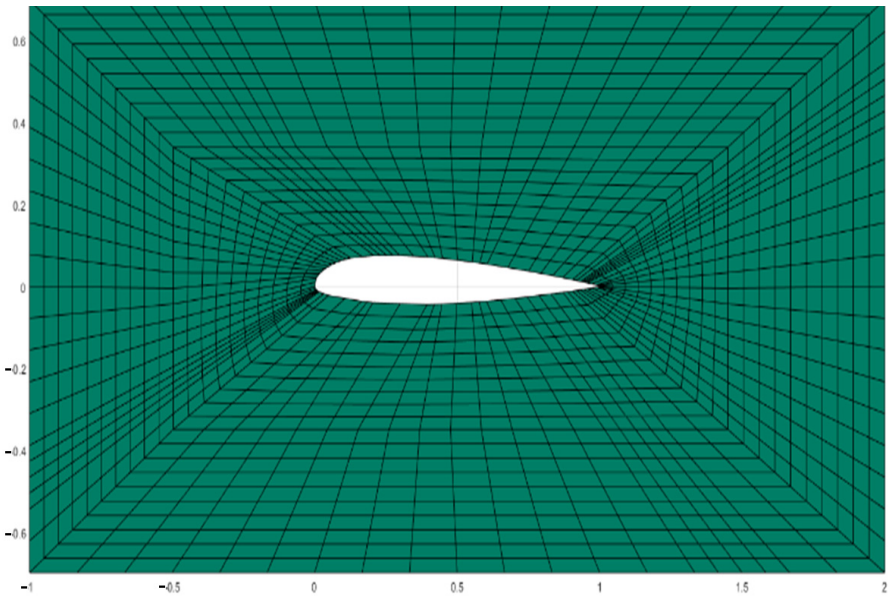


Figure 5. IGA mesh generation for airfoil NACA 2412 domain.

For the numerical integration within this meshed domain, a Gaussian quadrature rule is employed. Specifically, five quadrature points are used in each direction, providing the necessary precision for the subsequent calculations.

On the subject of FEM meshing, as previously stated, the geometry of the problem is a NACA 2412 airfoil with a domain 0.7 × 2 m, and this configuration is commonly employed in the study of aerodynamic phenomena surrounding airfoils (Figure 6). A commercial software is utilized for the generation of the CFD-FEM mesh, which automatically refines the mesh at the airfoil's surface. A quadratic mesh (prism layer mesh) [16] is used at the airfoil's boundary layer (see Figure 7) where a lot of important physics happen, so it is important to have a fine mesh here to accurately capture the velocity gradient near the wall of the airfoil {f. For the remainder of the domain, a tetrahedral mesh is employed.

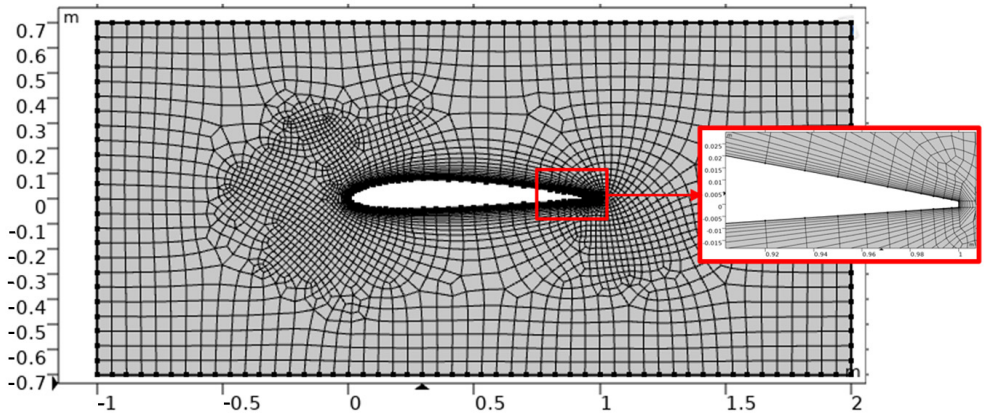


Figure 6. Mesh generation for airfoil NACA 2412 domain and isotropic mesh for boundary layer.

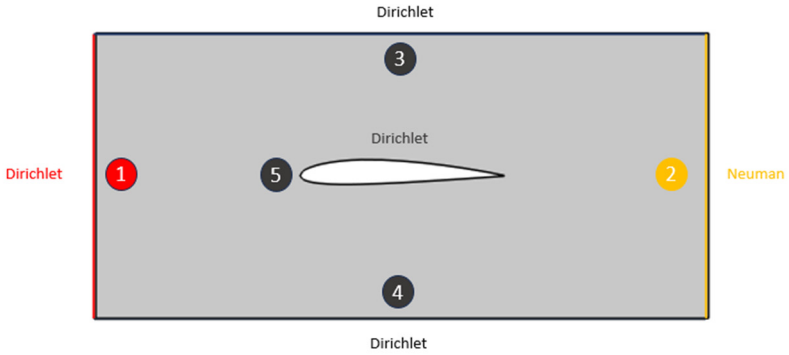


Figure 7. Boundary conditions applied for airfoil NACA 2412 geometry.

This adapted CFD-FEM mesh allows for the efficient and accurate definition of velocity and pressure at as many points in the domain as possible [17].

Assume that the boundary conditions are specified as follows: the velocity vectors (u,v) are set to (0,0) on boundaries 3, 4, and 5 and (0.5,0) on boundary 1, while boundary 2 is left unconstrained. The source term f is zero, and the dynamic viscosity μ is 10^{-5}.

The ensuing discussion will elucidate the key findings of our study, with a particular emphasis on the velocity and pressure plots in the figures below. These plots are critical for understanding the aerodynamic performance of the airfoil, as they provide insights into the flow behavior and pressure distribution around the airfoil surface. The performance metrics will be highlighted, as will the computational efficiency and aerodynamic improvements observed through these plots (Figures 8 and 9).

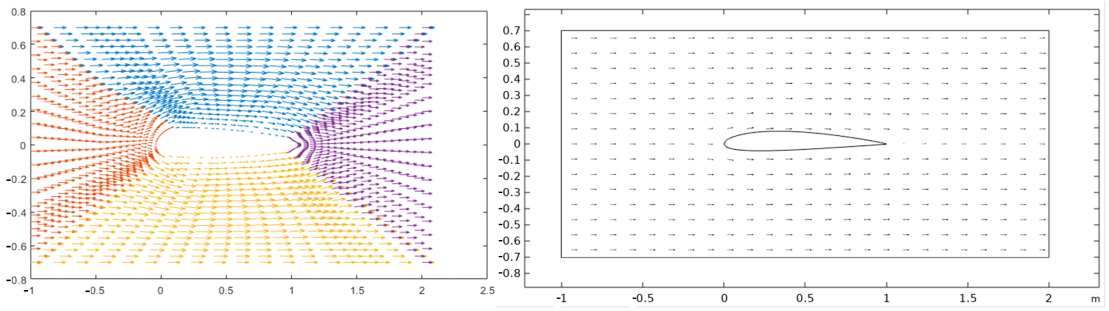


Figure 8. (**a**) Velocity field evaluation using NURBS method; (**b**) velocity field evaluation using FEM.

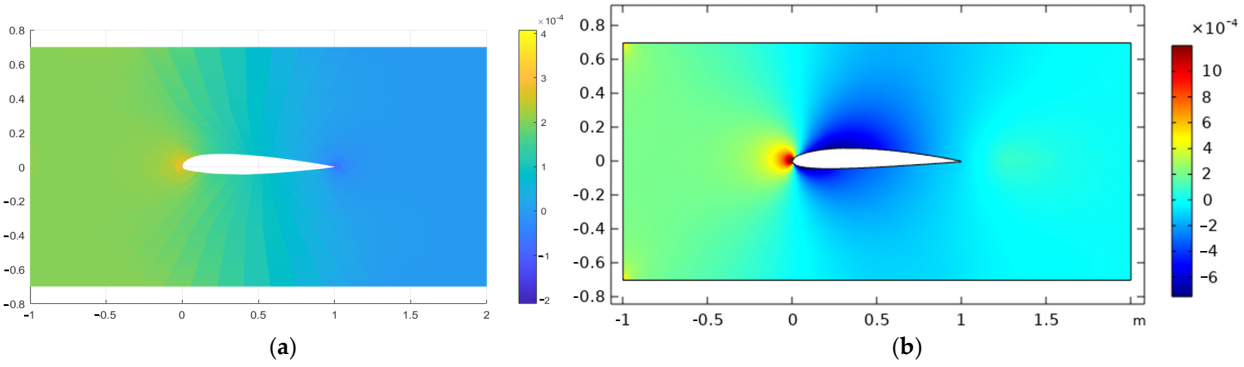
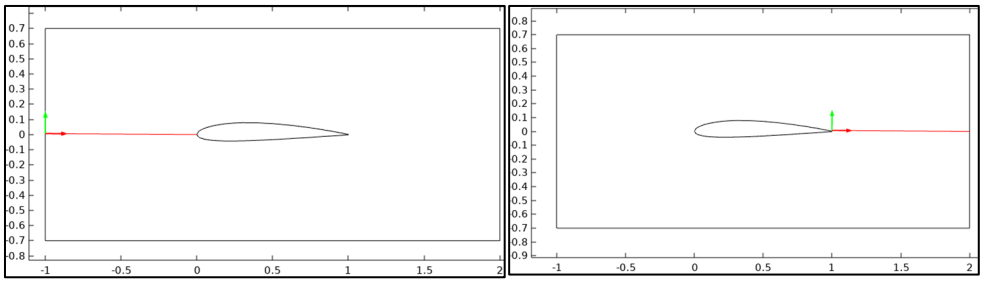


Figure 9. (**a**) Pressure field evaluation using NURBS method; (**b**) pressure field evaluation using FEM.

Furthermore, the numerical results are collected and compared with those obtained from traditional finite element methods.

A horizontal cut at (y = 0) in Figure 10 was conducted to enable a rigorous examination of the pressure distribution across the left and right sides of the airfoil.



(**a**) (**b**)

Figure 10. (**a**) Horizontal line left-side evaluation; (**b**) horizontal line right-side evaluation.

This procedure ensures a thorough analysis of the variations in pressure along this axis and their implications for the aerodynamic performance of the airfoil (Figures 11–14).

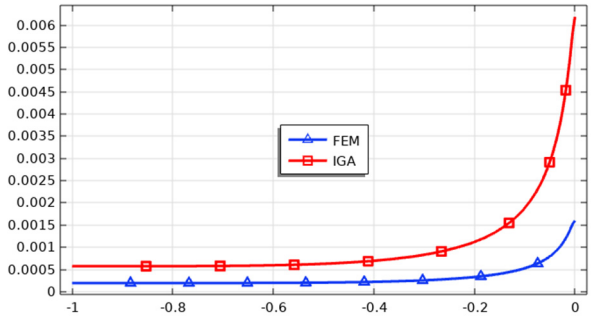


Figure 11. Horizontal line left-side pressure evaluation using FEM and IGA.

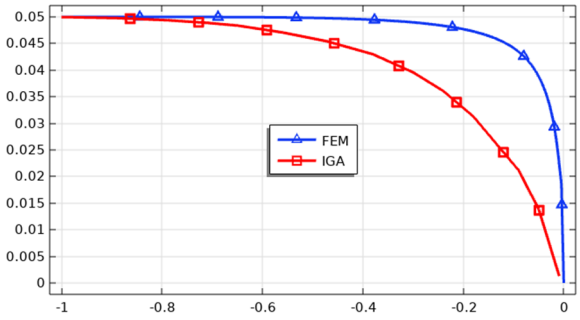


Figure 12. Horizontal line left-side velocity evaluation using FEM and IGA.

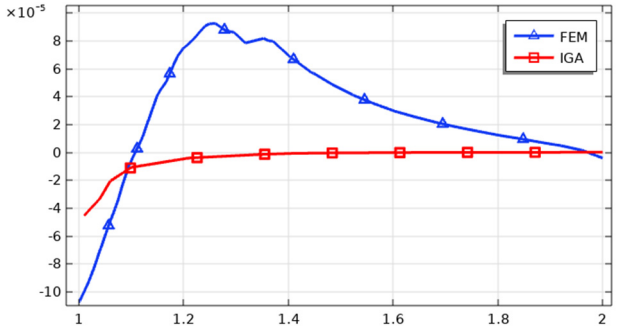


Figure 13. Horizontal line right-side pressure evaluation using FEM and IGA.

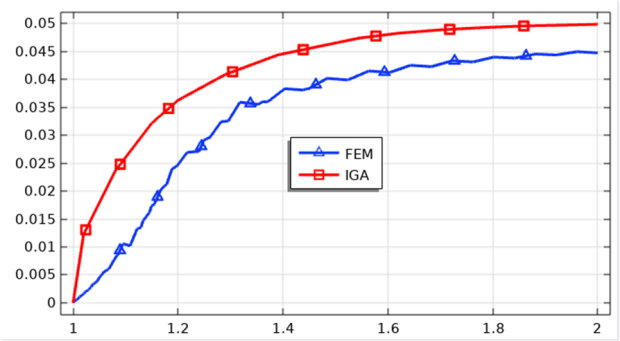
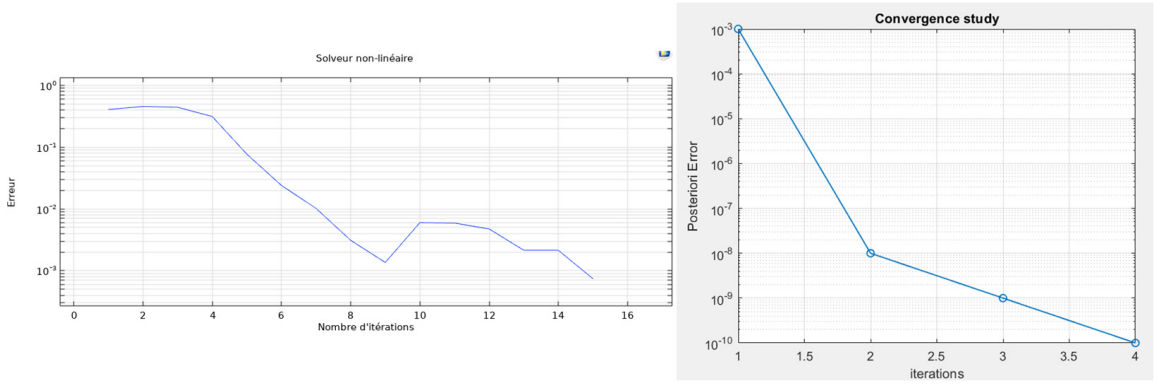


Figure 14. Horizontal line right-side velocity evaluation using FEM and IGA.

Studies show that NURBS-based CFD analysis offers significant improvements over traditional methods, particularly in geometric precision and computational efficiency. NURBS provide a more accurate representation of complex airfoil geometries, resulting in smoother and more detailed pressure and velocity fields. This approach also simplifies mesh generation, reducing computational time and resources [18]. The MATLAB code implementation further demonstrates the practicality of NURBS for real-world applications, highlighting its benefits for precise and efficient aerodynamic simulations.

The values obtained are more precise, and the convergence of both velocity and pressure is achieved more rapidly with IGA than with the FEM. This faster convergence translates to significant cost savings in computational industries.

The enhanced precision of NURBS/IGA is evident in the detailed pressure and velocity plots, which show a 3.75% more accurate representation of the aerodynamic characteristics of the airfoil. The rapid convergence of these values compared to the FEM ones, as shown in Figure 15, not only improves computational efficiency but also reduces the overall computational time/iterations and resources required. This efficiency is particularly beneficial in industries where computational costs are a significant concern, such as aerospace engineering and automotive design.



(**a**) (**b**)

Figure 15. (**a**) The convergence study of the FEM error curve; (**b**) the convergence study of the IGA posteriori error curve.

6. Conclusions

This study introduces a major advancement by integrating NURBS with CFD for airfoil design, specifically the NACA 2412 airfoil. By enhancing geometric representation, NURBS offer improved accuracy and computational efficiency in analyzing complex aerodynamic surfaces. The key findings include the following: NURBS provide a more precise and flexible modeling of airfoil shapes, crucial for aerodynamic optimization. They reduce computational time while improving simulation accuracy, essential for aerospace applications. Comparative analysis shows that NURBS outperform the traditional finite

element method (FEM) in both accuracy and efficiency, capturing nuanced aerodynamic phenomena more effectively; isogeometric analysis using NURBS integrates design and simulation, streamlining the process and reducing mesh generation needs, and rigorous numerical validation confirms the effectiveness of NURBS-based CFD simulations. Overall, NURBS in CFD represents a significant advancement in computational aerodynamics, promising higher accuracy and efficiency for aerodynamic design and optimization, with potential for future application to other complex geometries and flow conditions [19].

Several studies have highlighted the advantages of NURBS/IGA over the traditional FEM and emphasized the improved convergence rates and computational efficiency of IGA in various engineering applications. Additionally, the posteriori error analysis in these studies confirms the reliability of IGA in solving complex problems such as the Stokes–Darcy problem and modeling incompressible viscous fluid dynamics [20].

In conclusion, the adoption of NURBS/IGA in CFD analysis for airfoil design significantly enhances the accuracy of the results while also providing substantial economic benefits through reducing 3.75% of computational costs. This dual advantage positions NURBS/IGA as a promising approach for future advancements in aerodynamic design and analysis, offering both improved precision and efficiency in tackling complex aerodynamic challenges. Additionally, the consideration of boundary conditions, as discussed in similar works, such as in the resolution of Stokes equations using mixed finite element methods [21,22], underscores the importance of accurate boundary treatment in achieving reliable CFD results.

Author Contributions: Conceptualization, S.G., L.E.O., A.E.A. and A.E.; Methodology, S.G., L.E.O., A.E.A. and A.E.; Software, S.G. and A.E.A.; Validation, S.G., L.E.O., A.E.A., A.E., S.V. and M.L.S.; Formal analysis, A.E.A., A.E. and S.V.; Investigation, S.G., L.E.O. and A.E.A.; Resources, L.E.O. and A.E.A.; Data curation, L.E.O. and A.E.; Writing—original draft, S.G.; Writing—review & editing, S.V. and M.L.S.; Visualization, L.E.O., A.E.A., A.E., S.V. and M.L.S.; Supervision, A.E., S.V. and M.L.S.; Project administration, A.E.; Funding acquisition, S.V. and M.L.S. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by Transilvania University of Brasov.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, Y.; Song, Y.; Zhao, S.; Li, X.; Shao, L.; Yan, H.; Ding, S. A comprehensive aerodynamic-thermal-mechanical design method for fast response turbocharger applied in aviation piston engines. *Propuls. Power Res.* **2024**, *13*, 145–165. [CrossRef]
2. Koubaiti, O.; El-Mekkaoui, J.; Elkhalfi, A. Complete study for solving Navier-Lamé equation with new boundary condition using mini element method. *Int. J. Mech.* **2018**, *12*, 46–58.
3. El Fakkoussi, S.; Moustabchir, H.; Elkhalfi, A.; Pruncu, C.I. Application of the Extended Isogeometric Analysis (X-IGA) to Evaluate a Pipeline Structure Containing an External Crack. *J. Eng.* **2018**, *1*, 4125765. [CrossRef]
4. Shrewsbury, G.D. Numerical study of a research circulation control airfoil using Navier-Stokes methods. *J. Aircr.* **1989**, *26*, 29–34. [CrossRef]
5. Cottrell, J.; Hughes, T.; Bazilevs, Y. *Isogeometric Analysis: Toward Integration of CAD and FEA*; John Wiley & Sons: Hoboken, NJ, USA, 2009. [CrossRef]
6. Langer, U.; Moore, S.E.; Neumüller, M. Space–time isogeometric analysis of parabolic evolution problems. *Comput. Methods Appl. Mech. Eng.* **2016**, *306*, 342–363. [CrossRef]
7. Chasapi, M.; Mester, L.; Simeon, B.; Klinkel, S. Isogeometric analysis of 3D solids in boundary representation for problems in nonlinear solid mechanics and structural dynamics. *Int. J. Numer. Methods Eng.* **2022**, *123*, 1228–1252. [CrossRef]
8. Scutaru, M.L.; Guendaoui, S.; Koubaiti, O.; El Ouadefli, L.; El Akkad, A.; Elkhalfi, A.; Vlase, S. Flow of Newtonian Incompressible Fluids in Square Media: Isogeometric vs. Standard Finite Element Method. *Mathematics* **2023**, *11*, 3702. [CrossRef]

9. Hughes, T.J.; Cottrell, J.A.; Bazilevs, Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 4135–4195. [CrossRef]
10. Boffi, D.; Brezzi, F.; Fortin, M. *Mixed Finite Element Methods and Applications*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 44, pp. xiv–685.
11. Nielsen, P.N.; Gersborg, A.R.; Gravesen, J.; Pedersen, N.L. Discretizations in isogeometric analysis of Navier–Stokes flow. *Comput. Methods Appl. Mech. Eng.* **2011**, *200*, 3242–3253. [CrossRef]
12. Wang, Q.; Zhou, W.; Cheng, Y.; Ma, G.; Chang, X.; Liu, B. A NURBS-enhanced improved interpolating boundary element-free method for 2D potential problems and accelerated by fast multipole method. *Eng. Anal. Bound. Elem.* **2019**, *98*, 126–136. [CrossRef]
13. Wang, K.; Yu, S.; Wang, Z.; Feng, R.; Liu, T. Adjoint-based airfoil optimization with adaptive isogeometric discontinuous Galerkin method. *Comput. Methods Appl. Mech. Eng.* **2019**, *344*, 602–625. [CrossRef]
14. Nazir, T.; Abbas, M.; Iqbal, M.K. New cubic B-spline approximation technique for numerical solutions of coupled viscous Burger's equations. *Eng. Comput.* **2021**, *38*, 83–106. [CrossRef]
15. Vázquez, R. A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0. *Comput. Math. Appl.* **2016**, *72*, 523–554. [CrossRef]
16. Guiza, G.; Larcher, A.; Goetz, A.; Billon, L.; Meliga, P.; Hachem, E. Anisotropic boundary layer mesh generation for reliable 3D unsteady RANS simulations. *Finite Elem. Anal. Des.* **2020**, *170*, 103–345. [CrossRef]
17. Ordaz, I.; Li, W.; Campbell, R.L. Automated tetrahedral mesh generation for CFD analysis of aircraft in conceptual design. In Proceedings of the 52nd Aerospace Sciences Meeting, National Harbor, MD, USA, 13–17 January 2014.
18. Li, D.; Tong, S.; Yang, H.; Hu, Q. Time-Synchronized Control for Spacecraft Reorientation with Time-Varying Constraints. *IEEE/ASME Trans. Mechatron.* 2024; *in press*.
19. Chen, X.; Zhong, S.; Liu, T.; Ozer, O.; Gao, G. Manipulation of the flow induced by afterbody vortices using sweeping jets. *Phys. Fluids* **2024**, *36*, 035147. [CrossRef]
20. Elakkad, A.; Elkhalfi, A.; Guessous, N. An a posteriori error estimate for mixed finite element approximations of the Navier-Stokes equations. *J. Korean Math. Soc* **2011**, *48*, 529–550. [CrossRef]
21. El-Mekkaoui, J.; Elkhalfi, A.; Elakkad, A. Resolution of Stokes Equations with the Ca, Boundary Condition Using Mixed Finite Element Method. *Trans. Math.* **2013**, *12*, 586–597.
22. Montassir, S.; Moustabchir, H.; Elkhalfi, A.; Scutaru, M.L.; Vlase, S. Fracture modelling of a cracked pressurized cylindrical structure by using extended iso-geometric analysis (X-IGA). *Mathematics* **2021**, *9*, 2990. [CrossRef]