

NURBS-enhanced finite element method for Euler equations

R. Sevilla, S. Fernández-Méndez and A. Huerta^{*,†}

Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III, E.T.S. de Ingenieros de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Jordi Girona 1, E-08034 Barcelona, Spain

SUMMARY

In this work, the NURBS-enhanced finite element method (NEFEM) is combined with a discontinuous Galerkin (DG) formulation for the numerical solution of Euler equations of gas dynamics. With NEFEM, numerical fluxes along curved boundaries are computed much more accurately due to the exact geometric representation of the computational domain. The proper implementation of the wall boundary condition and the exact geometry provide accurate results even with a linear approximation of the solution. A detailed comparison of NEFEM in front of isoparametric finite elements is presented, demonstrating the superiority of NEFEM approach for both linear and higher-order computations. Copyright © 2008 John Wiley & Sons, Ltd.

Received 30 July 2007; Revised 5 November 2007; Accepted 7 November 2007

KEY WORDS: NURBS; discontinuous Galerkin; CAD; exact geometry representation; high-order isoparametric elements; Euler equations

1. INTRODUCTION

The importance of the geometrical model in finite element (FE) simulations has recently been pointed out by several authors, see [1–6] to name a few. For instance, in [3] the error induced by the approximation of curvilinear geometries with isoparametric elements is analyzed in the context of Poisson and Maxwell problems. Using an exact mapping for the geometry in the numerical solution of Maxwell's equations, the error is reduced by an order of magnitude.

*Correspondence to: A. Huerta, Laboratori de Càlcul Numèric (LaCàN), E.T.S. Ingenieros de Caminos, Universitat Politècnica de Catalunya, Jordi Girona 1, E-08034 Barcelona, Spain.

†E-mail: antonio.huerta@upc.es, <http://www.lacan.upc.es>

Contract/grant sponsor: Ministerio de Ciencia y Tecnología; contract/grant numbers: BIA2007-66965, DPI2007-62395

When a DG formulation is adopted, the importance of the geometrical model is crucial in some applications, such as the numerical solution of Euler equations of gas dynamics. In [1], the authors demonstrate that using a linear approximation for the geometry, it is not possible to converge to the steady-state solution, even if the mesh is drastically refined near the curved boundary. In [2], a detailed study of this problem is presented to conclude that accurate results can only be obtained taking into account the curvature of the domain. More recently, in [5] a new methodology is presented for the computation of the fluxes across curved boundaries but, unfortunately, the proposed method is not conservative.

The importance of the geometrical model in the numerical solution of compressible Euler equations is not exclusive of DG methods. In [7, 8], the problem is identified in the context of finite volume (FV) methods, and more recent advances in this area can be found in [9, 10].

Non-uniform rational B-splines (NURBS, see [11]) are widely used for geometry description in CAD (computer-aided design). This fact has motivated new numerical methodologies considering an exact representation of the computational domain with NURBS, such as the *isogeometric analysis* [4] and the NURBS-enhanced finite element method (NEFEM) [6]. The *isogeometric analysis* considers the same NURBS basis functions for both the description of the entire geometry and the approximation of the solution. This idea was first introduced in [12] in the context of thin shell analysis. The NEFEM also considers an exact representation of the domain, but it differs from the *isogeometric analysis* in two main points: the geometry is given by the NURBS description of the boundary (i.e. the information usually provided by CAD), and standard FE polynomial interpolation is considered for the approximation of the solution. Thus, in the large majority of the domain—for elements not intersecting the boundary—standard FE interpolation and numerical integration are used, preserving the computational efficiency of classical FE techniques. Specifically designed piecewise polynomial interpolation and numerical integration are required only for those FEs along the NURBS boundary.

In [6], NEFEM is applied to the numerical solution of Poisson and electromagnetic scattering problems. Using NEFEM the optimal rate for h -convergence is obtained for any polynomial degree. Moreover, exponential convergence is observed when a p -refinement strategy is considered. In the context of electromagnetic scattering applications, the use of NEFEM reveals an important improvement with respect to classical isoparametric FE. For the same spatial discretization, an important reduction in the error is observed, in some cases more than one order of magnitude. In addition, for a desired precision NEFEM is also more efficient because it allows to compute the solution with an important reduction in the number of degrees of freedom.

In this paper, NEFEM is presented as a powerful method for numerical solution of Euler equations using a DG formulation. Sections 2 and 3 recall the system of Euler equations and its DG discretization. The basic concepts of NEFEM are recalled in Section 4, with special attention to the interpolation and numerical integration in those elements with one curved edge defined by NURBS. Section 5 presents a classical test for inviscid flow methods in order to evaluate the efficiency and accuracy of NEFEM in front of classical FE. Low- and high-order approximations are tested, and a comparison between isoparametric FE and NEFEM in terms of the entropy error and other aerodynamic quantities of interest is presented, demonstrating the superiority of NEFEM approach for the simulation of compressible flow problems.

2. EULER EQUATIONS

Euler equations of gas dynamics express the conservation of mass, momentum and energy for a compressible, inviscid and non-conducting fluid. The strong form of these conservation laws, in the absence of external volume forces, can be expressed in conservative form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_k(\mathbf{U})}{\partial x_k} = \mathbf{0} \quad (1)$$

where Einstein notation is assumed (that is, repeated indices are implicitly summed over), \mathbf{U} is the vector of conservation variables and $\mathbf{F}_k(\mathbf{U})$ are the flux vectors for each spatial dimension x_k , that is

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho \mathbf{v} \\ \rho E \end{pmatrix}, \quad \mathbf{F}_k(\mathbf{U}) = \begin{pmatrix} \rho v_k \\ \rho \mathbf{v} v_k + \mathbf{e}_k p \\ (\rho E + p) v_k \end{pmatrix}$$

where ρ is the density, $\rho \mathbf{v}$ is the momentum, ρE is the total energy per unit volume, \mathbf{e}_k is the unitary vector in the x_k direction and p is the pressure, see [13] for more details.

An equation of state, relating the internal energy to pressure and density, completes this system of nonlinear hyperbolic equations. For a perfect polytropic gas, the equation of state is

$$p = (\gamma - 1) \rho \left(E - \frac{1}{2} \|\mathbf{v}\|^2 \right)$$

where γ is the ratio of the specific heat coefficients (specific heat at constant pressure to specific heat at constant volume), with value $\gamma = 1.4$ for air.

A usual quantity for postprocess of inviscid flow computations is the Mach number, defined as

$$M = \frac{\|\mathbf{v}\|}{c}$$

where $c = \sqrt{\gamma p / \rho}$ is the speed of sound. For a more detailed presentation of the Euler equations, see, for instance, [14–16].

Other useful quantities for the evaluation of the accuracy are the entropy error

$$\varepsilon_{\text{ent}} = \frac{p}{p_\infty} \left(\frac{\rho_\infty}{\rho} \right)^\gamma - 1$$

the pressure loss

$$p_{\text{loss}} = \frac{p}{p_\infty} \left(\frac{1 + 0.5(\gamma - 1)M^2}{1 + 0.5(\gamma - 1)M_\infty^2} \right)^{\gamma/\gamma - 1}$$

and the pressure coefficient

$$C_p = \frac{p - p_\infty}{0.5 \rho_\infty v_\infty^2}$$

where the subscript ∞ indicates free-stream values, see [2, 5] for more details.

3. DISCONTINUOUS GALERKIN FORMULATION

This section recalls the basic concepts of DG [17] for the solution of Euler equations in an open bounded domain $\Omega \subset \mathbb{R}^n$. A regular partition in elements $\bar{\Omega} = \bigcup_e \bar{\Omega}_e$ is assumed, and an element-by-element discontinuous approximation space is considered.

The weak problem for the strong form of Euler equations (1) is stated for each element Ω_e . By multiplying by a test vector function \mathbf{W} , integrating over Ω_e and integrating by parts, the following equation is obtained:

$$\int_{\Omega_e} \mathbf{W} \cdot \frac{\partial \mathbf{U}_e}{\partial t} d\Omega - \int_{\Omega_e} \frac{\partial \mathbf{W}}{\partial x_k} \cdot \mathbf{F}_k(\mathbf{U}_e) d\Omega + \int_{\partial\Omega_e} \mathbf{W} \cdot \mathbf{F}_{n_e}(\mathbf{U}_e) d\Gamma = 0 \quad \forall \mathbf{W}$$

where \mathbf{U}_e denotes the restriction of \mathbf{U} to element Ω_e , \mathbf{n}_e is the outward unit normal vector on $\partial\Omega_e$, and the normal flux is defined as

$$\mathbf{F}_n(\mathbf{U}) = \mathbf{F}_k(\mathbf{U})n_k$$

with n_k the k th component of \mathbf{n} . As usual in DG methods, to take into account the discontinuous nature of the approximation, the normal flux at the boundary of the element is replaced by a numerical flux, $\widehat{\mathbf{F}}_{n_e}(\mathbf{U}_e, \mathbf{U}_e^{\text{out}})$, which is evaluated in terms of the solution in the current element Ω_e and the solution at neighboring elements

$$\mathbf{U}_e^{\text{out}}(\mathbf{x}) = \lim_{\varepsilon \rightarrow 0^+} \mathbf{U}(\mathbf{x} + \varepsilon \mathbf{n}_e) \quad \text{for } \mathbf{x} \in \partial\Omega_e \quad (2)$$

The resulting DG weak formulation, to be discretized at each element Ω_e , is

$$\int_{\Omega_e} \mathbf{W} \cdot \frac{\partial \mathbf{U}_e}{\partial t} d\Omega - \int_{\Omega_e} \frac{\partial \mathbf{W}}{\partial x_k} \cdot \mathbf{F}_k(\mathbf{U}_e) d\Omega + \int_{\partial\Omega_e} \mathbf{W} \cdot \widehat{\mathbf{F}}_{n_e}(\mathbf{U}_e, \mathbf{U}_e^{\text{out}}) d\Gamma = 0 \quad \forall \mathbf{W} \quad (3)$$

Some conditions are required for the definition of a numerical flux function: it must be conservative, Lipschitz and verify some consistency conditions. Some popular flux functions for the numerical solution of the Euler equations are the exact Riemann solver, the Roe solver, the Lax–Friederichs solver or the Harten–Lax–van Leer (HLLE) solver, see [18].

Boundary conditions are implemented following the ideas in [19], initially developed in the context of FV methods. Fictitious elements are considered along the boundary, and the value of the solution is set to impose the boundary conditions through the numerical flux. As usual in the solution of Euler equations, a characteristic analysis is performed at the boundary to decide the quantities to be prescribed, see, for instance, [14–16, 20] for implementation details.

The DG formulation (3) is discretized in each element, leading to a system of ordinary differential equations

$$\mathbf{M} \frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0} \quad (4)$$

where \mathbf{U} is the vector of nodal values (or approximation coefficients in a more general case), \mathbf{M} is a block diagonal mass matrix and $\mathbf{R}(\mathbf{U})$ is the residual vector. As shown in the examples, the spatial discretization may be performed using isoparametric FE or NEFEM, see Section 4. The ODE system (4) is advanced in time using the explicit third-order total variation diminishing

Runge–Kutta (TVD-RK) scheme presented in [21]. Nevertheless, it is worth noting that semi-implicit and implicit time integration schemes seem to be an efficient alternative to steady-state computations, see [22, 23]. The stability condition for the TVD-RK method requires a Courant–Friedrichs–Lewy (CFL) number $\leq 1/(2p+1)$, where p is the degree of the functional approximation, see [24].

The evaluation of the residual $\mathbf{R}(\mathbf{U})$, which involves the flux at the interior of the elements and their boundaries, can be carried out, as usual, with two non-equivalent options: a quadrature-free implementation and a full quadrature version, see [13]. With a quadrature-free implementation, the flux at the integration points is interpolated in terms of the flux at nodal values, whereas with a full quadrature version, fluxes are evaluated at integration points, in terms of the solution at each integration point. The use of a quadrature-free implementation leads to an important save in computational cost, thanks to the use of elemental matrices, instead of a loop on integration points. Moreover, for triangles with straight sides (or tetrahedras with planar faces) these elemental matrices can be computed, using the Jacobian, from matrices previously computed at the reference element, see [25], with an important reduction in computational time. However, numerical experiments reveal that a quadrature-free implementation for the Euler equations suffers from instability problems in the vicinity of stagnation points, see [26] for a detailed explanation. Thus, all computations presented here are obtained with a full quadrature implementation.

4. NEFEM FUNDAMENTALS

A domain $\Omega \subset \mathbb{R}^2$ is considered, whose boundary $\partial\Omega$, or a portion of its boundary, is defined by NURBS curves. A NURBS curve is a curve parametrized by a piecewise rational function, whose definition changes at the so-called *breakpoints*, see [11] for a detailed definition. In fact, in practical applications CAD manipulators use *trimmed* NURBS, which are defined by a restriction of the NURBS parametrization to a subspace of the parametric space.

A triangularization of the domain $\bar{\Omega} = \bigcup_e \bar{\Omega}_e$ is also assumed, such that every triangle Ω_e has at most one side, Γ_e , on the NURBS boundary. Figure 1 shows a domain with part of the boundary described by a NURBS circular curve and a valid triangulation for NEFEM. As usual in mesh generation codes, it is also assumed that every curved boundary side belongs to a unique NURBS. That is, a side cannot be defined by different NURBS curves, or, equivalently, every trimmed NURBS is cut to pieces corresponding to the boundary sides. It is important to note that the

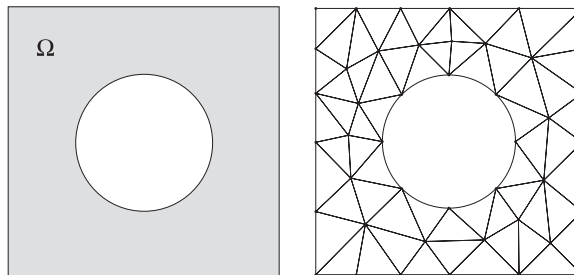


Figure 1. Physical domain with part of the boundary defined by a circular NURBS curve (left) and a valid triangulation for NEFEM (right).

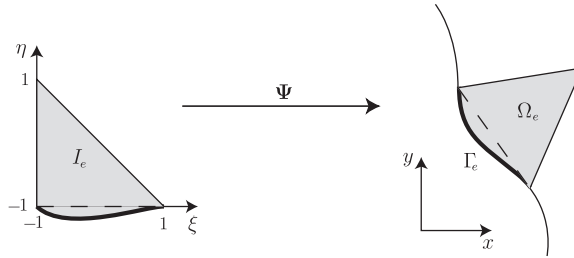


Figure 2. Linear transformation, mapping the curved element $I_e := \Psi^{-1}(\Omega_e)$ to the physic element Ω_e .

breakpoints, which characterize the piecewise nature of NURBS, are independent of the mesh discretization. Thus, the NURBS parametrization can change its definition inside one side, that is, breakpoints may belong to element sides and do not need to coincide with FE nodes. This is another major advantage compared with *isogeometric analysis* [4].

All elements not intersecting the NURBS boundary use the standard FE interpolation and numerical integration. The basis of NEFEM for an element with one side on the NURBS boundary is recalled next.

Let Ω_e be an element with two straight interior sides and one side defined by a trimmed NURBS, $\Gamma_e = \mathbf{C}([\lambda_1^e, \lambda_2^e])$. A *linear transformation* Ψ from local coordinates to physical coordinates is considered. The linear transformation Ψ^{-1} maps the element in physical space Ω_e to a curved element $I_e := \Psi^{-1}(\Omega_e)$, see Figure 2. Note that I_e plays the role of the reference element in standard FE, but here it depends on the definition of its NURBS side. Thus, special numerical strategies are required for every element I_e , see the following sections.

Remark 1

In order to simplify the presentation, it is assumed that the interior vertex of Ω_e is mapped to the vertex $(-1, 1)$ in I_e . The implementation of this condition only requires a proper local numbering of the vertices of the element.

4.1. FE polynomial basis

In order to work with standard FE polynomial approximations, Lagrange polynomials, i.e. standard nodal interpolation, can be considered. In fact, they can be defined on the curved triangle, I_e , in the reference domain or, equivalently, in the actual element in the physical domain, Ω_e . The use of a linear transformation from the local (reference) coordinates $\xi = (\zeta, \eta)^T$ in I_e to Cartesian coordinates $\mathbf{x} = (x, y)^T$ in Ω_e ensures that a complete polynomial interpolation of degree m in ξ leads to a polynomial interpolation with the same degree in \mathbf{x} . Thus, consistency and accuracy of the approximation are ensured even for elements Ω_e far from being a straight-sided element.

In order to make the computation of the Lagrange polynomials more systematic, for any order and for any distribution of nodes, the implementation proposed in [27] is adopted.

Different options can be considered to define nodal distributions in I_e . For low-order elements, equally spaced nodal distributions can be implemented. Nevertheless, high-order computations require special distributions of nodes in order to reduce the condition number of the resulting elemental matrices, see [28, 29] for details. Fekete points [30] are a good example of such distributions. For curved elements, nodes can be located in the straight-sided triangle given by the

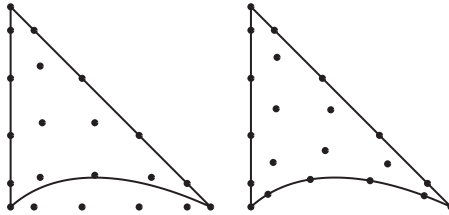


Figure 3. Fifth-order nodal distributions in I_e : Fekete points in the straight-side triangle (left) and adapted to the NURBS side (right).

vertexes of I_e , see left distribution in Figure 3, or adapted to the exact geometry using the NURBS description, see right distribution in Figure 3. Adapted distributions have a positive influence on the condition number. Moreover, numerical experiments reveal that the non-adapted distribution leads to a more restrictive CFL stability condition; thus, in the numerical examples adapted distributions are used.

4.2. Numerical integration

The weak form to be solved requires integration along element sides and in the interiors. All the elements not having a side along the NURBS boundary are integrated using standard procedures. Elements with one side, Γ_e , on the NURBS boundary require special attention.

All line integrals to be computed in NEFEM can be expressed as

$$\int_{\Gamma_e} f \, d\ell = \int_{\lambda_1^e}^{\lambda_2^e} f(\mathbf{C}(\lambda)) |J_{\mathbf{C}}(\lambda)| \, d\lambda$$

where f is a generic function (polynomial), the side of the element is given by a trimmed NURBS $\Gamma_e = \mathbf{C}([\lambda_1^e, \lambda_2^e])$, and $|J_{\mathbf{C}}|$ denotes the norm of the differential of the NURBS parametrization \mathbf{C} (this is not a polynomial). As usual, a 1D numerical quadrature is used for the numerical computation of the integral, namely

$$\int_{\Gamma_e} f \, d\ell \approx \sum_{i=1}^{n_{ip}} f(\mathbf{C}(\lambda_i)) |J_{\mathbf{C}}(\lambda_i)| \omega_i \quad (5)$$

where λ_i and ω_i are, respectively, the coordinates and weights of the n_{ip} integration points in $[\lambda_1^e, \lambda_2^e]$. Recall that the parametrization of a trimmed NURBS, \mathbf{C} , is a piecewise rational function whose definition changes at breakpoints. Thus, an independent numerical quadrature must be considered at every interval between breakpoints to account for the discontinuous nature of the parametrization. Numerical experiments reveal that Gauss–Legendre quadratures are a competitive choice in front of other quadrature rules such as trapezoidal and Simpson composite rules or Romberg's integration.

NEFEM also requires to compute integrals over an element Ω_e with one side Γ_e on the NURBS boundary, see Figure 2, that is

$$\int_{\Omega_e} f \, dx \, dy = |J_{\Psi}| \int_{I_e} f \, d\xi \, d\eta \quad (6)$$

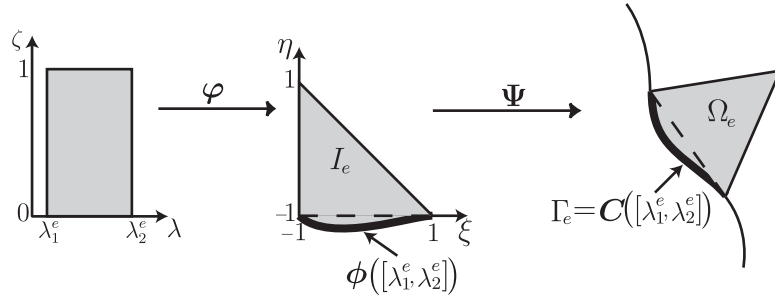


Figure 4. Transformation from $[\lambda_1^e, \lambda_2^e] \times [0, 1]$ to I_e and Ω_e .

where $|J_{\Psi}|$ is the determinant of the Jacobian of the linear transformation Ψ . The computation of (6) requires a numerical quadrature for every curved element I_e . Reference [31] presents different alternatives and discusses their advantages and disadvantages. The best alternative corresponds to the definition of a transformation from the rectangle $[\lambda_1^e, \lambda_2^e] \times [0, 1]$ to the curved element I_e , see Figure 4. That is, under the non-restrictive assumption that nodes are numbered following Remark 1,

$$\begin{aligned} \boldsymbol{\phi} : [\lambda_1^e, \lambda_2^e] \times [0, 1] &\longrightarrow I_e \\ (\lambda, \zeta) &\longmapsto \begin{Bmatrix} \varphi_1 \\ \varphi_2 \end{Bmatrix} := \begin{Bmatrix} \phi_1(\lambda)(1-\zeta) - \zeta \\ \phi_2(\lambda)(1-\zeta) + \zeta \end{Bmatrix} \end{aligned} \tag{7}$$

where $\boldsymbol{\phi} = (\phi_1, \phi_2)^T := \Psi^{-1} \circ \mathbf{C}$ is the parametrization of the trimmed NURBS corresponding to the curved side in I_e . Note that such a parametrization is linear in ζ ; this induces some important practical advantages.

Thus, using the transformations shown in Figure 4, integral (6) is computed as

$$\int_{\Omega_e} f \, dx \, dy = |J_{\Psi}| \int_{I_e} f \, d\zeta \, d\eta \simeq |J_{\Psi}| \sum_{i=1}^{n_{ip}} \sum_{j=1}^{m_{ip}} f(\xi_{ij}) |J_{\boldsymbol{\phi}}(\lambda_i, \zeta_j)| \omega_i \varpi_j \tag{8}$$

where n_{ip} and m_{ip} are the numbers of integration points in the λ and ζ directions, respectively, $\xi_{ij} := \boldsymbol{\phi}(\lambda_i, \zeta_j)$, $\{\lambda_i, \omega_i\}$ and $\{\omega_i, \varpi_i\}$ are the 1D quadrature points and weights for $[\lambda_1^e, \lambda_2^e]$ and $[0, 1]$, respectively, and $|J_{\boldsymbol{\phi}}|$ is the determinant of the Jacobian of the transformation $\boldsymbol{\phi}$.

5. NUMERICAL RESULTS

To evaluate the accuracy and efficiency of the proposed methodology, a classical inviscid test case is considered: subsonic flow around a circle at free-stream Mach number $M_{\infty} = 0.3$. A DG formulation is adopted for every numerical computation. As first studied in [1] and later in [2, 5, 32], DG discretization of the wall boundary condition is very sensitive to the geometrical description of curved boundaries. More precisely, in [1] the authors show that it is not possible to converge to the correct physical solution if the computational boundary is approximated with piecewise linear polynomials.

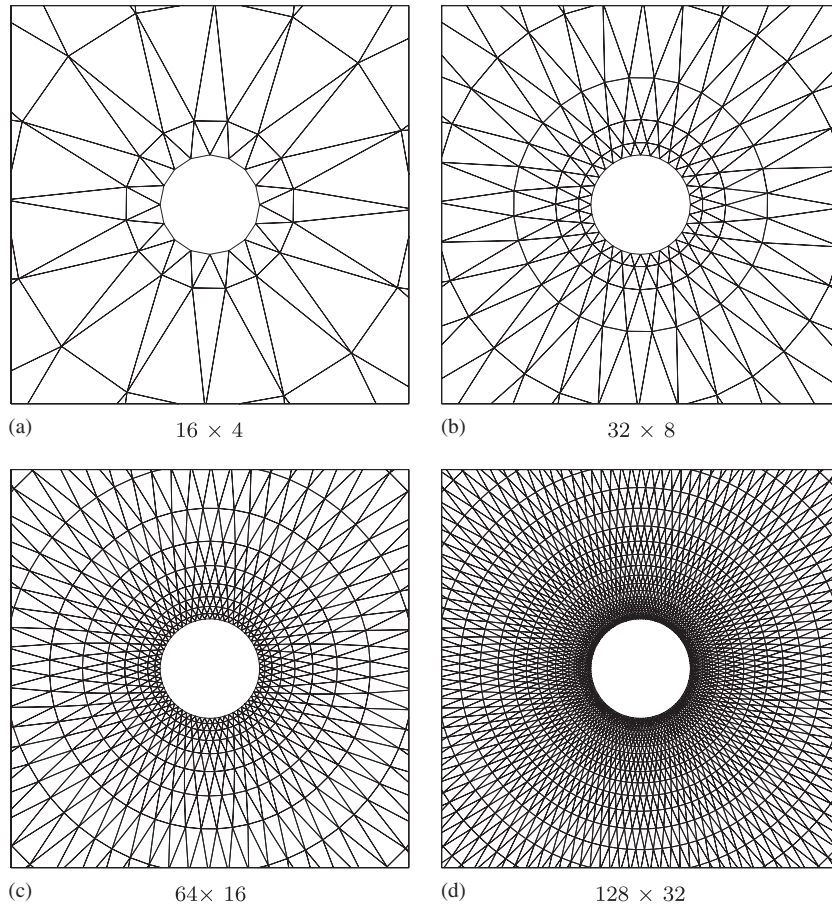


Figure 5. Details of O -meshes for low-order computations.

The behavior of NEFEM for both linear and high-order approximations is studied and compared with isoparametric FE in the following sections. All computations are stopped when the density residual is reduced to 10^{-10} in the $\mathcal{L}_2(\Omega)$ norm. The approximate Roe solver is considered for the evaluation of the numerical flux. In fact, the Roe flux provides more accurate results than the Lax–Friederichs one for low-order approximations, but no significant differences are observed for high-order approximations. However, it is very important to remark that the conclusions of the work derived from the comparison of NEFEM and FEM are exactly the same with a Lax–Friederichs flux.

5.1. Low-order computations

Four O -meshes with 16×4 , 32×8 , 64×16 and 128×32 nodes (i.e. 128, 512, 2048 and 8192 elements, respectively) are considered for low-order computations. A detailed view of these meshes near the circle is represented in Figure 5, see [5] for mesh generation details.

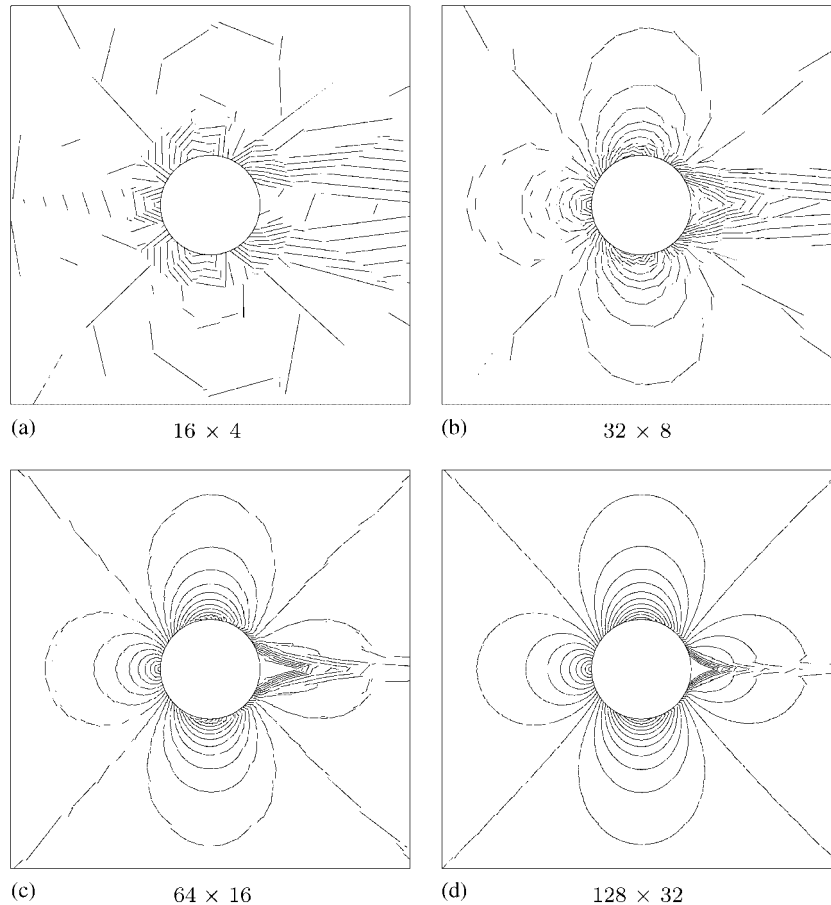
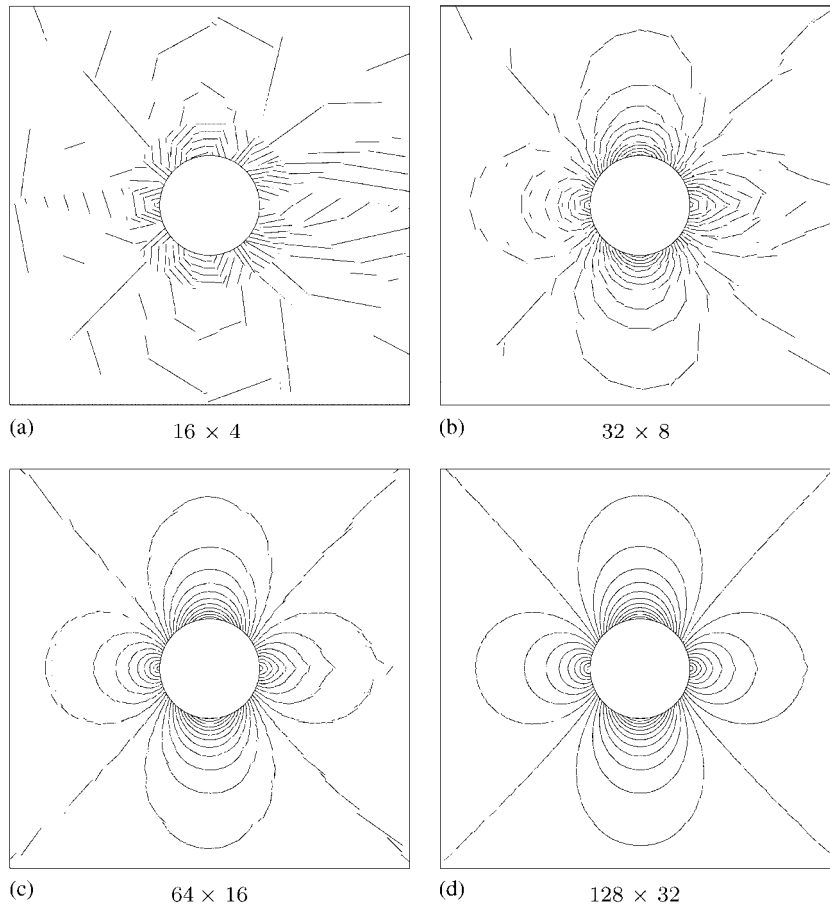


Figure 6. Mach number isolines with isoparametric FE and $p=1$.

Figure 6 shows Mach number isolines for isoparametric FE with linear approximation. The results corroborate the conclusions first published by Bassi and Rebay [1] in the context of DG methods. Even if the mesh is highly refined near the circle, for instance, using the fine mesh of Figure 5 with 128 curved elements along the circular boundary, a non-physical entropy production is observed behind the wall. As it is commented in [33], the singularities of the polygonal approximation of the boundary generate entropy, and the solution develops a non-physical wake that makes the convergence to the correct solution impossible.

Figure 7 shows Mach number isolines computed with NEFEM using linear approximation. The results reveal a very good symmetry of the Mach number patterns, even if coarse meshes are used. Moreover, NEFEM allows convergence to the correct physical solution using the fine mesh with a piecewise linear approximation of the solution. The exact computation of the outward unit normal improves the imposition of the solid wall boundary condition. This issue and the exact representation of the domain drastically reduce the entropy production compared with isoparametric FE.

Figure 7. Mach number isolines with NEFEM and $p=1$.Table I. \mathcal{L}^2 entropy error at the upper mid of the circle and convergence rate (r) for $p=1$.

| Mesh | FEM | | NEFEM | |
|-----------------|----------------------------|------|----------------------------|------|
| | ε_{ent} | r | ε_{ent} | r |
| 16×4 | $1.85\text{E}-02$ | — | $1.18\text{E}-02$ | — |
| 32×8 | $1.21\text{E}-02$ | 0.61 | $3.07\text{E}-03$ | 1.94 |
| 64×16 | $7.38\text{E}-03$ | 0.71 | $6.72\text{E}-04$ | 2.19 |
| 128×32 | $4.19\text{E}-03$ | 0.82 | $1.53\text{E}-04$ | 2.13 |

Entropy errors, in \mathcal{L}^2 norm, measured on the upper mid circle are reported in Table I. For isoparametric FE, the entropy production observable in Figure 6 deteriorates the h -convergence rate. In contrast, NEFEM exhibits the optimal convergence rate for linear approximation. Results

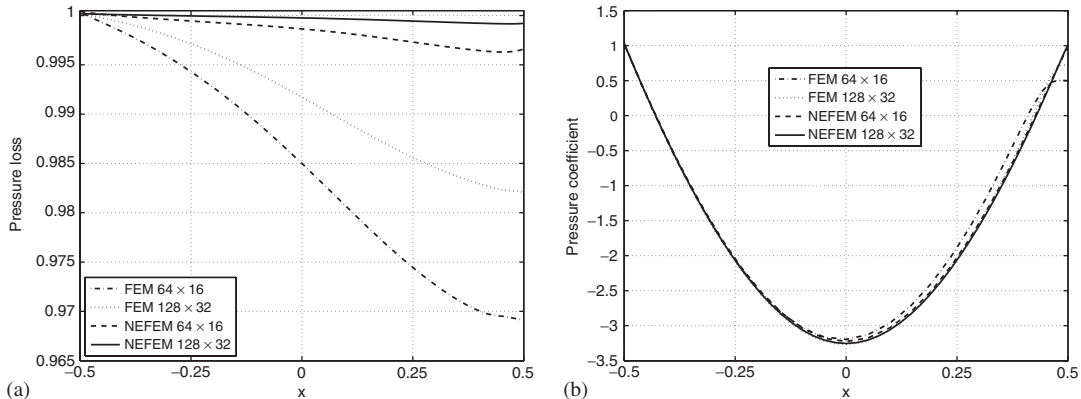


Figure 8. (a) Pressure loss distribution and (b) pressure coefficient distribution, at the upper mid of the circle for $p = 1$.

of Table I also show that to achieve an entropy error of 4×10^{-3} with isoparametric FE a fine mesh is necessary, whereas the second mesh suffices to obtain the same precision with NEFEM. Thus, the extra computational cost associated with the numerical integration of NEFEM is clearly surpassed by the drastic saving in number of degrees of freedom (ten times fewer degrees of freedom).

Figure 8 shows pressure loss and pressure coefficient distributions on the upper mid of the circle. At the most critical point, the stagnation point behind the circle, the maximum pressure loss error with isoparametric FE in the fine mesh is 1.8×10^{-2} , whereas NEFEM maximum error is reduced more than one order of magnitude, namely 8.4×10^{-4} . Moreover, in the fine mesh, the pressure coefficient error at the stagnation point is 2.8×10^{-1} for standard FE and 4×10^{-3} for NEFEM, almost two orders of magnitude more precise for the same number of degrees of freedom.

In [1], it is shown that the use of a good approximation of the normal on the solid boundary is crucial to accurately treat the DG boundary. In particular, the quality of the linear approximation is drastically improved using a quadratic approximation of the geometry. In fact, as noted by other authors [1, 5, 32, 33], the use of an accurate normal is crucial. If the exact normal is used with the linear approximation, the results are drastically improved (similar improvement as with superparametric). Figure 9 shows Mach number isolines computed with linear approximation and exact normal at the boundary. Note the reduced entropy production behind the circle. However, although the global error (entropy error in the whole domain) is reduced using the exact normal due to the reduction in the entropy, the error measured on the circle surface is almost the same than using standard linear isoparametric elements. Moreover, it is important to note that NEFEM results are more accurate. Thus, exact normal evaluation is not enough and an accurate geometrical approximation (exact approximation for NEFEM) is also important.

It is also important to note that improving the approximation of the normal (or superparametric representations) does not significantly improve the solution if high-order elements are employed [1]. Therefore, this analysis will not be included in the following section for high-order elements.

To conclude, it is important to recall that the issue associated with solid wall boundary conditions is not specific to DG methods. In [8], the same problem had been observed by Barth in the FV framework. Thus, the benefits of the NURBS-enhanced concept are extensible to FV methods.

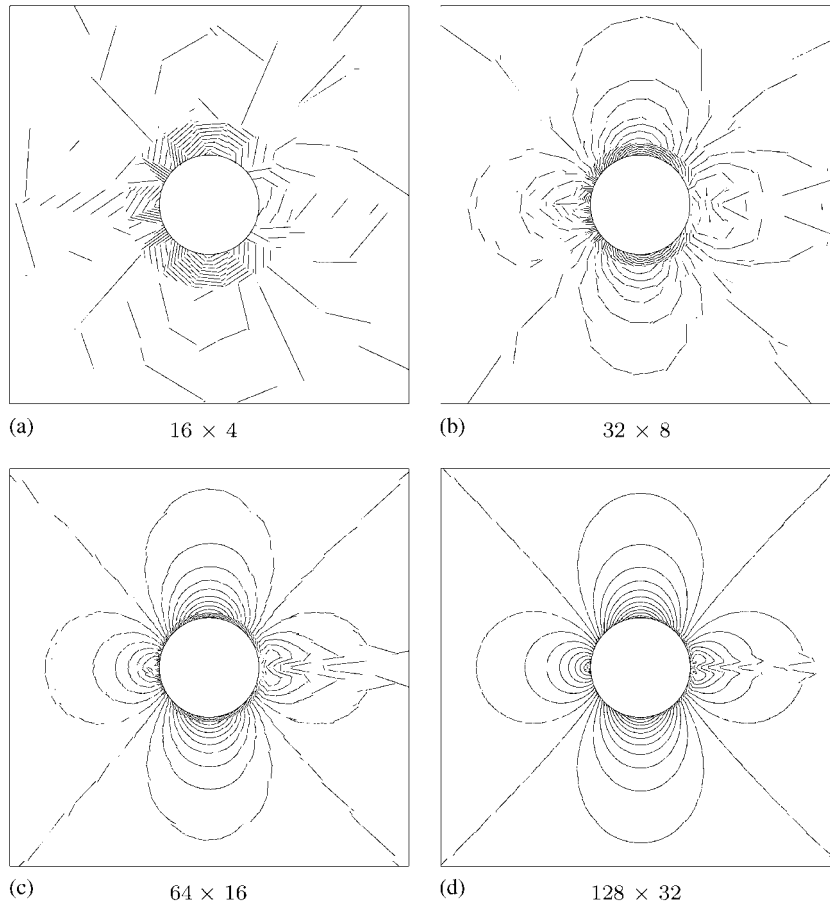


Figure 9. Mach number isolines using the exact normal for flux computations and $p=1$.

Finally, a comparison between h - and p -refinement strategies for NEFEM is presented in order to recall the superiority of high-order interpolations, see [34, 35]. For the p -refinement strategy, the coarser mesh in Figure 5 is considered with an interpolation degree $p=1, 2, 4$. Figure 10 shows the logarithm of the \mathcal{L}^2 entropy error in the upper mid of the circle as a function of the logarithm of the number of degrees of freedom. Although the h -refinement process shows the optimal rate of convergence (straight line with slope $p/2$), it is clearly surpassed by the exponential decay of the error for the p -refinement strategy. Thus, the advantage of using high-order interpolations is clear, especially in the context of a DG formulation. The performance of NEFEM for high-order computations is studied next.

5.2. High-order computations

Following [33], for high-order computations outer boundaries are placed near the obstacle, and the symmetry with respect to the x -axis is used in order to reduce the number of degrees of freedom. Two computational meshes, represented in Figure 11, are considered. The coarser mesh has

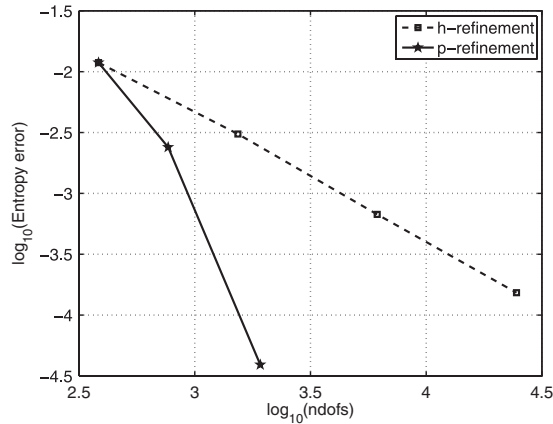


Figure 10. \mathcal{L}^2 entropy error at the upper mid of the circle for h - and p -refinement using NEFEM.

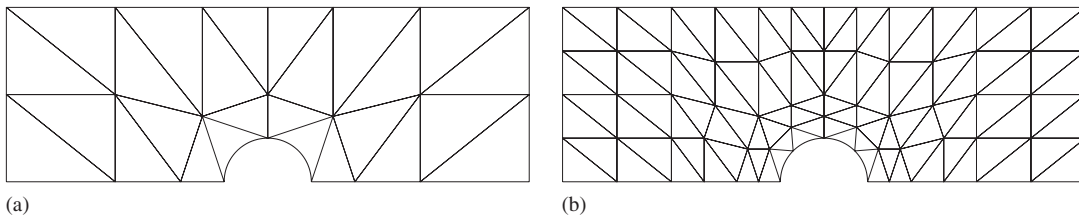


Figure 11. Computational meshes for high-order computations.

26 elements and only two curved elements to describe a half of a circle, and the fine mesh is obtained by uniform refinement. A new measure is proposed in order to quantify the so-called *error in symmetry* and compare different orders or methods, namely, the difference between the Mach number on the right part of the circle ($x > 0$) and on the left part ($x < 0$) in the \mathcal{L}^2 norm.

Figures 12 and 13 show Mach number isolines in the coarser mesh of Figure 11 using isoparametric FE and NEFEM, respectively. For both methods, a degree of $p=2$ or $p=4$ is not sufficient to properly capture the solution, and the entropy production behind the circle is clearly observed. For higher degrees, namely for $p=6$ and 8 , the isoparametric FE solution shows a slight asymmetry with respect to the y -axis due to the spurious entropy production. The error in symmetry is 3.7×10^{-3} for $p=6$ and 3.1×10^{-3} for $p=8$. NEFEM with $p=6$ exhibits better symmetry than isoparametric FE with $p=8$. The error in symmetry with NEFEM is 7.4×10^{-4} for $p=6$ and 2.8×10^{-4} for $p=8$.

Figures 14 and 15 show Mach number isolines for the fine mesh in Figure 11 using isoparametric FE and NEFEM, respectively. Quadratic elements, for both methods and this mesh, do not properly capture the solution. As the degree of the approximation increases, the results improve. However, the improvement is clearly faster with NEFEM compared with isoparametric FE. Isoparametric FE induce an error in symmetry of 2.5×10^{-4} for $p=4$ and 5×10^{-5} for $p=6$, whereas NEFEM has an error in symmetry of 5.5×10^{-5} for $p=4$ and 5.2×10^{-6} for $p=6$. Thus, NEFEM has

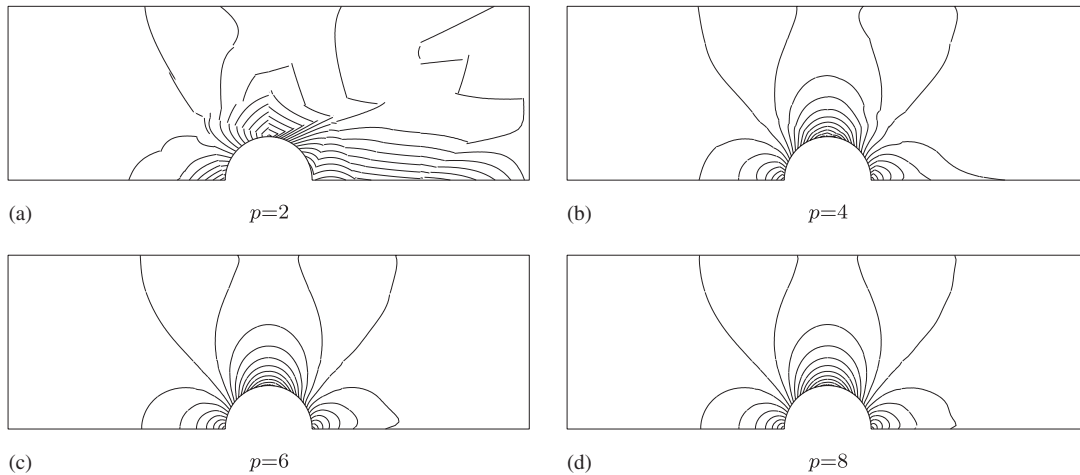


Figure 12. Mach number isolines for isoparametric FE in the coarse mesh.

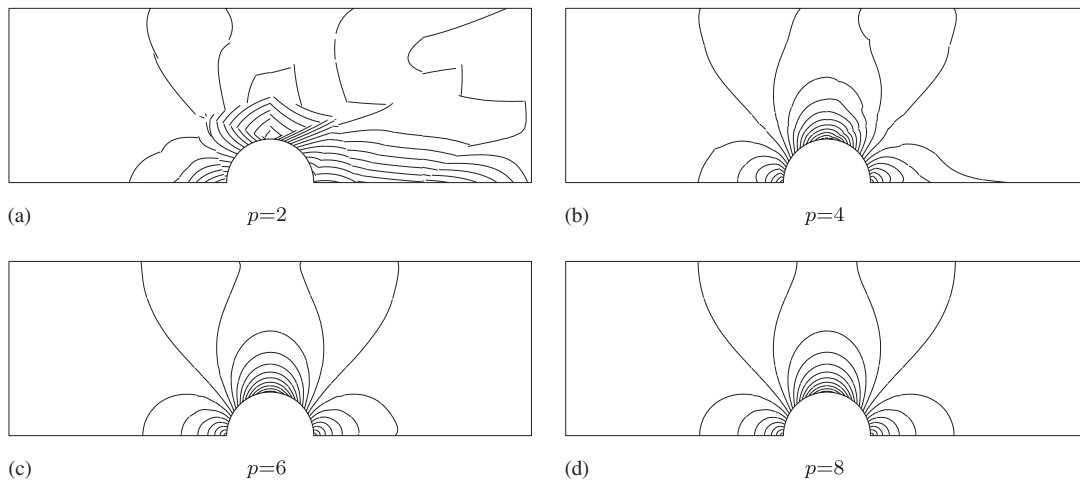


Figure 13. Mach number isolines for NEFEM in the coarse mesh.

a comparable accuracy with $p = 4$ as isoparametric FE with $p = 6$. Thus, the computational overhead of NEFEM becomes negligible compared with the drastic reduction in degrees of freedom.

The accuracy in terms of the entropy error for high-order computations is analyzed next. Figure 16 compares isoparametric FE and NEFEM for the entropy error (in \mathcal{L}^2 norm on the upper mid of the circle) as a function of the square root of the number of degrees of freedom. For isoparametric FE computations, p -convergence is clearly deteriorated for high-order approximations, whereas NEFEM maintains the exponential p -convergence. The results reveal that the best accuracy that can be obtained with isoparametric FE on the coarser mesh corresponds to an error of 5×10^{-4} . In order to improve the accuracy of the isoparametric FE computation, an hp -refinement

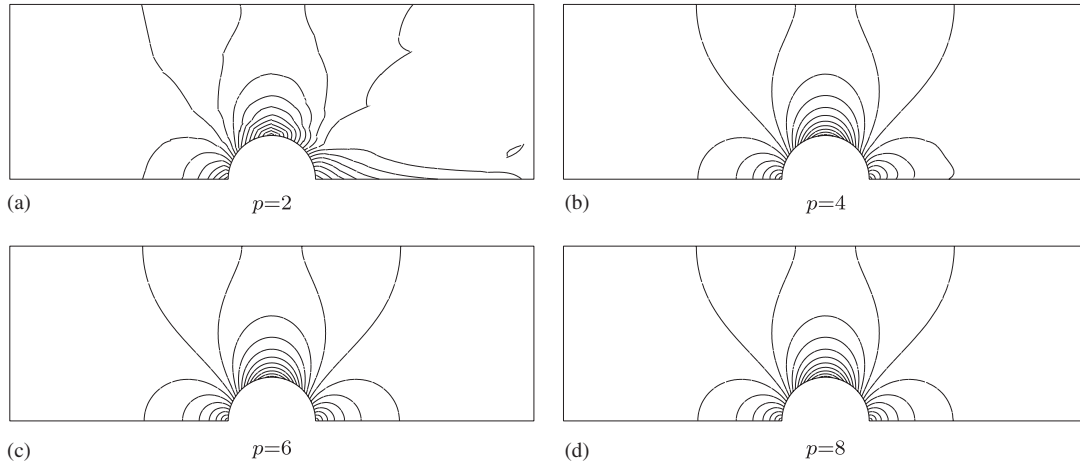


Figure 14. Mach number isolines for isoparametric FE in the fine mesh.

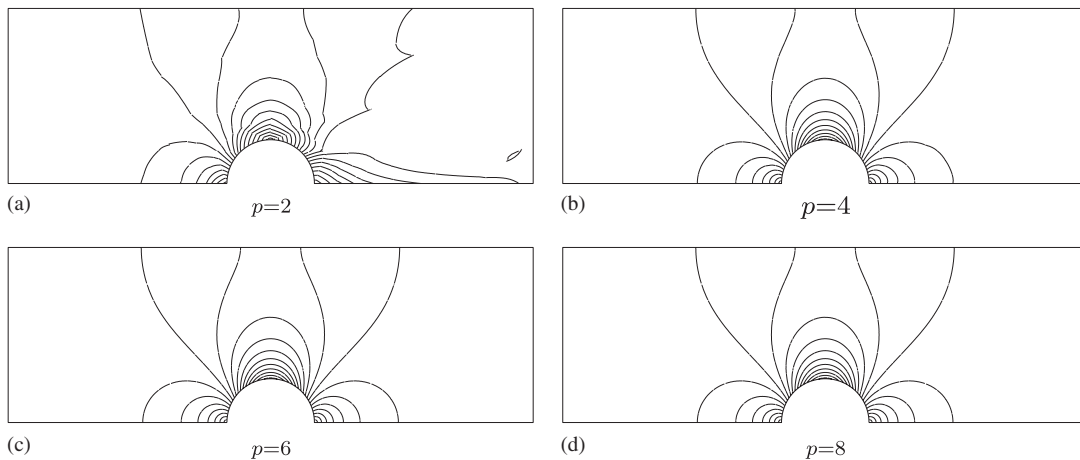


Figure 15. Mach number isolines for NEFEM in the fine mesh.

strategy must be adopted. For instance, an error below 10^{-4} can be achieved with the second mesh shown in Figure 11. Note that for NEFEM with the coarse mesh, the error with $p=8$ is already 4×10^{-5} , more than one order of magnitude more precise than isoparametric FE with the same number of degrees of freedom. Moreover, if the second mesh is considered, the NEFEM error with $p=8$ is 10^{-6} , two orders of magnitude more precise than standard isoparametric FE. It is also worth noting that with an approximation of degree $p=8$, similar accuracy is obtained using isoparametric FE in the fine mesh and NEFEM in the coarse one, that is, NEFEM requires four times fewer degrees of freedom.

Finally, it is important to recall that in a FE adaptive process, see [36], the computational mesh must be locally refined to properly capture both the solution and the geometry, whereas in a

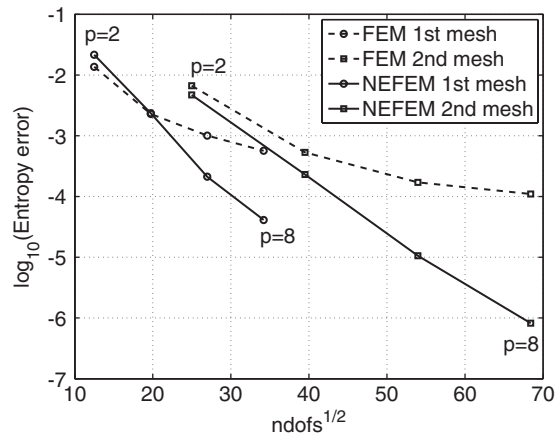


Figure 16. Entropy error on the circle surface for p -refinement.

NEFEM context the adaptive process is controlled only by the complexity of the solution, with no dependence on the geometrical complexity of the domain, therefore reducing the necessary number of degrees of freedom to achieve a desired accuracy.

6. CONCLUDING REMARKS

NEFEM combined with a DG formulation is proposed for the numerical solution of compressible Euler equations. A classical test for inviscid flow solvers is considered to evaluate the accuracy and efficiency of the proposed methodology in front of isoparametric FE. The proper imposition of the wall boundary condition in NEFEM allows accurate computations even with a linear approximation of the solution. With isoparametric FE, it is not possible to converge to the correct solution using linear approximation, even if the mesh is drastically refined near curved boundaries. Moreover, under p -refinement, exponential convergence is achieved with NEFEM and not for isoparametric FE.

Numerical results demonstrate that NEFEM is a powerful method for the solution of Euler equations of gas dynamics, more efficient than classical isoparametric FE. The extra cost of NEFEM, due to the numerical integration over elements along the NURBS boundary, is surpassed by the important saving in the number of degrees of freedom. More precisely, NEFEM provides similar accuracy than isoparametric FE using between four and ten times fewer degrees of freedom.

REFERENCES

1. Bassi F, Rebay S. High-order accurate discontinuous finite element solution of the 2D Euler equations. *Journal of Computational Physics* 1997; **138**(2):251–285.
2. van der Ven H, van der Vegt JJW. Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows. II. Efficient flux quadrature. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**(41–42):4747–4780.
3. Xue D, Demkowicz L. Control of geometry induced error in hp finite element (FE) simulations. I. Evaluation of FE error for curvilinear geometries. *International Journal of Numerical Analysis and Modelling* 2005; **2**:283–300.

4. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**(39–41):4135–4195.
5. Krivodonova L, Berger M. High-order accurate implementation of solid wall boundary conditions in curved geometries. *Journal of Computational Physics* 2006; **211**(2):492–512.
6. Sevilla R, Fernández-Méndez S, Huerta A. NURBS-enhanced finite element method (NEFEM). *International Journal for Numerical Methods in Engineering* 2007; DOI: 10.1002/nme.2311.
7. Dadone A, Grossman B. Surface boundary conditions for the numerical solution of the Euler equations. *AIAA Journal* 1994; **32**(2):285–293.
8. Barth T. Simplified numerical methods for gas dynamics systems on triangulated domains. *Ph.D. Thesis*, Department of Aeronautics and Astronautics, Stanford University, 1998.
9. Wang ZJ, Sun Y. A curvature-based wall boundary condition for the Euler equations on unstructured grids. *Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit*, Nevada. AIAA: New York, 2002.
10. Wang ZJ, Liu Y. Extension of the spectral volume method to high-order boundary representation. *Journal of Computational Physics* 2006; **211**:154–178.
11. Piegel L, Tiller W. *The NURBS Book*. Springer: London, 1995.
12. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering* 2000; **47**(12):2039–2072.
13. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. Wiley: Chichester, 2002.
14. Anderson JD. *Modern Compressible Flow: With Historical Perspective*. McGraw-Hill: New York, 1982.
15. Hirsch C. *Numerical Computation of Internal and External Flows. Vol. 2: Computational Methods for Inviscid and Viscous Flows*. Wiley: Chichester, 1988.
16. Laney CB. *Computational Gasdynamics*. Cambridge University Press: Cambridge, 1998.
17. Cockburn B. Discontinuous Galerkin methods for computational fluid dynamics. *Encyclopedia of Computational Mechanics*, Chapter 4, vol. 3. Wiley: New York, 2004.
18. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics* (2nd edn). Springer: Berlin, 1997.
19. Leveque R. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press: Cambridge, 2002.
20. Kuzmin D, Möller M. Algebraic flux correction. II. Compressible Euler equations. *Flux-corrected Transport: Principles, Algorithms, and Applications*. Springer: Berlin, 2005.
21. Cockburn B, Shu CW. TVB Runge–Kutta local projection discontinuous Galerkin finite-element method for conservation-laws. II. General framework. *Mathematics of Computation* 1989; **52**(186):411–435.
22. Dolejší V, Feistauer M. A semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow. *Journal of Computational Physics* 2004; **198**(2):727–746.
23. Persson P, Peraire J. An efficient low memory implicit DG algorithm for time dependent problems. *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*, Nevada. AIAA: New York, 2006.
24. Gottlieb S, Shu CW. Total variation diminishing Runge–Kutta schemes. *Mathematics of Computation* 1985; **67**(221):73–85.
25. Atkins HL, Shu CW. Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations. *AIAA Journal* 1998; **36**(5):775–782.
26. Hillewaert K, Chevaugnon N, Geuzaine P, Remacle J. Hierarchic multigrid iteration strategy for the discontinuous Galerkin solution of the steady Euler equations. *International Journal for Numerical Methods in Fluids* 2006; **51**:1157–1176.
27. Hesthaven JS, Warburton T. Nodal high-order methods on unstructured grids. I. Time-domain solution of Maxwell’s equations. *Journal of Computational Physics* 2002; **181**(1):186–221.
28. Chen Q, Babuška I. Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Computer Methods in Applied Mechanics and Engineering* 1995; **128**(3–4):405–417.
29. Chen Q, Babuška I. The optimal symmetrical points for polynomial interpolation of real functions in the tetrahedron. *Computer Methods in Applied Mechanics and Engineering* 1996; **137**:89–94.
30. Taylor MA, Wingate BA, Vincent RE. An algorithm for computing Fekete points in the triangle. *SIAM Journal on Numerical Analysis* 2000; **38**(5):1707–1720.
31. Sevilla R, Fernández-Méndez S. Numerical integration for the 2D NURBS-enhanced finite element method, 2007, submitted.
32. Luo H, Baum JD, Löhner R. A fast, p -multigrid discontinuous Galerkin method for compressible flows at all speeds. *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*, Nevada. AIAA: New York, 2006.
33. Cockburn B, Karniadakis GE, Shu CW. The development of discontinuous Galerkin methods. In *Discontinuous Galerkin Methods*, Cockburn B, Karniadakis GE, Shu C-W (eds). Lecture Notes in Computational Science and Engineering, vol. 11. Springer: Berlin, 2000; 3–50.

34. Szabó B, Babuška I. *Finite Element Analysis*. Wiley: New York, 1991.
35. Szabó B, Düster A, Rank E. The p -version of the finite element method. *Encyclopedia of Computational Mechanics*, Chapter 5, vol. 1. Wiley: New York, 2004.
36. Huerta A, Rodríguez-Ferran A, Díez P, Sarrate J. Adaptive finite element strategies based on error assessment. *International Journal for Numerical Methods in Engineering* 1999; **46**:1803–1818.