

A Framework for Curved Boundary Representation in 2D Discontinuous Galerkin Euler Solvers

B. Baghapour¹ and V. Esfahanian^{2†} and A. Nejat³

¹ School of Mechanical Engineering, College of Engineering, University of Tehran, North Kargar Ave., P. O. Box 14395-1335, Tehran, Iran

†Corresponding Author Email: evahid@ut.ac.ir

(Received September 26, 2013; accepted December 8, 2013)

ABSTRACT

Finite-element based CFD solvers like the family of Discontinuous Galerkin (DG) solvers suffer severely from inaccurate boundary reconstruction. In this matter, developing an accurate and flexible strategy is highly demanded to provide high-order curved boundary representation in DG simulations. In this paper, a general framework is introduced to design the curved elements in discontinuous Galerkin finite-element (DGFEM) simulations. The aim is to connect the boundary to the surrounding mesh by defining an appropriate set of basis functions which deliver the curvature information inside the mesh region adjacent to the boundary. This information is then used in flux integral calculations. The proposed framework is applied in Lagrangian and Hermitian boundary representations. The efficiency of the method is analyzed for compressible inviscid flow test cases using the discontinuous Galerkin scheme. It is illustrated that using the curved-side elements in the present approach, is adequate to reduce the artificial entropy generation near the boundaries. This leads to the simulations with the desired order of accuracy. The results show a well consistency in h/p -refinement which advocates the use of the proposed approach in high-order CFD simulations.

Keywords: Curvature distribution, Geometric transformation, Euler equations, Discontinuous Galerkin

NOMENCLATURE

c	degree of freedom	n_b, n_t	number of DOF for curve representation
C_D	drag coefficient	P	pressure
C_L	lift coefficient	q	convergence rate
C_p	pressure coefficient	Q	conservative variable
e_t	total internal energy	R	nonlinear residual
E_k	ratio of errors	R_i	inner radius in supersonic vortex problem
F	inviscid flux	\vec{r}	position vector in Cartesian coordinates
f, g	flux components	\vec{r}_b	position vector at the curved boundary
H	refers to Hermitian curved elements	S	entropy error generation
h	numerical convective flux	\bar{s}	set of geometric degrees of freedom
h_t	total enthalpy	t	time
J	matrix containing the metrics	u, v	Cartesian velocity components
$ J $	volumetric Jacobian	U_i	inner wall velocity of the supersonic vortex
$ \hat{J} $	surface Jacobian	V	velocity magnitude
L	refers to Lagrangian curved elements	x, y	Cartesian (physical) coordinates
l_s	Lobatto basis functions	X_k	ratio of mesh sizes
M	mass matrix	$x_{\bar{\xi}}, y_{\bar{\xi}}$	metric along the first reference coordinate
M_i	inner wall Mach number of supersonic vortex	x_{η}, y_{η}	metric along the second reference coordinate
M_{∞}	free-stream Mach number	α	angle of attack
N	shape functions for the curved-side	γ	ratio of specific heats
n_p	Number of DOFs for solution reconstruction		

n_e	Number of elements	Δt	time step
ξ	first reference (local) coordinates	ε	entropy
ϕ, Ψ	basis function	η	second reference (local) coordinates
ω	Hermite basis functions	σ	element face in the mapped space
ρ	density	κ	discrete element in the mapped space
ρ_i	exact density at the inner wall	Ω	discrete element in the physical space
ρ_e	exact density in supersonic vortex problem	$\partial\Omega$	element face in the physical space

1. INTRODUCTION

The main goal of employing the high-order discretization methods is to capture a wide range of details by a minimized mesh density (Hirsh 1975, Ekaterinaris 2005, Ollivier-Gooch *et al.* 2009). However, inaccurate boundary representation ruins drastically the desired order of the numerical accuracy expected from the high-order methods (Var der Vedgt and Van der Ven 2002).

Several attempts have been made for high-order surface reconstruction by considering curvature information of the immersed bodies. Various non-physical solutions have been observed in inviscid flows even in high-order discretizations mainly due to low-order boundary representation. Bassi and Rebay (1997) declared the necessity of the curved representation in high-order discontinuous Galerkin (DG) methods. They reported non-physical vortex generation over the circular cylinder in inviscid flow with linear boundary segmentation. They showed considerable improvement in the accuracy of the simulations using Lagrangian curved-side elements on the body surface. Following this issue, many researches have been conducted on efficient and accurate boundary representation schemes to overcome the mentioned drawbacks and their major attention is the inviscid flow (Euler) test cases.

Wang & Liu (2006) introduced simplified curved elements for the Spectral Volume (SV) methods. Moreover, the effect of curvature with entropy-based boundary condition was studied recently by Kannan and Wang (2011). Fidkowski and Darmofal (2007) developed a cell cutting procedure for a non-conforming mesh. In order to find curved boundary coordinates, cubic intersection problems should be solved. Karimian and Ardakani (2011) studied the ability of immersed boundary method on Cartesian grid for 2D compressible inviscid flows for stationary and moving objects.

In addition, Vymazal *et al.* (2011) studied the effect of curved elements for residual distribution schemes by portioning each element into linear sub-elements to reuse the original simple form using standard sub- or iso-parametric transformation. They showed that the expected order of accuracy would be achieved only with accurate approximation of the geometry by the curved elements. Sevilla *et al.* (2008) used NURBS (Non-Uniform Rational B-Splines) to generate the curved boundary for the finite element flow simulations. In addition, Gao *et al.* (2010) introduced the Bezier mapping technique for the C^1 continuous curved boundaries for high-order Lifting Collocation Penalty (LCP) methods (Wang and Gao 2009).

In Recent years, a group of researchers makes effort to investigate the effect of artificial entropy generation due to boundary reconstruction. In this matter, smooth inviscid flows are chosen to eliminate the physical source of entropy generation like viscosity and shock wave in order to focus on the effect of boundary representation on the accuracy of the fluid flow simulations.

At a first glance, constructing curved elements is related to a specific surface generation technique. It takes some intermediate steps to provide necessary information from the initial data. Moreover, the CFD researchers are usually confined to the features provided by the geometry modeling and grid generation tools. In this matter, designing a generalized definition of the curved elements to conveniently exploit the geometric information in high-order CFD simulations seems to be highly demanded.

Constructing traditional types of curved elements such as those used by Bassi and Rebay (1997) requires inverting a system of equation based on information about available node coordinates. In this matter, curved boundary representation as well as analytical surface function should be directly considered in the CFD code. In many situations, this may not possible since prevalent grid generation schemes do not directly handle the curvature reconstruction. This reduces the flexibility of a solver for accurate simulations.

The aim of this paper is to streamline the procedure by introducing a tensor product of low-order and high-order polynomials representing the curved elements. In fact, there is no need to define an integrated 2D curved element space as in the case of traditional strategies. In other words, the tensor production fulfills the requirements of generating the 2D curved space by distributing the curvature data provided by the high-order polynomials. Therefore, the curved boundaries can be reconstructed separately by any surface generation techniques and the effect of curvature can be implemented conveniently in domain discretization section of the CFD codes. This procedure brings a flexibility to define a framework for exploiting a wide range of reconstruction techniques in recent CFD methods like the discontinuous Galerkin schemes.

A general framework for constructing curved-side elements is introduced for CFD applications. The main idea is to use directly the surface reconstruction techniques, e.g. Lagrangian and Hermitian methods, to provide curvature information for the geometric transformation inside the computational domain. This is realized by

blending the high-order polynomials as the curvature constructor with the linear basis functions as the data distributor in the computational field. The consequence is the generation of P1Pk curved elements where k is associated with the order of the boundary approximation.

The paper is organized as follows. In Section 2, a brief description of the DG discretization method is presented. The procedure of constructing the curved-side elements is explained in Section 3. Finally, the efficiency of the proposed boundary representation is examined in Section 4.

2. DISCONTINUOUS GALERKIN

The two-dimensional Euler equations in conservative form are written as:

$$\frac{\partial Q}{\partial t} + \nabla \cdot F(Q) = 0 \tag{1}$$

where Q is the set of conservative variables and $F = (f, g)$ is flux vector. The conservative field variables and the Cartesian flux components are:

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{pmatrix}, f = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho u h_t \end{pmatrix}, g = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + P \\ \rho v h_t \end{pmatrix} \tag{2}$$

The total enthalpy per unit mass of the perfect gas is defined as $h_t = e + V^2/2 + P/\rho$ where the velocity magnitude of flow field calculated by $V = \sqrt{u^2 + v^2}$. The static pressure of the working fluid is then obtained by $P = \rho(\gamma - 1)(e_t - V^2/2)$ where γ indicates the ratio of the specific heats of the fluid.

DG discretization on each element in weak statement form is expressed by:

$$\int_{\Omega} \psi_j \frac{\partial Q}{\partial t} d\Omega + \oint_{\partial\Omega} \psi_j h dS - \int_{\Omega} \nabla \psi_j \cdot F(Q) d\Omega = 0 \tag{3}$$

where ψ_j is the weight function related to the j -th equation of the weak solution in the Galerkin approach. The second term in Eq. (3) represents the numerical face flux balance for each element $h = h(Q^-, Q^+)$ where Q^\pm are the trace of the conservative variables on both side of a common face of the two adjacent elements. Therefore, appropriate numerical flux function is needed to compute the convective flux passing through the faces of the element. In this work, the Roe (1981) numerical flux function is employed. In order to solve the system of DG equations in a general framework, the Eq. (3) is mapped onto the space of reference element as follows:

$$\int_{\Omega} \psi_j \frac{\partial Q}{\partial t} |J| d\kappa + \oint_{\partial\Omega} \psi_j h |\tilde{J}| d\sigma - \int_{\Omega} \nabla \psi_j \cdot F(Q) |J| d\kappa = 0 \tag{4}$$

where \mathcal{K} is the reference space for quadrilateral elements, $|J|$ is the volumetric Jacobian and $|\tilde{J}|$ is the surface Jacobian between the real and reference geometric spaces on each integration point. The conservative variables are then simulated using the finite element approach as follows:

$$Q^m = \sum_{i=1}^{n_p} c_i^m \phi_i \tag{5}$$

where c_i^m is the i -th DOF related to m -th component in vector of conservative field variables. In addition, n_p is related to the shape functions used in DG simulation. The polynomial set ϕ_j can be regarded as nodal or hierarchical polynomial basis functions. It is worth mentioning that in the Galerkin approach the set of basis functions, $\{\phi\}$, and the weight functions, $\{\psi\}$, are taken to be equal.

In this work, hierarchical edge and bubble functions developed by Solin (2004) are used. Table 1 presents the hierarchical basis functions including linear element, edge and bubble (Lobatto bases) functions.

Table 1 Hierarchical Lobatto shape functions up to fourth-order accuracy.

Function Type	Formula
Linear element	$\frac{1}{4}(1 \pm \xi)(1 \pm \eta)$
Lobatto basis (l_s)	$\frac{1}{2}(1 \pm \xi), \sqrt{\frac{3}{8}}(\xi^2 - 1), \sqrt{\frac{5}{8}}\xi(\xi^2 - 1)$
Edge functions	$l_0(\xi)l_s(\eta), l_1(\xi)l_s(\eta)$ $l_1(\eta)l_s(\xi), l_0(\eta)l_s(\xi)$ $2 \leq s \leq p$
Bubble functions	$l_m(\xi)l_n(\eta), 2 \leq m, n \leq p$

The overall semi-discretized relation of Euler equations using explicit discontinuous Galerkin method leads to the following formula:

$$\frac{dQ}{dt} = M^{-1}R(Q) \tag{6}$$

where R is the numerical residual including interior and convective flux terms and M is the mass matrix related to each element. The employed time discretization is the stable TVD Runge–Kutta scheme developed for discontinuous Galerkin method by Cockburn and Shu (1998):

$$Q^{(1)} = Q^n + \Delta t M^{-1}R(Q^n)$$

$$Q^{(2)} = \frac{3}{4}Q^n + \frac{1}{4}[Q^{(1)} + \Delta t M^{-1}R(Q^{(1)})]$$

$$Q^{(3)} = \frac{1}{3}Q^n + \frac{2}{3}[Q^{(2)} + \Delta t M^{-1}R(Q^{(2)})] \tag{7}$$

The boundary fluxes required for the solid wall are computed with ghost-state approach described by Bassi and Rebay (1997). The numerical integration procedure is performed by Gaussian quadrature rule. The required number of quadrature points is

chosen to integrate exactly on the reference quadrilateral elements of the order $2p$ where p denotes the solution accuracy (Bassi and Rebay 1997). In the case of linear, quadratic, and cubic shape functions, the volume integrals are evaluated using four, nine, and sixteen points and the boundary integrals are evaluated using two, three, and four points, respectively.

3. CURVED ELEMENTS

Many high-order finite-element-based methods are involved in calculating the flux integrals. In a computational domain with a general mesh and curved boundaries, a geometrical transformation is required to map all the discrete elements into a unique reference element. Therefore, a systematic procedure is required to construct an appropriate reference element based on the mesh point and the curvature data. In general CFD solvers, this procedure should be accomplished in a way that brings an acceptable order of accuracy in flow simulations, and also does not append a considerable complexity to the code infrastructure.

A common procedure to reconstruct the boundary curvature in CFD simulations has the following steps:

1. Discretizing the computational domain by a desirable mesh generation technique.
2. Reconstructing the curved boundary with local interpolating polynomials, e.g. Lagrange, Hermite, NURBS and etc.
3. Constructing the curved-side elements based on available geometric data.
4. Obtaining the geometric transformation and the normal vectors to the curved boundary to be used in the flux integral calculations.

Nowadays, the family of traditional Lagrangian curved elements is widely used in high-order DG and FV simulations. However, the procedure of accuracy adaptation requires additional correction step to the master element which increases the level of complexity in underlying source code (Li 2006). On the other hand, there are several situations where the exact geometry is not available and the curvature definition should be carried out e.g. by approximating the metric in the spline reconstruction (Esfahanian *et al.* 2011) or by defining the control nodes in the Bezier reconstruction process (Gao *et al.* 2010).

3.1 General Framework

In high-order boundary representation, the boundary nodes should be connected together by local defined curvatures. In finite-element based methods, the set of boundary-adjacent curved-side elements should be define to connect the curved boundary, generated by surface capturing techniques, to the whole system of mesh points.

In this paper, a three-step procedure for considering the curved boundary in high-order DGFEM solvers is presented as follows:

1. Define the curvature: Detect the mesh points on the linear segmented (low-order) boundary. Generate the local curvature between each pair of boundary nodes using a conventional surface generation technique. This generates the first reference coordinate, $\xi \in [-1,1]$.
2. Define the connecting region: Defining the high-order basis functions, representing the curvature, from the boundary to the linear surrounding mesh. This introduces the second reference coordinate, $\eta \in [-1,1]$.
3. Define the curved-side elements: Generate the 2D space for the boundary-adjacent curved-side elements. This is achieved by distributing the surface and volume quadrature points according the 2D geometric transformation for the flux integral calculations.

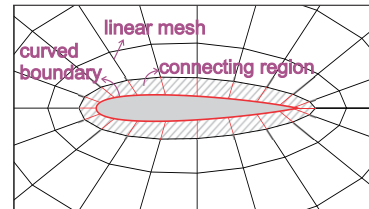


Fig. 1. Connecting the curved boundary to the linear mesh in the proposed general framework.

Figure 1 demonstrates how the connecting region, including the curved-side elements, connects the curved boundary to the surrounding linear mesh. The whole procedure of constructing the curved-side elements is also depicted in Fig. 2.

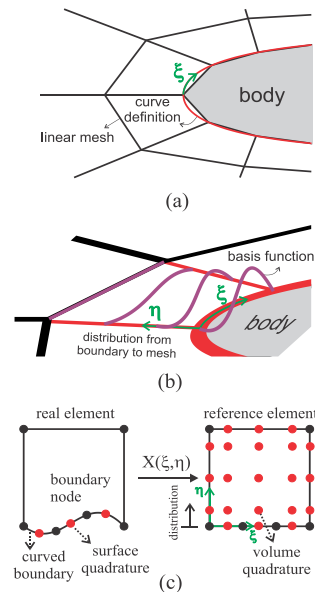


Fig. 2. Constructing the connecting space by the curved-side elements in a three-step procedure.

Table 2 one-dimensional interpolation polynomials for curved boundary reconstruction

1D geometric basis functions	Formulation
Linear bases, $P=1$	$\omega_1 = \frac{1}{2}(1-\xi), \omega_2 = \frac{1}{2}(1+\xi)$
Lagrange, $P=2$	$\omega_1 = -\frac{1}{2}\xi(1-\xi), \omega_2 = (1-\xi)(1+\xi)$ $\omega_3 = \frac{1}{2}\xi(1+\xi)$
Lagrange, $P=3$	$\omega_1 = -\frac{9}{16}\left(\frac{1-\xi^2}{9}\right)(1-\xi)$ $\omega_2 = \frac{27}{16}\left(\frac{1-\xi}{3}\right)(1-\xi^2)$ $\omega_3 = \frac{27}{16}\left(\frac{1+\xi}{3}\right)(1-\xi^2)$ $\omega_4 = -\frac{9}{16}\left(\frac{1-\xi^2}{9}\right)(1+\xi)$
Hermite, $P=3$	$\omega_1 = \frac{1}{2} - \frac{3}{4}\xi + \frac{1}{4}\xi^3$ $\omega_2 = \frac{1}{2} + \frac{3}{4}\xi - \frac{1}{4}\xi^3$ $\omega_3 = \frac{1}{4} - \frac{1}{4}\xi - \frac{1}{4}\xi^2 + \frac{1}{4}\xi^3$ $\omega_4 = -\frac{1}{4} - \frac{1}{4}\xi + \frac{1}{4}\xi^2 + \frac{1}{4}\xi^3$

In a geometry description of a curved boundary in 2D domains, a unidirectional interpolation is defined between each boundary segments as follows:

$$\vec{r}_b(\xi) = \sum_{j=1}^{n_b} \omega_j(\xi) \vec{s}_j^b \quad (7)$$

where ξ is the local direction along with the boundary (Fig. 2(a)) which is exploited as the first reference coordinate the procedure of constructing the curved-side elements. The set of the geometry information, including the boundary points, \vec{r}_b as well as the metrics $\vec{r}_{b,\xi}$ construct the set of geometric degrees of freedom at the curved boundary denoted by $\{\vec{s}^b\}$. It is worth mentioning that, reconstruction the curvature is the first step among the procedure of considering the curved geometry in the flow simulations.

The formulation of the unidirectional polynomials is presented in Table 2 for Lagrangian and Hermitian types. The linear polynomials are considered for the nodes which coincide with the curved boundary ($\omega_{1,2}$ for quadrilateral elements).

In addition, Fig. 3 shows the distribution of the 1D polynomials for different basis functions constructing the first reference coordinate at the curved boundary.

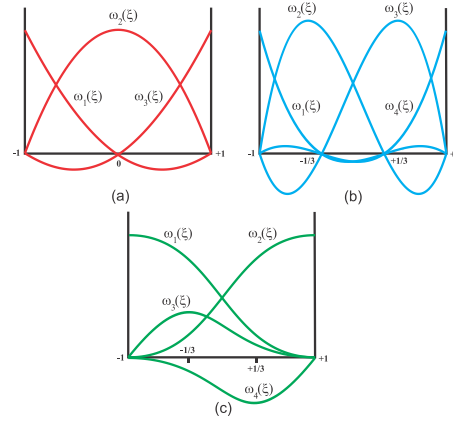


Fig. 3. Distribution of 1D geometric basis functions in the reference frame, (a) Lagrange, $P=2$, (b) Lagrange, $P=3$, (c) Hermite, $P=3$.

The second step is to connect the curved geometry, defined by Eq. (7), to the surrounding mesh points (Fig. 2(b)). In this matter, the interval between to spaces (curved boundary and linear mesh) can be filled by introducing the intermediate space of the curved-side elements (Fig. 2(c)). Since all the mesh segments, except the boundary ones, consist of straight lines, a proper linear distribution of the geometric data can properly convey the information from the curved boundary to the mesh points. Having this idea, any surface generation techniques can be directly join to this procedure and a variety of curved-side elements are produced. The introduced frame work can be cast by the following relation:

$$\vec{r}(\xi, \eta) = \sum_{j=1}^{n_r} N_j(\xi, \eta) \vec{s}_j \quad (8)$$

where $\{\vec{s}\}$ is the set of total geometric DOFs. Moreover, the geometrical two-dimensional shape functions have the following formula:

$$N_j(\xi, \eta) = \omega_k(\xi) \tau^\pm(\eta) \quad (9)$$

where ω_k is the unidirectional interpolation polynomials and $\tau^- = l_0, \tau^+ = l_1$ are the linear Lobatto shape functions defined at $\eta = -1, 1$. In this case, the high-order curvature reconstruction is imposed at the curved boundary while the linear property is preserved on the surrounding mesh points.

3.2 Obtaining the Geometric DOFs

After constructing the curved-side element, the required geometric DOFs should be provided according to each type of boundary representation. In Lagrangian based curved elements, the extra nodes are required. These are obtained by locating them between each boundary interval using the analytical geometry profile. Therefore, having the geometry function, the extra DOFs are calculated at the interval $x \in [x_a, x_b]$ as follows:

$$x(\xi) = \frac{x_a + x_b}{2} + \xi \left(\frac{x_b - x_a}{2} \right), -1 \leq \xi \leq +1$$

$$y(\xi) = f(x(\xi))$$
(10)

where f is the analytical profile. For the Hermitian type with quadrilateral elements, the local metrics at boundary nodes are required which can be obtained as follows:

$$\frac{\partial x(\xi)}{\partial \xi} = \frac{\partial x}{\partial s} \frac{\partial s}{\partial \xi}, \frac{\partial x}{\partial s} = (1 + y'^2)^{0.5}$$

$$\frac{\partial y(\xi)}{\partial \xi} = \frac{\partial y}{\partial s} \frac{\partial s}{\partial \xi}, \frac{\partial y}{\partial s} = y' (1 + y'^2)^{0.5}$$
(11)

where s in the arc length of the interval and $\partial s / \partial \xi$ is obtained by averaging the discrete form, $\Delta s / \Delta \xi$ at each boundary nodes belonged to the interval. The metrics can be obtained by the analytic profile or can be approximated using finite-difference methods (Esfahanian *et al.* 2011). The list of the required DOFs for quadrilateral curved-side elements are presented in Table 3.

Table 3 Required geometric DOFs for element types of curved-side elements.

Curved element	Required DOFs
Lagrangian PIP2L, PIP3L	At corners: $\bar{r}_j, j = 1, \dots, 4$ Extra nodes: $\bar{r}_k, k = 4 + m, (m = 1, 2)$
Hermitian PIP3H	At corners: $\bar{r}_j, j = 1, \dots, 4$ metrics at corners: $\frac{\partial \bar{r}_1}{\partial \xi}, \frac{\partial \bar{r}_2}{\partial \xi}$

4. NUMERICAL TESTS

In this section, two compressible smooth flow test cases are considered to examine the efficiency of the proposed approach.

4.1 Supersonic Vortex Flow

The shock-free supersonic vortex flow test case is considered as an isentropic flow in a circular channel. The geometry of the channel consists of two concentric circular arcs which $R_1 = 2.0$ and $R_2 = 3.0$ and the inner inlet Mach number is $M_{in} = 2.0$. Figure 4 demonstrates the mesh layout (h -refinement). The number of mesh points is double in each direction at each step. Figure 5(a) and 5(b) show the L1 error of the solution for different mesh sizes. It can be seen from this figure that using curved-side elements is crucial to achieve the desired rate of convergence by increasing the number of mesh points in the h -refinement process (Nejat 2007). The convergence rate, q , is the average value of the slope defined by:

$$q_k = \frac{\log_{10}(E_k)}{\log_{10}(X_k)} \tag{12}$$

where $X_k = \Delta x_k / \Delta x_{k+1}$ is the ratio of mesh sizes, and $E_k = \|E\|_{1,k} / \|E\|_{1,k+1}$ is the ratio of L1 norm of the error in density defined in each k -th level of mesh refinement as follows:

$$L_i := \|E\|_1 = \frac{1}{n_e} \left(\sum_{i=1}^{n_e} |\bar{E}_i| \right)$$

$$\bar{E}_i = \frac{1}{\Omega_i} \int_{\Omega_i} (\rho - \rho_e) d\Omega$$
(13)

where ρ_e is the exact solution obtained by the following formula (Aftosis *et al.* 1994):

$$\rho_e = \rho_i \left(1 + \frac{\gamma-1}{2} M_i^2 \left(1 - \frac{R_i^2}{r^2} \right) \right)^{\frac{1}{\gamma-1}}$$

$$U_i = M_i \rho_i^{\frac{\gamma-1}{2}}, U = \frac{U_i R_i}{r}, P = \frac{\rho^\gamma}{\gamma}$$
(14)

According to Fig. 5, reducing the numerical error in DG is tightly related to the accuracy of boundary representation. This issue is more illustrative for higher accuracy, e.g. the cubic (fourth-order) solution.

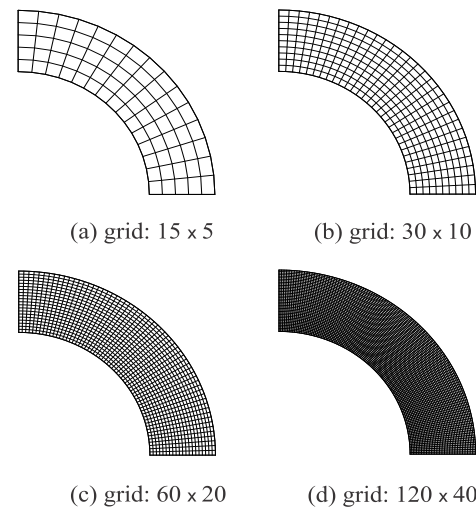


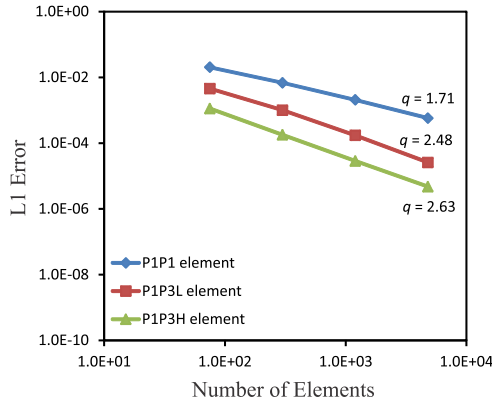
Fig. 4. Mesh refinement for supersonic vortex flow.

4.2 Subsonic Flow around 2D Airfoils

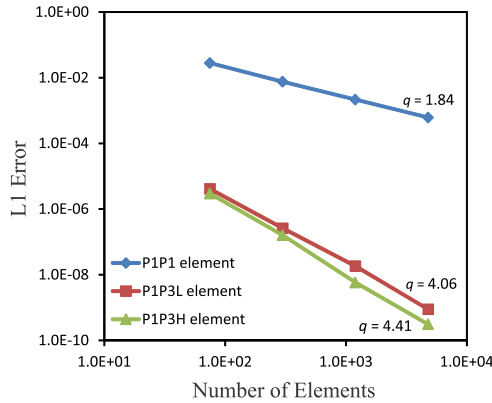
Subsonic flows around 2D airfoils are considered to testify the efficiency of the proposed approach in more general cases. The outer boundary is placed 20 chords far from the airfoil. A structured grid with quadrilateral elements (O-type grid with 1400 elements: 70×20) is considered.

For the symmetric NACA 0012 airfoil, the free-stream Mach number is $M=0.63$ and the angle of attack is 2.0° . For the non-symmetric NACA 4412, the free-stream Mach number is $M=0.2$ and the angle of attack is 8.0° .

Figure 6 shows the mesh layout near the leading and trailing edges of the NACA 0012. Figure 7 and 8 demonstrates the effect of using curved-side elements in reducing the generated entropy error around the leading edge and trailing edge of the NACA 0012 airfoil for linear (second-order) and cubic (fourth-order) solutions. It can be seen from these figures that increasing the order of boundary representation reduces noticeably the entropy error especially for cubic (fourth-order) solution.



(a) Linear solution (P1)



(b) Cubic solution (P3)

Fig. 5. Error Analysis for supersonic vortex flow

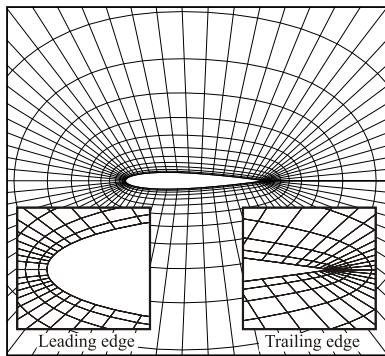


Fig. 6. Mesh layout around NACA 0012

Table 4 compares the norm of the entropy error, drag coefficient C_D , and lift coefficient C_L around

the NACA 0012 airfoil for the family of P1PK elements. It can be concluded that the ability to reduce the error is more achieved by P1P3H element since the C^1 condition (continuity of the slope at the boundary nodes) is preserved by applying the Hermitian surface reconstruction. This issue is better illustrated in Fig. 9 by demonstrating the distribution of the entropy error $S = (\varepsilon - \varepsilon_\infty) / \varepsilon_\infty$ where the entropy is defined as $\varepsilon = P / \rho^\gamma$. In this figure, the entropy generation over NACA 0012 is compared for P1P2L, P1P3L, and P1P3H elements. According to this figure, P1P3H has better performance in reducing the entropy generation.

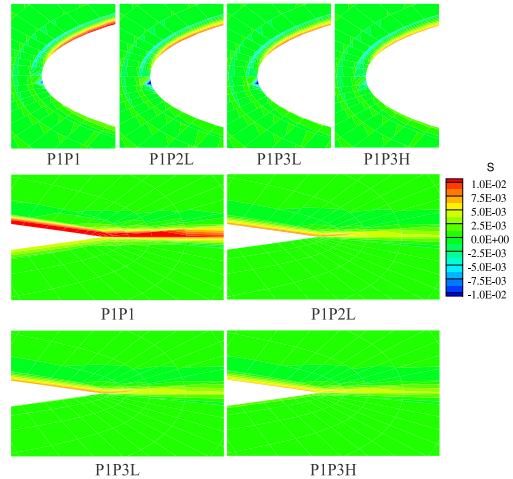


Fig. 7. Entropy Error over NACA 0012 with different curved elements for the linear solution

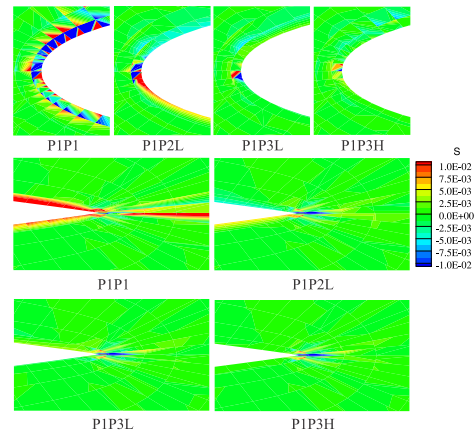


Fig. 8. Entropy Error over NACA 0012 with different curved elements for the cubic solution

Figure 10 compares the pressure field around the leading edge of the NACA 0012 airfoil for low-order linear segmentation and the locally curved segmentation using P1P3H elements. It is clear that using high-order surface reconstruction yields to high quality results especially for the cubic (fourth-order) solution.

Figure 11 compares the distribution of the pressure coefficient C_p around the NACA 0012 airfoil for different curved elements. The results show a well consistency with the previous works of DG solution of Krivodonova-Berger (2006) and the spectral difference solution of Liang *et al.* (2009).

Table 4 Effect of curved elements in DG simulation for NACA 0012 with 1400 elements, 20 chords farfield. (FVM: The finite volume results of Nejat 2007 with 1245 cells, 25 chords farfield)

Entropy L_1 Error			
Cases	Linear	Quadratic	Cubic
PIP1	0.0274595	0.0209295	0.0286842
PIP2L	0.0120888	0.0102421	0.0113728
PIP3L	0.0119507	0.0088419	0.0036157
PIP3H	0.0110249	0.0078582	0.0014382
Drag coefficient (C_D)			
Cases	Linear	Quadratic	Cubic
PIP1	0.0018324	0.0006625	0.0006572
PIP2L	0.0010574	0.0004067	0.0003744
PIP3L	0.0010521	0.0003813	0.0003726
PIP3H	0.0010243	0.0003807	0.0003691
FVM	0.00097664	0.00184889	0.00072576
Lift coefficient (C_L)			
Cases	Linear	Quadratic	Cubic
PIP1	0.309728	0.316488	0.316926
PIP2L	0.317204	0.321948	0.322372
PIP3L	0.317254	0.321996	0.322701
PIP3H	0.317639	0.322613	0.322715
FVM	0.322318	0.317393	0.322223

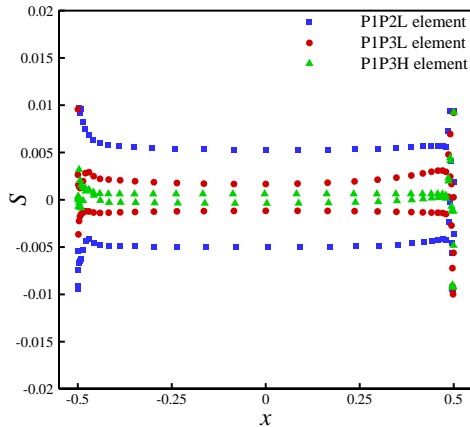


Fig. 9. Comparison of the entropy error generation over the surface of the NACA 0012 for different curved elements in the cubic solution

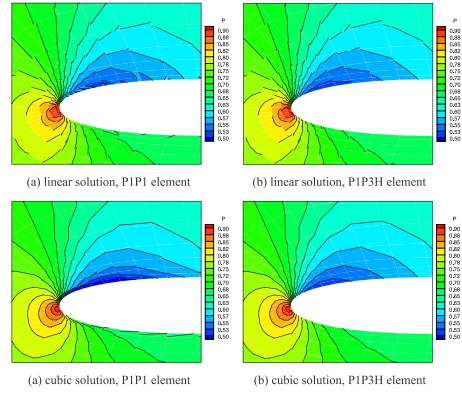


Fig. 10. Pressure contours over the NACA 0012 near leading edge for the linear segmentation and cubic curved-side elements in boundary representation

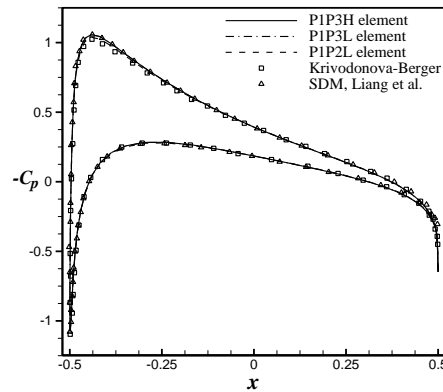


Fig. 11 Pressure coefficient distribution over the NACA 0012 for different boundary representations.

Figure 12 shows the mesh layout for the non-symmetric NACA 4412. Figure 13 demonstrates pressure contours over the airfoil for quadratic (third-order) solution. Figure 14 compares the solution with linear PIP1 and PIP3H curved element in the leading edge of the airfoil. It can be seen from this figure that the proposed curved element can properly simulation the inviscid flow over the NACA 4412. Figure 15 compares the distribution of the pressure coefficient C_p around the NACA 4412 airfoil with the work of Mason (2006). The result shows a well agreement with the mentioned work which emphasizes on the ability of the presented model for accurate boundary representation in a more general case.

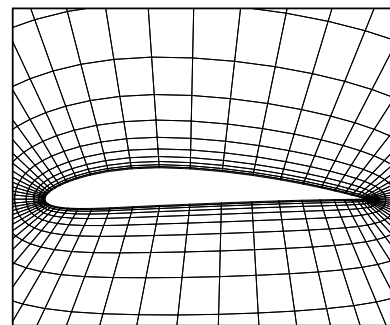


Fig. 12. Mesh layout around NACA 4412

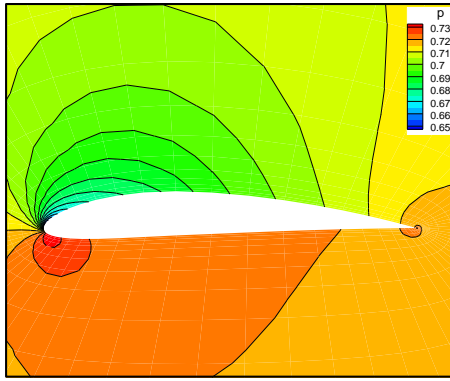
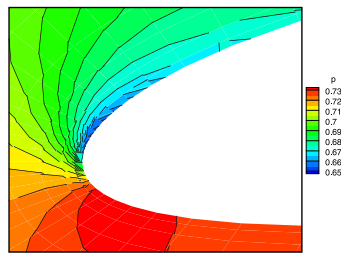
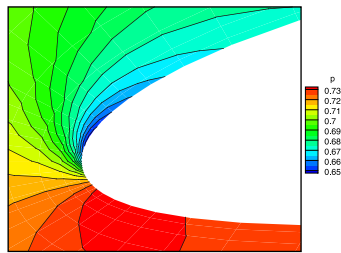


Fig. 13. Pressure contours over the NACA 4412 ($M_\infty = 0.2, \alpha = 8^\circ$)



(a) P1P1 element



(b) P1P3H element

Fig. 14. Pressure contours over the NACA 4412 near leading edge for linear and cubic boundary representations

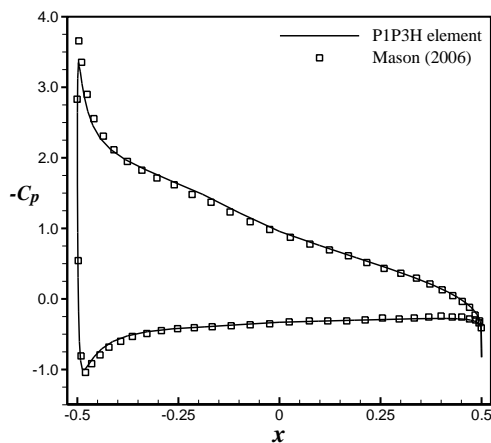


Fig. 15. Pressure coefficient distribution over the NACA 4412 with cubic P1P3H elements

5. CONCLUSIONS AND OUTLOOKS

A general framework to consider the curved boundary representation in compressible flow simulations is introduced using quadrilateral curved-side elements. A three-step procedure is developed for constructing the curved elements and providing the required geometrical data for the flux integral calculations in the reference space. The discontinuous Galerkin (DG) method is used to discretize the governing equations. The ability to reduce the entropy generation is studied for the family of P1PK elements including Lagrangian (P1P2L and P1P3L) and Hermitian (P1P3H) types. It is illustrated that P1P3H, which exploits the C^1 property, leads to the lower level of error generation near the curved boundary. The formulations are presented in a way that any other surface generation techniques can be directly used. In our future work, we will concentrate on extending the proposed approach to three-dimensional case with hexahedral and tetrahedral elements.

ACKNOWLEDGEMENTS

The authors would like to thank the Vehicle, Fuel and Environment Research Institute (VFERI) of University of Tehran for general support during this study.

REFERENCES

- Hirsh R. S. (1975). Higher order accurate difference solutions of fluid mechanics problems by a compact differencing technique. *Journal of Computational Physics* 19, 90-109.
- Ekaterinaris, J. A. (2005). High-order accurate, low numerical diffusion methods for aerodynamics. *Progress in Aerospace Sciences* 41,192-300.
- Ollivier-Gooch, C. and Nejat, A. and Michalak, K. (2009). Obtaining and verifying high-order unstructured finite volume solutions to the Euler equations. *AIAA Journal* 74, 2105-2120.
- Nejat, A. and Ollivier-Gooch, C. (2008). A high-order accurate unstructured finite volume Newton-Krylov algorithm for inviscid compressible flows. *Journal of Computational Physics* 227, 2582-2609.
- Van der Vegt, J. J. W. and Van der Ven, H. (2002). Slip flow boundary conditions in discontinuous Galerkin discretization of the Euler equations of gas dynamics. *Fifth World Congress on Computational Mechanics, Vienna, Austria, NLR-TP-2003-300*.
- Bassi, F. and Rebay, S. (1997). High-order accurate discontinuous finite element solution of the 2D Euler equations. *Journal of Computational Physics* 138, 251-285.

Wang, Z. J. and Liu, Y. (2006). Extension of the

- spectral volume method to high-order boundary representation, *Journal of Computational Physics* 211, 154-178.
- Kannan, and Wang, Z. J. (2011). Curvature and entropy based wall boundary condition for the high order spectral volume Euler solver. *Computers & Fluids* 44, 79-88.
- Fidkowski, K. J. and Darmofal, D. L. (2007) A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations, *Journal of Computational Physics* 225, 1653–1672.
- Karimian, S. M. H. and Ardakani, M. (2011) Immersed boundary method for the Solution of 2D inviscid compressible flow using finite volume approach on moving Cartesian grid, *Journal of Applied Fluid Mechanics* 4(2), 27-36.
- Vymazal, M. and Quintino, T. and Villedieu, N. and Deconinck, H. (2011) High-order upwind residual distribution schemes on isoparametric curved elements, *Journal of Computational Physics* 230, 890–906.
- Sevilla, R. and Fernandez-Mendez, S. and Huerta, A. (2008). NURBS-enhanced finite element method for Euler equations. *International Journal of Numerical Methods in Fluids* 57, 1051-1069.
- Gao, H. and Wang, Z. J. and Liu, Y. (2010). A study of curved boundary representation for 2D high-order Euler solvers. *Journal of Scientific Computing* 44, 323-336.
- Wang, Z. J. and Gao, H. (2009) A unifying collocation penalty formulation for the Euler and Navier-Stokes equations on mixed grids. *AIAA-2009-0401*.
- Roe, P. L. (1981). Approximate Riemann solvers, parameter vectors, and difference scheme. *Journal of Computational Physics* 43, 357 – 372.
- Solin, P. and Segeth, K. and Dolezel, I. (2003). High order finite element methods, Chapman and Hall.
- Cockburn, B. and Shu, C-W. (1998). The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems, *Journal of Computational Physics* 141, 199-224.
- Li, B.Q. (2006). Discontinuous finite elements in fluid dynamics and heat transfer, Springer.
- Esfahanian, V. and Nejat, A. and Baghapour, B. (2011). High-order cubic spline boundary representation for discontinuous Galerkin simulation of compressible inviscid flows. *AIAA* 2011-3838.
- Krivodonova, L. and Berger, M. (2006). High-order accurate implementation of solid wall boundary condition in curved geometries, *Journal of Computational Physics* 211, 492-512.
- Liang, C. and Jameson, A. and Wang, Z.J. (2009). Spectral difference method for compressible flow on unstructured grids with mixed elements. *Journal of Computational Physics* 228, 2847-2858.
- Nejat, A. (2007). A High-order Accurate Unstructured Finite-Volume Newton-Krylov Algorithm for Inviscid Compressible Flows. Ph.D. thesis. University of British Columbia, Canada.
- Aftosmis, and Gaitonde, M. D. and Tavares, T.S. (1994) On the accuracy, stability and monotonicity of various reconstruction algorithms for unstructured meshes. *AIAA-94-0415*.
- Mason, W. H. (2006) Configuration Aerodynamics, Chapter 6: Subsonic aerodynamics of airfoils and wings, AOE 4124. Virginia Polytechnic Institute and State University.