6th European Conference on Computational Mechanics (ECCM 6)
7th European Conference on Computational Fluid Dynamics (ECFD 7)
1115 June 2018, Glasgow, UK

ON NUMERICAL INTEGRATIONS OF DISCONTINUOUS GALERKIN METHODS OVER HIGH-ORDER CURVED ELEMENTS

Hojun You[1] **and Chongam Kim**[1]

[1] Department of Mechanical and Aerospace Engineering, Seoul National University
Seoul 08826, Republic of Korea,
e-mail: chongam@snu.ac.kr

Key words: Discontinuous Galerkin Methods, Numerical Integration, Quadrature Rules, High-order Curved Elements

Abstract. In this study, novel numerical approaches to handle the volume and surface integration terms of discontinuous Galerkin (DG) methods are proposed to significantly reduce computational costs especially on higher-order curved elements. One approach, called Direct Quadrature Method (DQM), generates a quadrature rule directly on the physical domain for a given set of points that are optimized to have well-conditioned Vandermonde matrix. The other one, called Direct Reconstruction Method (DRM), re-expresses the flux terms as Lagrange polynomials directly on the physical domain and integrates them in the pre-processing step. Since the proposed methods directly handle the integrations on the physical domain, they solely rely on the order of the solution approximation, not on the order of the elements. Thus, they are particularly efficient on higher-order curved elements. In order to evaluate the proposed methods, two benchmark problems of the compressible Euler equations are computed on high-order curved quadrilateral meshes in two dimensions. Results verify that the proposed methods reduce the substantial amount of computational cost without compromising the required order-of-accuracy.

1 INTRODUCTION

The high-order methods are emerging as the latest trend in computational fluid dynamics (CFD) community to simulate more complex flow problems with better resolution and less numerical damping. Among the high-order methods based on finite element methods (FEM), the DG method has been the most popular because of its mathematical and numerical rigor, grid flexibility of shape functions and intuitive formulation [1]. The high-order methods, however, still face a few obstacles such as shock-driven instabilities and severe computational costs, which severely limit their applicability. Particularly, heavy computational cost mainly prevents both academia and industry from applying the high-order methods to practical real-life problems.

In the DG method, efficient handling of numerical integrations in its weak formulation is a critical component to determine numerical accuracy and efficiency. Conventional approaches to integrate these terms are based on the prescribed quadrature rules that are iteratively calculated and optimized on the reference domain. Its computing costs, however, exponentially increase according to the element order because coordinate transformation to the reference domain contains very high-degree polynomial terms that require even higher-order quadrature rules. In the case of three dimensions, for example, the number of quadrature points for the volume integration is proportional to the cubic orders of both solution approximation and element, and on top of it, state variables and flux functions should be evaluated at each point. This greatly increases the computational cost of the DG method. In order to resolve these difficulties, novel numerical approaches to integrate the surface and volume integral terms appeared in the DG method are considered.

From analysis and validation, the proposed methods are turned out to be much more efficient than the conventional one. Direct integrations on the physical domain without coordinate transformation make them independent of the element order, which significantly increase computational efficiency in higher-order elements. The present work is organized as follows. In Section 2, we briefly review the DG formulation and conventional way of handling the integration terms. In Section 3, two novel approaches, called the DQM and DRM approaches, are provided to efficiently compute the numerical integrations in the DG formulation. These new approaches are applied to some representative benchmark problems and compared with the conventional one in Section 4. Finally, concluding remarks and future works are given in Section 5.

2 Discontinuous Galerkin Formulation

We consider the hyperbolic conservation laws

$$\frac{\partial q}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F} = 0 \tag{1}$$

subjected to the well-posed initial condition $q(\mathbf{x}, 0) = q_0(\mathbf{x})$ and boundary conditions on the appropriate physical domain $\Omega = \cup_{i=1}^{N} \Omega_i$, where Ω_i is a non-overlapping polygonal element. The flux function is given by $\mathbf{F} = \left[F^1, \cdots, F^d\right]^T$, where d is dimension. We assume that Eq. (1) is scalar for the purpose of simplicity, but it can be easily extended into vector equations.

In the DG method, the approximated solution is contained in a global function space $V_h = \oplus_{i=1}^{N} V_h^i$, where $V_h^i \equiv \text{span}\{\phi_i^{(j)}\}_{j=1,\cdots,N_k}$ and $\phi_i^{(j)}$ are basis functions that are usually chosen to be the orthonormal polynomials in $\mathbb{P}^k(\Omega_i)$. Here, $\mathbb{P}^k(\Omega_i)$ is a set of k^{th}-degree polynomials defined on Ω_i. For a given element configuration, $\phi_i^{(j)}$ can be simply computed by using the modified Gram-Schmidt (MGS) process [2]. Multiplying the basis function to Eq. (1) and integrating it over Ω_i, we can get the following weak formulation.

$$\int_{\Omega_i} \frac{\partial q_h}{\partial t} \phi_i^{(j)} dV + \int_{\partial \Omega_i} \phi_i^{(j)} \hat{\mathbf{F}}(q_h^-, q_h^+) \cdot \mathbf{n} dA = \int_{\Omega_i} \mathbf{F} \cdot \boldsymbol{\nabla} \phi_i^{(j)} dV. \tag{2}$$

Here, $\hat{\mathbf{F}}$ is a monotone numerical flux from the finite volume method. Applying the orthonormality of the basis polynomials, the first term on the LHS of Eq. (2) is re-expressed as

$$\int_{\Omega_i} \frac{\partial q_h}{\partial t} \phi_i^{(j)} dV = \frac{d}{dt} \begin{pmatrix} q_i^{(1)} \\ \vdots \\ q_i^{(N_k)} \end{pmatrix}, \tag{3}$$

where $q_i^{(j)}$ are the modal coefficients satisfying

$$q_h(\mathbf{x}, t) = \sum_{j=1}^{N_k} q_i^{(j)}(t) \phi_i^{(j)}(\mathbf{x}) \tag{4}$$

on each cell Ω_i. The second term on the LHS and the RHS term of Eq. (2) are the surface and volume integrations, respectively. In order to achieve the formal order-of-accuracy, those integrations should be computed with the $(2k+1)^{\text{th}}$- and $(2k)^{\text{th}}$-accuracy, respectively.

A conventional way to integrate the terms in Eq. (2) is to apply pre-determined numerical quadrature or cubature rules on the reference domain. There are lots of quadrature or cubature rules in literatures that were iteratively computed and optimized for various reference element configurations. These reference elements usually contain triangle and quadrilateral in two-dimensional case, and tetrahedron/hexahedron/prism/pyramid in three-dimensional case. These are common element types in mixed meshes. Let $\mathfrak{T}_i$ be an invertible transformation mapping from the reference element $\tilde{\Omega}$ to a physical element Ω_i in the physical domain.

$$\mathfrak{T}_i : \mathbf{r} = (\xi, \eta, \chi) \in \tilde{\Omega} \mapsto \mathbf{x} = (x, y, z) \in \Omega_i. \tag{5}$$

Here, the tilde symbol indicates the reference domain. Applying coordinate transformation, the volume and surface integration terms of Eq. (2) are re-expressed as follows.

$$\int_{\tilde{\Omega}} |\mathbf{J}| (\mathbf{F} \cdot \boldsymbol{\nabla} \phi) \circ \mathfrak{T} dV, \tag{6}$$

$$\int_{\partial\tilde{\Omega}} |\mathbf{J}| \left\{ (\phi \hat{\mathbf{F}}) \circ \mathfrak{T} \right\} \cdot (\mathbf{J}^{-1})^T \tilde{\mathbf{n}} dA, \tag{7}$$

where $\mathbf{J}$ is the Jacobian matrix of the transformation. For convenience, we delete the cell index i and the basis index j without any confusion. Both integrations are then computed using a consistent quadrature rule with the required accuracy.

For simplex elements, the transformation for high-order curved Pn-elements consists of n^{th}-degree polynomials. Thus, the formal degrees of the integrands of Eqs. (6-7), which are denoted as $d_{v,d}$ and $d_{f,d}$, respectively, are given as follows.

$$d_{v,d} = 2kn + d(n-1), \tag{8}$$

$$d_{f,d} = (2k+1)n + (d-1)(n-1), \tag{9}$$

where d is dimension. The minimum possible number of optimal quadrature points for Eqs. (6-7) is then estimated as

$$N_{v,d} = \left\lceil \frac{{}_{(d_{v,d}+d)}C_d}{d+1} \right\rceil \sim O((kn)^d), \tag{10}$$

$$N_{f,d} = \left\lceil \frac{{}_{(d_{f,d}+d)}C_d}{d_{f,d}+d} \right\rceil \sim O((kn)^{d-1}), \tag{11}$$

where $\lceil \cdot \rceil$ is the ceiling symbol. As shown in Tabs. (1-2), they rapidly increase according to the orders of approximation and element.

A huge number of quadrature points pose a few hurdles. Firstly, very high-order quadrature rules are rarely presented in literatures. Secondly, even higher quadrature rules are necessary for over-integration techniques, say, to prevent aliasing-driven instabilities in turbulent simulation. Such higher quadrature rules are not only unpractical in actual computations, but also leading to severe computational burden since they require a large number of flux evaluations and state variable computations for a huge number of quadrature points. Thus, it is essential to develop efficient and practical methods to handle the integration terms of the DG method. From this perspective, we propose newly developed integration methods for both the volume and surface integrations; (1) Direct Quadrature Method (DQM) and (2) Direct Reconstruction Method (DRM).

Table 1: Minimum possible numbers($N_{v,d}$) of optimal quadrature points for DG volume integration of Eq. (6) on two- and three-dimensions(d) with respect to the orders of DG approximation(k) and element(n). Element is assumed to be simplex

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	2	5	10	15	22	31	3	9	21	42	72	114
	2	10	22	40	64	92	126	30	91	204	385	650	1015
	3	22	51	92	145	210	287	114	333	732	1364	2285	3548
	4	40	92	166	260	376	514	285	819	1785	3311	5525	8555
	5	64	145	260	409	590	805	575	1637	3548	6559	10920	16882
	6	92	210	376	590	852	1162	1015	2870	6201	11440	19019	29370

3 Proposed Numerical Integration Methods

3.1 Direct Quadrature Methods (DQM)

3.1.1 DQM for Volume Integration

Here, the volume integration is directly computed by the newly provided quadrature rules on the physical domain. Unlike the conventional approach that has been executed

Table 2: Minimum possible numbers($N_{f,d}$) of optimal quadrature points for DG surface integration Eq. (7) on two- and three-dimensions(d) with respect to the orders of DG approximation(k) and element(n). Element is assumed to be simplex

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	2	3	4	5	6	7	4	7	12	19	26	35
	2	4	6	8	10	12	14	15	31	51	77	109	145
	3	6	9	12	15	18	21	35	70	117	176	247	330
	4	8	12	16	20	24	28	64	126	210	316	442	590
	5	10	15	20	25	30	35	100	199	330	495	694	925
	6	12	18	24	30	36	42	145	287	477	715	1001	1335

in the reference domain, this does not require any transformation during runtime computation. In order to create a quadrature rule to compute the volume integration term with an optimal order-of-accuracy, we need to choose a proper target space that covers $\mathbf{F} \cdot \boldsymbol{\nabla}\phi$ on the physical domain. Since the solution is approximated by the k^{th}-degree polynomials, the target space is trivially given by

$$\mathbf{F} \cdot \boldsymbol{\nabla}\phi \in \mathbb{P}^{2k}(\Omega). \tag{12}$$

Then we can find an orthonormal basis set $\{\varphi^{(j)}\}_{j=1,\cdots,N_{2k}}$ that spans the target space by using the MGS process with the monomials of $\mathbb{P}^{2k}(\Omega)$ as initial bases. Every quadrature rule should satisfy the following equations.

$$\sum_{m=1}^{N_{2k}} w_m \varphi^{(j)}(\mathbf{x}_m) = \delta_{1,j} \int_{\Omega} \varphi^{(j)} dV, \tag{13}$$

where $\delta_{a,b}$ is Kronecker delta. Here, the lowest mode, $\varphi^{(1)}$, is a constant that satisfies orthonormality, *i.e.*

$$\int_{\Omega} \varphi^{(i)} \varphi^{(j)} dV = \delta_{i,j}. \tag{14}$$

Thus, $\int_{\Omega} \varphi^{(1)} dV = \sqrt{|\Omega|}$. Equation (13) is generally a system of non-linear equations that has w_m and $\mathbf{x}_m$ as unknowns, and it requires an iterative method such as the Newton-Rapson method that requires heavy computing cost. If we pre-determine the coordinates of the quadrature points, however, Eq. (13) would be nothing but a system of linear equations that only has w_m as unknowns, and thus it just requires one matrix inversion that is much more efficient than the Newton-Rapson iteration. The resulting quadrature rule has a very small number of quadrature points compared with the conventional methods as shown in Tabs. 1 and 3. At the same time, the number of quadrature points is independent on the element order.

It still remains how to choose the quadrature points for a given element configuration. One direction is to minimize the condition number of the matrix in Eq. (13). For a reference triangle, we already know, as a proper candidate, a set of the alpha-optimized points [1]. For an arbitrary high-order curved triangle, however, it is not practical to find such points one by one. Thus, we use the transformed alpha-optimized points by the element transformation $\mathfrak{T}$. For a reference quadrilateral, we use a kind of brute force approach to find a set of points that has a relatively small condition number. For an arbitrary high-order curved quadrilateral, we use the transformed points. The coordinates of such points are provided in Appendix A.

Table 3: The numbers of quadrature points of DQM for the DG volume integration on two- and three-dimensions(d) with respect to the orders of DG approximation(k) and element(n). Effective regions are shaded.

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	6	15	28	45	66	91	10	35	84	165	286	455
	2	6	15	28	45	66	91	10	35	84	165	286	455
	3	6	15	28	45	66	91	10	35	84	165	286	455
	4	6	15	28	45	66	91	10	35	84	165	286	455
	5	6	15	28	45	66	91	10	35	84	165	286	455
	6	6	15	28	45	66	91	10	35	84	165	286	455

3.1.2 DQM for Surface Integration

In the similar way described in Section 3.1.1, the surface integration can be directly computed by the newly provided quadrature rules on the physical domain. In order to do this, we firstly find a target space for the integrand $\phi\hat{\mathbf{F}}\cdot\mathbf{n}$. Note that this term is not covered by polynomial space because the normal vector is not a polynomial on arbitrary high-order curved surface. In this case, the proper target space is given by

$$\phi\hat{\mathbf{F}}\cdot\mathbf{n} \in \left[\mathbb{P}^{2k+1}(\Omega)\right]^d \cdot \mathbf{n} \equiv \mathbb{P}^{2k+1}(\Omega)n_1 + \cdots + \mathbb{P}^{2k+1}(\Omega)n_d. \tag{15}$$

Then we can find a basis set $\{\varphi^{(j_1)}n_1 + \cdots + \varphi^{(j_1)}n_d\}_{(j_1,\cdots,j_d)=(1,\cdots,N_{2k+1})^d}$, where φ are derived by the MGS process. As shown in Section 3.1.1, the number of quadrature points for DQM is the same as the number of basis of the target space. The required number of point for applying DQM to the surface integration is shown in Tab. 4. Effective region of DQM, however, is rather small for practical purpose. Thus, we will not consider DQM for the surface integration in this work. Instead of DQM, DRM is provided in the next section.

Table 4: The numbers of quadrature points of DQM for the DG surface integration on two- and three-dimensions(d) with respect to the orders of DG approximation(k) and element(n). Effective regions are shaded.

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	20	42	72	110	156	210	60	168	360	660	1092	1680
	2	20	42	72	110	156	210	60	168	360	660	1092	1680
	3	20	42	72	110	156	210	60	168	360	660	1092	1680
	4	20	42	72	110	156	210	60	168	360	660	1092	1680
	5	20	42	72	110	156	210	60	168	360	660	1092	1680
	6	20	42	72	110	156	210	60	168	360	660	1092	1680

3.2 Direct Reconstruction Methods (DRM)

3.2.1 DRM for Volume Integration

The volume integration is computed by directly reconstructing the target integrand using the nodal basis polynomials in the physical domain. In expressing the total integrand, $\mathbf{F} \cdot \boldsymbol{\nabla}\phi$, as a linear combination of the nodal basis, we only represent the flux function by the Lagrange polynomials since the exact value of ϕ is already known.

$$\mathbf{F} = \sum_{m=1}^{N_{k+1}} \mathbf{F}(\mathbf{x}_m)\ell^{(m)}, \tag{16}$$

where ℓ are the Lagrange polynomials with $\ell^{(a)}(\mathbf{x}_b) = \delta_{a,b}$. We assume the flux function is in $[\mathbb{P}^{k+1}(\Omega)]^d$. Let $\{\varphi^{(j)}\}_{j=1,\cdots,N_{k+1}}$ be a set of basis spanning $\mathbb{P}^{k+1}(\Omega)$. Moreover, the following is valid.

$$\phi^{(j)} = \varphi^{(j)}, \quad \text{for} \;\; j = 1, \cdots, N_k. \tag{17}$$

The Lagrange polynomials can be easily derived by the following relation.

$$\mathbf{L} \equiv \begin{pmatrix} \ell^{(1)} \\ \vdots \\ \ell^{(N_{k+1})} \end{pmatrix} = \mathbf{V}^{-1} \begin{pmatrix} \varphi^{(1)} \\ \vdots \\ \varphi^{(N_{k+1})} \end{pmatrix} \equiv \mathbf{V}^{-1}\boldsymbol{\varphi}, \tag{18}$$

where $\mathbf{V}$ is the Vandermonde matrix with $\mathbf{V}_{(i,j)} = \varphi^{(i)}(\mathbf{x}_j)$. Thus, we have

$$\begin{aligned} \int_\Omega \mathbf{F} \cdot \boldsymbol{\nabla}\phi dV &= \int_\Omega \sum_{i=1}^{d} (\mathbf{F}^i \cdot \mathbf{L}) \frac{d\boldsymbol{\phi}}{dx^i} dV = \int_\Omega \sum_{i=1}^{d} (\mathbf{F}^i \cdot \mathbf{V}^{-1}\boldsymbol{\varphi}) \frac{d\boldsymbol{\phi}}{dx^i} dV \\ &= \sum_{i=1}^{d} \left[\left(\int_\Omega \frac{d\boldsymbol{\phi}}{dx^i} \boldsymbol{\varphi}^T dV \right) (\mathbf{V}^{-1})^T \right] \mathbf{F}^i \equiv \sum_{i=1}^{d} \mathbf{S}_v^i \mathbf{F}^i, \end{aligned} \tag{19}$$

where $\mathbf{S}_v^i \equiv [(\int_\Omega \frac{d\boldsymbol{\phi}}{dx^i}\boldsymbol{\varphi}^T dV)(\mathbf{V}^{-1})^T]$, $\mathbf{F}^i \equiv [F^i(\mathbf{x}_1), \cdots, F^i(\mathbf{x}_{N_{k+1}})]^T$, and $\boldsymbol{\phi} \equiv [\phi^{(1)}, \cdots, \phi^{(N_k)}]^T$. Here, we call $\mathbf{S}_v^i$ the DRM stiff matrix with respect to the x^i coordinate for the volume integration. The volume integration is computed after d-times matrix-vector multiplications between the DRM stiff matrices and the vectors of the flux nodal coefficients. In practice, all DRM stiff matrices are pre-computed in the pre-processing step, and DQM points are used as nodal points. The number of point required for the DRM volume integration is shown in Tab. 5. DRM is proved to be even more efficient than DQM.

Table 5: The numbers of points of DRM for the DG volume integration on two- and three- dimensions(d) with respect to the orders of DG approximation(k) and element(n). Effective regions are shaded.

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	6	10	15	21	28	36	10	20	35	56	84	120
	2	6	10	15	21	28	36	10	20	35	56	84	120
	3	6	10	15	21	28	36	10	20	35	56	84	120
	4	6	10	15	21	28	36	10	20	35	56	84	120
	5	6	10	15	21	28	36	10	20	35	56	84	120
	6	6	10	15	21	28	36	10	20	35	56	84	120

3.2.2 DRM for Surface Integration

The same DRM can be extended into the surface integration. In order to reconstruct the flux function, we firstly need to find the nodal polynomials in a proper target space. Because we want to directly handle the integration in the physical domain, our target space is trivially $\mathbb{P}^{k+1}(\Omega)|_{\partial\Omega}$. Then the modal basis polynomials can be computed by the MGS process in the target space. In order to realize that, we define the following inner product.

$$(f, g)_{\partial\Omega} \equiv \int_{\partial\Omega} \frac{fgdA}{\||\mathbf{J}|(\mathbf{J}^{-1})^T\tilde{\mathbf{n}}\|}, \tag{20}$$

It is then easy to show that Eq. (20) satisfies the conjugate symmetry, linearity, and positive-definiteness of an associated inner product space. As an initial set of the MGS process, the hierarchical monomials of $\mathbb{P}^{k+1}(\Omega)$ are used. The linearly dependent monomials can be easily removed by slightly modifying the conventional MGS algorithm. In addition, we also eliminate the monomials that are too stiff leading to an ill-conditioned Vandermonde matrix. For two-dimensional case, the cut-off condition number is set to 10 and it gives somewhat adaptive results depending on the curvature of the surface. Let $\boldsymbol{\varphi} \equiv [\varphi^{(1)}, \cdots, \varphi^{(M_{k+1})}]^T$ is a set of the resulting modal bases. If we choose a proper set of points, we can derive the Lagrange polynomials such that

$$\mathbf{L} \equiv \begin{pmatrix} \ell^{(1)} \\ \vdots \\ \ell^{(M_{k+1})} \end{pmatrix} = \mathbf{V}^{-1} \begin{pmatrix} \varphi^{(1)} \\ \vdots \\ \varphi^{(M_{k+1})} \end{pmatrix} \equiv \mathbf{V}^{-1}\boldsymbol{\varphi}, \tag{21}$$

where $\mathbf{V}$ is the Vandermonde matrix. Here, the Gauss-Legendre points were used as the nodal points in two dimensions by transformation. Thus, the following is achieved.

$$\begin{aligned} \int_{\partial\Omega} \phi \hat{\mathbf{F}} \cdot \mathbf{n} dA = \int_{\partial\Omega} \boldsymbol{\phi} \sum_{i=1}^{d} (\hat{\mathbf{F}}^i \cdot \mathbf{L}) n_i dA = \int_{\partial\Omega} \boldsymbol{\phi} \sum_{i=1}^{d} (\hat{\mathbf{F}}^i \cdot \mathbf{V}^{-1}\boldsymbol{\varphi}) n_i dA \\ = \sum_{i=1}^{d} \left[\left(\int_{\partial\Omega} \boldsymbol{\phi}\boldsymbol{\varphi}^T n_i dA \right) (\mathbf{V}^{-1})^T \right] \hat{\mathbf{F}}^i \equiv \sum_{i=1}^{d} \mathbf{S}_f^i \hat{\mathbf{F}}^i, \end{aligned} \tag{22}$$

where $\mathbf{S}_f^i \equiv [(\int_{\partial\Omega} \boldsymbol{\phi}\boldsymbol{\varphi}^T n_i dA)(\mathbf{V}^{-1})^T]$ is the DRM mass matrix with respect to the n_i component for the surface integration, and $\hat{\mathbf{F}}^i \equiv [\hat{F}^i(q^-(\mathbf{x}_1), q^+(\mathbf{x}_1)), \cdots, \hat{F}^i(q^-(\mathbf{x}_{M_{k+1}}), q^+(\mathbf{x}_{M_{k+1}}))]^T$ is the $i-$directional vector of the numerical flux at nodal points. Note that the numerical flux is defined for each direction, not combined with the normal vector. For the numerical flux using the contravariant velocity

$$\widehat{\mathbf{F} \cdot \mathbf{n}} = \frac{\mathbf{F}(q^-) + \mathbf{F}(q^+)}{2} \cdot \mathbf{n} - A(q^-, q^+, \mathbf{n}), \tag{23}$$

$\hat{F}^i$ is computed by

$$\hat{F}^i \equiv \frac{F^i(q^-) + F^i(q^+)}{2} - A(q^-, q^+, \mathbf{n}) n^i, \tag{24}$$

where $A(q^-, q^+, \mathbf{n})$ is a numerical diffusion. Thus, $\widehat{\mathbf{F} \cdot \mathbf{n}} = \hat{\mathbf{F}} \cdot \mathbf{n}$. In the DRM approach for the surface integration, it is difficult to estimate the required number of points with respect to the order of solution approximation and element, because we do not know the dimension of $\mathbb{P}^{k+1}(\Omega)|_{\partial\Omega}$ explicitly. One thing clear is that the upperbound is the dimension of $\mathbb{P}^{k+1}(\Omega)$ and this value is tabulated in Tab. 6.

4 Numerical Results

In order to examine the performances of the DQM and DRM methods, the compressible Euler equations are computed on high-order meshes. Euler advection problem is considered as an unsteady case, and circular cylinder problem is chosen as a steady case. Both problems are computed by DG-*P3* and -*P6* methods on curved *P3*- and *P6*-elements, respectively. For the time marching, the 4^{th}-order 5^{th}-stage strong stability preserving Runge-Kutta (SSPRK) method is used with the CFL number of 0.9. As a numerical flux, Roe's FDS is used. For each benchmark problem, conventional quadrature method is used as baseline test, and four tests were carried out by applying the proposed methods: (1) DQM and (2) DRM for the volume integration, (3) DQM and (4) DRM for the volume integration along with DRM for the surface integration. All computations are performed by Intel Xeon X5650 in serial.

Table 6: Upper bound of the numbers of points of the DRM for the DG surface integration on two- and three-dimensions(d) with respect to the orders of DG approximation(k) and element(n). Effective regions are shaded. Note that the effective regions become wider in actual computations.

d		2						3					
k		1	2	3	4	5	6	1	2	3	4	5	6
n	1	6	10	15	21	28	36	10	20	35	56	84	120
	2	6	10	15	21	28	36	10	20	35	56	84	120
	3	6	10	15	21	28	36	10	20	35	56	84	120
	4	6	10	15	21	28	36	10	20	35	56	84	120
	5	6	10	15	21	28	36	10	20	35	56	84	120
	6	6	10	15	21	28	36	10	20	35	56	84	120

4.1 Advection Problem

Advection equation given by the following initial condition

$$(\rho, u, v, P) = (1 + 0.2\sin(2\pi x)\sin(2\pi y), 1, 0.5, 1) \tag{25}$$

is computed on the computational domain of $[0, 1] \times [0, 1]$ until $t = 2.0$. The periodic boundary condition is applied. Cartesian meshes are transformed by

$$(x, y) \mapsto (x, y) + 0.1\cos(\pi(x - 0.5))\sin(2\pi(y - 0.5))(1, -1) \tag{26}$$

to manufacture high-order curved meshes. The coarsest mesh is shown in Fig. 1. Fig. 2 shows the computed results. Here, the length scale is defined by

$$\text{Length Scale} \equiv 1/\sqrt[d]{\text{nDOFs}}. \tag{27}$$

As shown in Fig. 2, the formal order-of-accuracy is well preserved and there is no difference among the errors calculated by each method. It is, however, clearly observed that the proposed methods significantly reduce the computational costs compared with the conventional method. This is because the number of points for quadrature and/or flux evaluations is substantially reduced, which is tabulated in Tab. 7. In both DG-*P3* and -*P6* cases, DRM shows better performance than DQM for the volume integration because it requires much less number of points. In DG-*P3* case, however, DRM for the surface integration is more expensive than the conventional surface integration, though the required number of point is still less. This is because additional computational efforts are required to split the numerical flux into each direction and to gather it during the multiplication step with the DRM mass matrices for the surface integration. In the case of DG-*P6*, DRM for the surface integration outperforms again. This is because the benefits of the substantially reduced number of points far outweigh the additional computational costs. Thus, DRM for the surface integration is efficient when higher-order elements are employed. It is observed that the coarsest results of DQM for the volume integration in DG-*P6* blow up because of numerical instability.

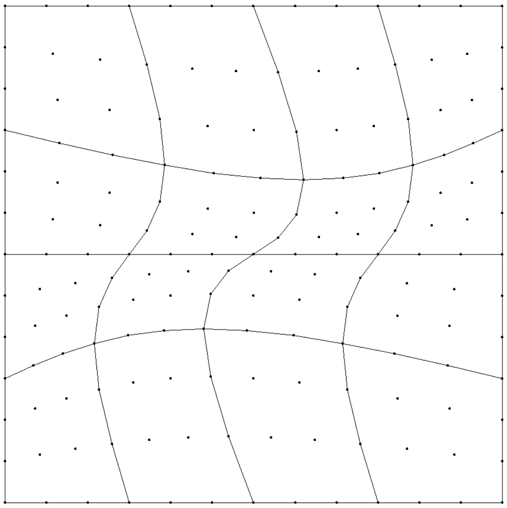
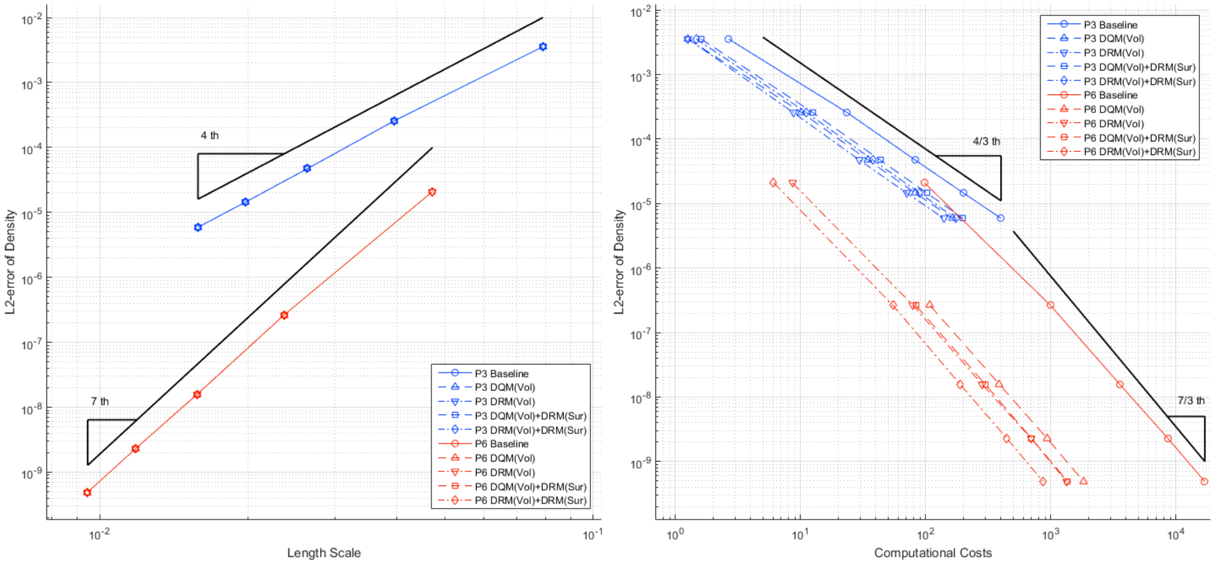


Figure 1: Manufactured high-order curved mesh in the advection problem.



(a)

(b)

Figure 2: Results of the advection problem: L_2-error of density with respect to (a) length scale and (b) computational cost.

Table 7: The number of points for quadrature and/or flux evaluations required in the advection problem.

Mesh	**DG-*P3***					**DG-*P6***				
	Volume			**Surface**		**Volume**			**Surface**	
	Base	**DQM**	**DRM**	**Base**	**DRM**	**Base**	**DQM**	**DRM**	**Base**	**DRM**
4×4	2304	448	240	480	338	28224	1456	576	1680	476
8×8	9216	1792	960	1728	1339	112896	5824	2304	6048	1632
12×12	20736	4032	2160	3744	2881	254016	13104	5184	13104	3380
16×16	36864	7168	3840	6528	4956	451584	23296	9216	22848	5593
20×20	57600	11200	6000	10080	7502	705600	36400	14400	35280	8396

4.2 Circular Cylinder Problem

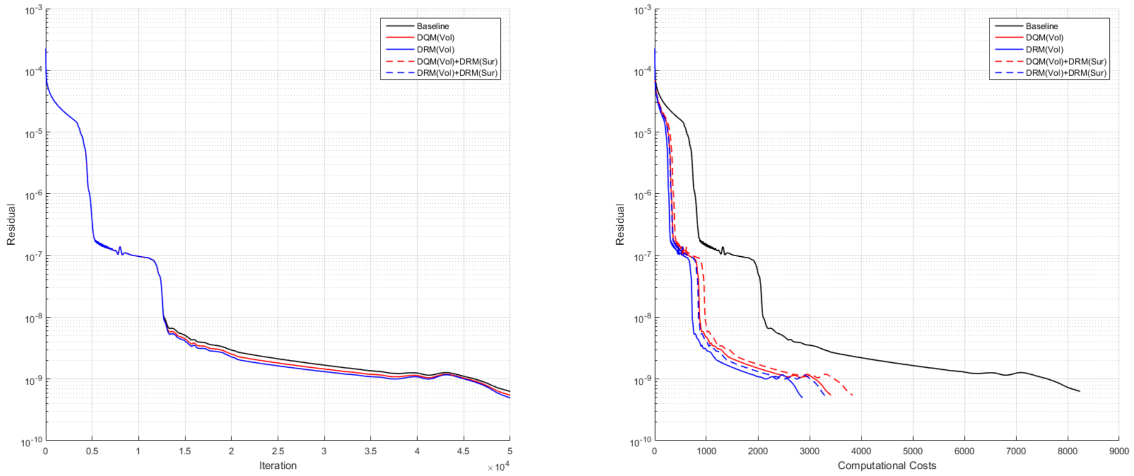
Owing to flow symmetry, an upper half cylinder with a radius of 1 located at the origin is considered. The mesh contains 480 curved quadrilateral elements with 16 elements along the surface of the half cylinder. The computational domain is extended to the radial direction by stretching the grid size exponentially with a factor of 0.1, and the first grid line off the wall is set to 0.2. The free stream Mach number is $M_\infty = 0.1$, and the flow field is initially set to the free stream values everywhere in the domain. For DG-*P3* case on *P3*-elements, a converged solution is obtained by up to 50,000 iterations. For DG-*P6* case on *P6*-elements, the initial flow field is firstly stabilized by DG-*P3* approximation until 5,000 iterations. After that, the intermediate result is lifted onto $\mathbb{P}^6$ space, and a final converged solution is obtained by up to 50,000 iterations with DG-*P6* approximation. At every iteration, the residual defined by

$$\text{Residual} \equiv \left(\frac{\sum_i \int_{\Omega_i} \left| u(t_{n+1}) - u(t_n) \right|^2 dV}{\sum_i |\Omega_i|} \right)^{1/2} \tag{28}$$

is computed by the conventional quadrature method.

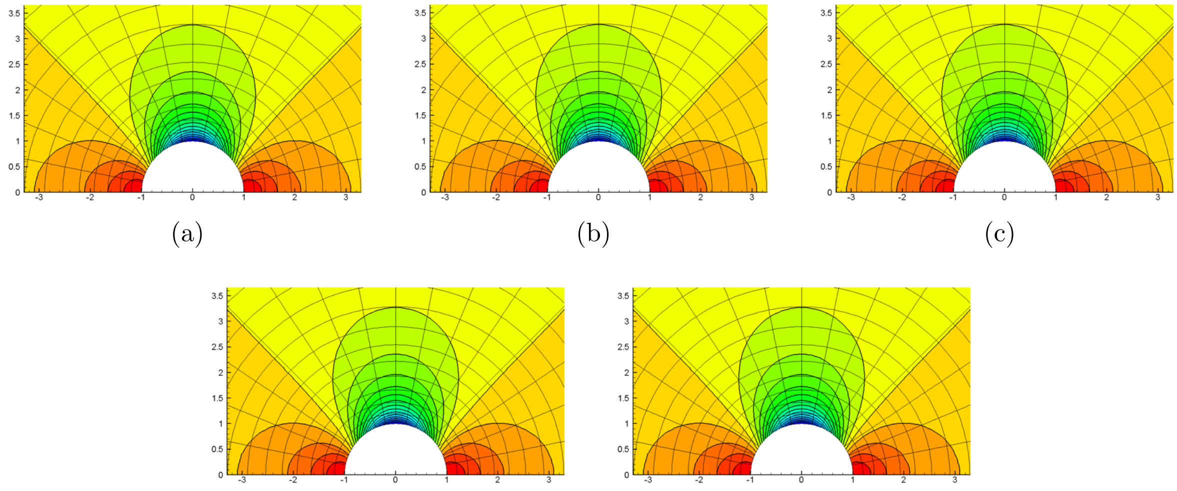
Fig. 3 shows the results of DG-*P3* case. All residual curves are almost overlapped into a single curve, but after about 13,000 iterations, DRM for the volume integration yields faster convergence regardless of the surface integration method. DQM for the volume integration also provides better performance than the conventional method. The residual curves are not affected by the DRM surface integration. All the proposed methods significantly reduce the computational costs as shown in Fig. 3. DQM and DRM for the volume integration bring 2.41× and 2.88× speedup, respectively. DRM for the surface integration provides 2.15× and 2.46× speedup for each volume integration method, indicating that it rather increases the computational cost. This is already observed in the advection problem, and the reason is the same as before. Converged solutions are plotted in Fig. 4.

Fig. 5 shows the DG-*P6* results. The residual curve of DRM for the volume integration is overlapped again with the baseline one, but after about 19,000 iterations, the



(a) (b)

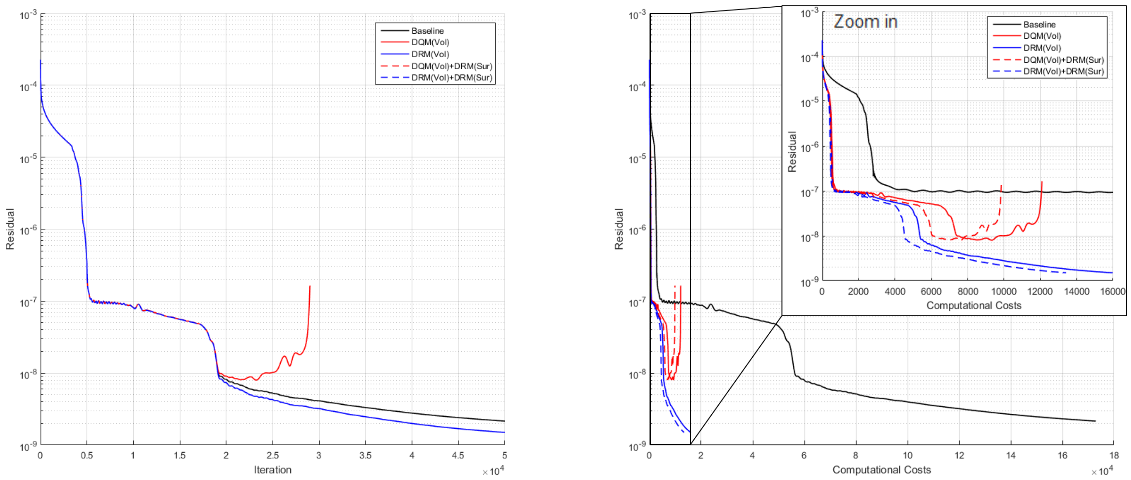
Figure 3: Results of the circular cylinder problem by DG-*P3* method: residual of density with respect to (a) iteration and (b) computational cost.



(a) (b) (c)

(d) (e)

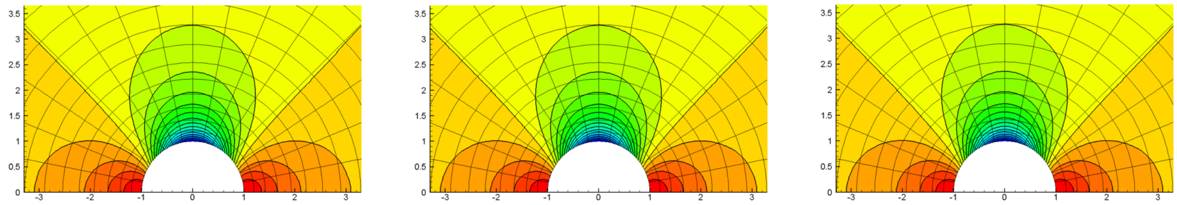
Figure 4: Density contours of converged results of the circular cylinder problem by DG-*P3* method: (a) Baseline, (b) DQM and (c) DRM for the volume integration, (d) DQM and (e) DRM for the volume integration with DRM for the surface integration.



(a)

(b)

Figure 5: Results of the circular cylinder problem by DG-*P6* method: residual of density with respect to (a) iteration and (b) computational cost.



(a)

(b)

(c)

Figure 6: Density contours of converged results of the circular cylinder problem by DG-*P6* method: (a) Baseline, (b) DRM for the volume integration and (c) DRM for the volume and surface integration.

DRM residual level is a bit lower. In the case of the DQM volume integration, the residual diverges after about 19,000 iterations. Numerical instability arises at the elements containing stagnation point. DRM for the volume integration remarkably reduces the computational cost. It brings 10.9× speedup, and 12.8× speedup with DRM for the surface integration. Contrary to DG-*P3* case, DRM for the surface integration successfully reduces the computational cost. Converged contours are plotted in Fig. 6, and the number of points for quadrature and/or flux evaluations is shown in Tab. 8.

5 Conclusion

Two novel approaches, called DQM and DRM, are proposed to alleviate the computational burden of the DG methods on high-order curved elements. From numerical design to computational experiments for the 2-D compressible Euler equations, both methods applied to the volume integration and substantially reduce the amount of required computation. As a result, computational efficiency is dramatically improved on high-order

Table 8: The number of points for quadrature and/or flux evaluations required for each method in the circular cylinder problem.

DG-*P3*					**DG-*P6***				
Volume			**Surface**		**Volume**			**Surface**	
Base	**DQM**	**DRM**	**Base**	**DRM**	**Base**	**DQM**	**DRM**	**Base**	**DRM**
69120	13440	7200	12072	8356	846720	43680	17280	42252	14171

elements such as DG-*P6* case. DQM, however, yields somewhat unstable behavior on DG-*P6* with SSPRK. In particular, DRM for higher-order elements turns out to be accurate and efficient in both the volume and surface integration. Both methods will be continuously extended into 3-D mixed curved elements, and extensive benchmark tests containing discontinuities will be carried out. As shown in Tabs. 1 and 2, computational efficiency is expected to be improved much better in 3-D cases.

6 Acknowledgement

Authors appreciate the financial supports provided by the program of Development of Space Core Technology (NRF-2015M1A3A3A05027630) and by the basic research project (UD170052CD) funded by ADD (Agency for Defense Development) and the Administration of Defense Acquisition Program. This work is also supported by the KISTI Supercomputing Center (KSC-2017-G2-0004).

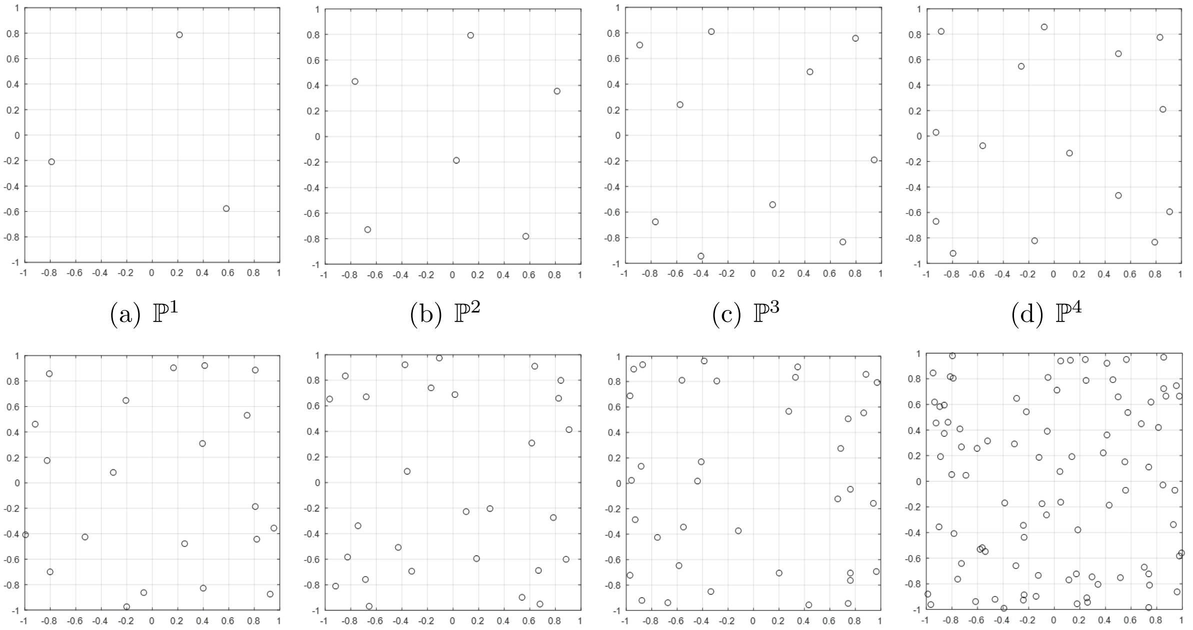
Appendix A Points for Quadrilateral Element

DQM and DRM for the volume integration require a set of points, $\{x_j\}$, that has a sufficiently small condition number for the Vandermonde matrix given by $V_{(i,j)} = \varphi^{(i)}(\mathbf{x}_j)$, where φ is a orthonormal basis for a given target space. In the quadrilateral element, the target space is given by $\mathbb{P}^m([-1,1] \times [-1,1])$. For $m = 1$, a set of points yielding a condition number of unity, which is the minimum possible value, can be found by hand calculation. In order to find the set of points for higher-order space, we apply a sort of brute force approach. With a parallel computing program which randomly searches a set of points that has a well-conditioned Vandermonde matrix, the results are plotted in Fig. 7, after a number of tries of $O(10^9)$ for $\mathbb{P}^2$, and $O(10^{10\sim 11})$ for others, respectively.

REFERENCES

[1] J. S. Hesthaven and T. Warburton, *Nodal Discontinuous Galerkin Methods.* Springer-Verlag Inc., 2008.

[2] F. Bassi, L. Botti, A. Colombo, and S. Rebay, "Agglomeration based discontinuous Galerkin discretization of the Euler and Navier-Stokes equations," *Computers and Fluids*, vol. 61, pp. 77–85, 2012.



(a) $\mathbb{P}^1$ (b) $\mathbb{P}^2$ (c) $\mathbb{P}^3$ (d) $\mathbb{P}^4$

(e) $\mathbb{P}^5$ (f) $\mathbb{P}^6$ (g) $\mathbb{P}^7$ (h) $\mathbb{P}^{12}$

Figure 7: The points for a reference quadrilateral element that empirically minimize a condition number of the Vandermonde matrix.