The Bassi Rebay 1 scheme is a special case of the Symmetric Interior Penalty formulation for discontinuous Galerkin discretisations with Gauss-Lobatto points.

Juan Manzanero[a,*], Andrés M. Rueda-Ramírez[a], Gonzalo Rubio[a,b], Esteban Ferrer[a,b]

[a] *ETSIAE-UPM - School of Aeronautics, Universidad Politécnica de Madrid, Plaza Cardenal Cisneros 3, E-28040 Madrid, Spain.*
[b] *CCS-UPM - Center for Computational Simulation, Universidad Politécnica de Madrid, Boadilla del Monte, 28660 Madrid, Spain.*

Abstract

In the discontinuous Galerkin (DG) community, several formulations have been proposed to solve PDEs involving second-order spatial derivatives (e.g. elliptic problems). In this paper, we show that, when the discretisation is restricted to the usage of Gauss-Lobatto points, there are important similarities between two common choices: the Bassi-Rebay 1 (BR1) method, and the Symmetric Interior Penalty (SIP) formulation. This equivalence enables the extrapolation of properties from one scheme to the other: a sharper estimation of the minimum penalty parameter for the SIP stability (compared to the more general estimate proposed by Shahbazi [1]), more efficient implementations of the BR1 scheme, and the compactness of the BR1 method for straight quadrilateral and hexahedral meshes.

Keywords: Discontinuous Galerkin, Gauss-Lobatto Legendre, Poisson equation, summation-by-parts property

1. Introduction

High order discontinuous Galerkin spectral element methods (DGSEM) have seen an increased popularity in solving partial differential equations (e.g. the Navier-Stokes equations). However, their lack of stability and robustness is still a concern, and the researchers have made efforts towards deriving schemes which discretely satisfy energy [2, 3, 4, 5, 6, 7] or entropy [8, 9] inequalities in order to derive provably stable schemes. One approach considers operators that satisfy the summation-by-parts property, hence, allowing to bound the discrete energy. In a discontinuous Galerkin framework, this can be achieved by selecting Gauss-Lobatto nodes as both interpolation and quadrature points, instead of the more common Gauss-Legendre points.

Several DG formulations exist to solve elliptic problems [10]. Among the most popular ones, we find the Bassi Rebay 1 [11], and the Symmetric Interior Penalty [12] formulations. Both formulations have been used to solve the compressible and incompressible Navier-Stokes equations [13, 9, 14, 15, 16, 17, 18, 19, 20, 21]. These formulations have traditionally been considered different approaches to discretise PDEs involving second order derivatives. However, when restricting the DG discretisation to Gauss-Lobatto points, we find that these two share common patterns, enabling the transfer of properties between formulations. The rest of the paper is organised as follows: in Section 2 the BR1 and SIP configurations are summarised. Then, in Section 3 these formulations are re-formulated for the particular case of Gauss-Lobatto points. Next,

*Corresponding author
Email address: juan.manzanero@upm.es (Juan Manzanero)

in Section 4, the BR1 scheme is shown to become a particular version of the SIP formulation. The main conclusions are gathered in Table 1. For comparison, we have included the estimates derived by Shahbazi [1] for the SIP method to be stable. It can be seen that our results provide lower penalties than Shahbazi's. Lastly, in Section 6, some remarks and consequences of these findings are summarised.

2. Problem description

We consider a nodal discontinuous Galerkin discretisation, applied to a linear Poisson equation:

$$f(\boldsymbol{x}) = \nabla^2 u \text{ in } \Omega, \quad u = 0 \text{ on } \partial\Omega. \tag{1}$$

Following Arnold et al. [10], this equation may be rewritten as a first-order system:

$$\boldsymbol{g} = \nabla u, \quad \nabla \cdot \boldsymbol{g} = f, \tag{2}$$

being $\boldsymbol{g} = \{g_x, g_y\}$ the solution gradient (in Cartesian coordinates). The weak formulation of (2),

$$\begin{aligned} \int_{\Omega^k} \boldsymbol{g} \cdot \boldsymbol{\tau} d\boldsymbol{x} &= \int_{\partial\Omega^k} u^* \boldsymbol{\tau} \cdot \boldsymbol{n} ds - \int_{\Omega^k} u \nabla \cdot \boldsymbol{\tau} d\boldsymbol{x}, \\ \int_{\Omega^k} f v d\boldsymbol{x} &= \int_{\partial\Omega^k} \boldsymbol{g}^* \cdot \boldsymbol{n} v ds - \int_{\Omega^k} \boldsymbol{g} \cdot \nabla v d\boldsymbol{x}, \end{aligned} \tag{3}$$

depends on the numerical flux choice for u^* and $\boldsymbol{g}^*$. In (3), v and $\boldsymbol{\tau}$ are scalar and vectorial test functions respectively, and $\boldsymbol{n}$ is the outward pointing normal vector. Two popular formulations are compared in this work: the Bassi-Rebay 1 (BR1, [11]) method and the Symmetric Interior Penalty (SIP, [22]) formulations. For the solution flux, u^*, both methods use the solution average:

$$u^*_{BR1} = u^*_{SIP} = \{\!\{u\}\!\}, \tag{4}$$

whilst for the gradient flux, $\boldsymbol{g}^*$, BR1 and SIP require:

$$\boldsymbol{g}^*_{BR1} = \{\!\{\boldsymbol{g}\}\!\}, \quad \boldsymbol{g}^*_{SIP} = \{\!\{\nabla u\}\!\} - \sigma [\![u]\!], \tag{5}$$

where $\{\!\{\bullet\}\!\}$ and $[\![\bullet]\!]$ stand for the DG average and jump operators:

$$\{\!\{u\}\!\} = \frac{u^+ + u^-}{2}, \quad [\![u]\!] = u^+ \boldsymbol{n}^+ + u^- \boldsymbol{n}^-, \quad \{\!\{\boldsymbol{g}\}\!\} = \frac{\boldsymbol{g}^+ + \boldsymbol{g}^-}{2}, \quad [\![\boldsymbol{g}]\!] = \boldsymbol{g}^+ \cdot \boldsymbol{n}^+ + \boldsymbol{g}^- \cdot \boldsymbol{n}^-, \tag{6}$$

and symbols + and − account for the left and right elements, as described in Figure 1(b).

3. Description of the DG-GL method in quadrilateral curvilinear meshes

This paper is based on the nodal DG method with tensor product introduced by Black [23]. In brief, the computational domain Ω is tessellated into non-overlapping quadrilateral elements, Ω^k, and the physical coordinates are transformed to a local coordinate system by means of a mapping,

$$\boldsymbol{X}^k = \boldsymbol{X}^k(\xi, \eta), \quad (\xi, \eta) \in [0, 1]^2. \tag{7}$$

In this work, we only consider 2D meshes but the extension of the proof to 3D hexahedral elements is direct. The mapping for the Laplace operator is performed using curvilinear bases as in [24]. These bases define a local curvilinear coordinate system:

$$\boldsymbol{a}_j = \frac{\partial \boldsymbol{X}^k}{\partial \xi_j}, \quad J\boldsymbol{a}^k = \boldsymbol{a}_i \times \boldsymbol{a}_j, \quad (i, j, k) \text{ cyclical}, \tag{8}$$

which is used to map the operators. Under these mappings, the gradient and divergence operators become:

$$\nabla u = \frac{1}{J}\sum_{d=1}^{N_D}\frac{\partial}{\partial \xi_d}\big(J\boldsymbol{a}^{\xi_d}u\big), \quad \nabla\cdot\boldsymbol{g} = \frac{1}{J}\sum_{d=1}^{N_D}\frac{\partial}{\partial \xi_d}\big(J\boldsymbol{a}^{\xi_d}\cdot\boldsymbol{g}\big). \tag{9}$$

Defining the nabla operator in the local basis, $\tilde{\nabla} = \big(\partial_\xi, \partial_\eta\big)$, the previous expressions can be seen as the divergence of the tensor $\tilde{\boldsymbol{F}}$, defined as:

$$\tilde{\boldsymbol{F}} = u\big[J\boldsymbol{a}^{\xi}, J\boldsymbol{a}^{\eta}\big], \tag{10}$$

and the contravariant gradients, $\tilde{\boldsymbol{g}}$, defined as:

$$\tilde{\boldsymbol{g}} = \boldsymbol{g}^T\cdot\big[J\boldsymbol{a}^{\xi}, J\boldsymbol{a}^{\eta}\big], \tag{11}$$

such that:

$$J\nabla u = \tilde{\nabla}\cdot\tilde{\boldsymbol{F}}, \quad J\nabla\cdot\boldsymbol{g} = \tilde{\nabla}\cdot\tilde{\boldsymbol{g}}. \tag{12}$$

With this new set of variables, the original system (2) is transformed in the first order system as described in [11, 10]:

$$J\boldsymbol{g} = \tilde{\nabla}\cdot\tilde{\boldsymbol{F}}, \quad Jf(x,y) = \tilde{\nabla}\cdot\tilde{\boldsymbol{g}}. \tag{13}$$

The solution is approximated in each element Ω^k by Nth degree polynomials (referred to with the sub-index h: u_h):

$$u_h^k(\xi,\eta;t) = \sum_{i=0}^{N}\sum_{j=0}^{N}u_{ij}^k(t)l_i(\xi)l_j(\eta). \tag{14}$$

The nodal version of both $\tilde{\boldsymbol{F}}$ and $\tilde{\boldsymbol{g}}$ are constructed with the Lagrange interpolation operator (see [25]):

$$\tilde{\boldsymbol{F}}_h = \sum_{i,j=0}^{N}u_{ij}\big[J\boldsymbol{a}_{ij}^{\xi}, J\boldsymbol{a}_{ij}^{\eta}\big]l_i(\xi)l_j(\eta), \quad \tilde{\boldsymbol{g}}_h = \sum_{i,j=0}^{N}\boldsymbol{g}_{ij}^T\cdot\big[J\boldsymbol{a}_{ij}^{\xi}, J\boldsymbol{a}_{ij}^{\eta}\big]l_i(\xi)l_j(\eta). \tag{15}$$

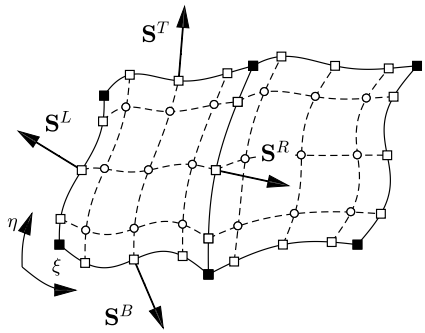
3.1. Comments regarding the mesh construction

The methodology presented herein assumes a *well-constructed mesh* (see [24]). This is essential to uniquely define the mapping terms at the inter-element shared edges, which prevents mapping discontinuities across adjacent cells. To achieve this condition, the evaluation of the interpolated mappings on the edges should be identical on adjacent elements. We define the mapping for a general curvilinear element as:

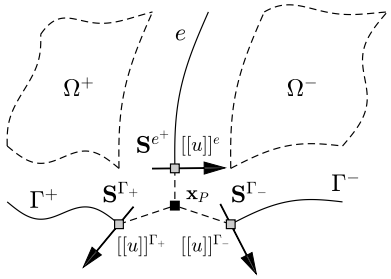
$$\begin{aligned}\boldsymbol{X}(\xi,\eta) = {}& \boldsymbol{\Gamma}_B(\xi)(1-\eta) + \boldsymbol{\Gamma}_R(\eta)\xi + \boldsymbol{\Gamma}_T(\xi)\eta + \boldsymbol{\Gamma}_L(\eta)(1-\xi)\\ & - \boldsymbol{x}_1(1-\xi)(1-\eta) - \boldsymbol{x}_2\xi(1-\eta) - \boldsymbol{x}_3\xi\eta - \boldsymbol{x}_4(1-\xi)\eta.\end{aligned}, \tag{16}$$

where $\boldsymbol{\Gamma}_B$, $\boldsymbol{\Gamma}_R$, $\boldsymbol{\Gamma}_T$, $\boldsymbol{\Gamma}_L$ are the curves that describe the bottom, right, top, and left edges, respectively, and $\boldsymbol{x}_i$ is the position of the i corner of the quadrilateral.

The mapping interpolant, $\boldsymbol{X}_h$, is obtained using interpolation with Nth degree polynomials. Since this work considers Gauss-Lobatto points, the interpolants will automatically match the original curves at their end-points (e.g. $\boldsymbol{\Gamma}_B(0) = \boldsymbol{x}_1$). When the Nth degree curve interpolants, $\boldsymbol{\Gamma}_h$, are introduced in (16), the Nth degree mapping interpolant, $\boldsymbol{X}_h$, is obtained. If the mapping is constructed following this procedure, it will satisfy continuity across elements. Additionally, we note that maintaining continuity ensures identical



(a) Mappings, curvilinear basis and interpolation nodes. Dashed lines represent $\xi = const.$ and $\eta = const.$. according to the curvilinear basis represented. The four normal vectors are a prolongation of $\nabla\xi$ and $\nabla\eta$ lines (orthogonal to the dashed lines). Empty circles represent interior nodes, empty squares represent edge-interior nodes, and filled squares represent edge end-points.



(b) Illustration of the edges that concur in an edge end-point, P. The edge e is shared by the elements Ω^+ and Ω^-, the edge Γ^+ belongs to the element Ω^+, and the edge Γ^- belongs to the element Ω^-. All edges (e, Γ^+ and Γ^-) meet in P. The three normal vectors, $\boldsymbol{S}^{e^+}$, $\boldsymbol{S}^{\Gamma_+}$ and $\boldsymbol{S}^{\Gamma_-}$, are represented. Notice that the jumps of the solution , $[\![u]\!]^e$, $[\![u]\!]^{\Gamma_+}$ and $[\![u]\!]^{\Gamma_-}$, follow their direction, respectively.

Figure 1: Description of a general curvilinear quadrilateral mesh.

normal vectors on both sides of a surface. The surface Jacobian vectors, $\boldsymbol{S}$, are obtained as a result of the prolongation to edges of the mapping derivatives, as described in [26]. For instance:

$$\begin{aligned} \boldsymbol{S}^B(\xi) &= -\nabla\eta = -J\boldsymbol{a}^\eta(\xi,0), \quad \boldsymbol{S}^R(\eta) = \nabla\xi = J\boldsymbol{a}^\xi(1,\eta), \\ \boldsymbol{S}^L(\eta) &= -\nabla\xi = -J\boldsymbol{a}^\xi(0,\eta), \quad \boldsymbol{S}^T(\xi) = \nabla\eta = J\boldsymbol{a}^\eta(\xi,1), \end{aligned} \tag{17}$$

where B,R,T, and L refer to the bottom, right, top, and left edges, respectively. Due to the construction of (16) with Gauss-Lobatto points, normal vectors obtained as a prolongation of the contravariant basis, $J\boldsymbol{a}^{\xi_j}$, will be identical when computed from both sides of a shared edge. If the discretisation is non conforming, it is necessary to adapt the interpolation curves, as described in [27]. Additionally, we define the jump operators referred to the local basis, $[\![u]\!]_{\hat{\boldsymbol{n}}}$ as:

$$[\![u]\!]_{\hat{\boldsymbol{n}}} = u^+\hat{\boldsymbol{n}}^+ + u^-\hat{\boldsymbol{n}}^-, \quad [\![\boldsymbol{g}]\!]_{\hat{\boldsymbol{n}}} = \boldsymbol{g}^+\cdot\hat{\boldsymbol{n}}^+ + \boldsymbol{g}^-\cdot\hat{\boldsymbol{n}}^-, \tag{18}$$

which yield a compact notation for the following expressions. Notice that, with this notation, it follows:

$$\begin{aligned} [\![u]\!]\cdot\boldsymbol{S}^+ &= \left(u^+\boldsymbol{n}^+ + u^-\boldsymbol{n}^-\right)\cdot\boldsymbol{S}^+ = \|\boldsymbol{S}^+\|(u^+ - u^-) \\ &= \|\boldsymbol{S}^+\|(u^+\hat{\boldsymbol{n}}^+ + u^-\hat{\boldsymbol{n}}^-)\cdot\hat{\boldsymbol{n}}^+ = \|\boldsymbol{S}^+\|\left([\![u]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}^+\right), \end{aligned} \tag{19}$$

where $\boldsymbol{n}^\pm = \boldsymbol{S}^\pm/\|\boldsymbol{S}^\pm\|$.

4. BR1 and SIP formulations on DG-GL quadrilateral curvilinear meshes

In this section show the similarities between the BR1 and SIP formulations applied to the discretisation of the Poisson equation, (13). Following [10], the first equation in (13) (i.e. the equation governing the solution gradient) is multiplied by a test function $\boldsymbol{\tau}_h$ and integrated by parts in element Ω^k:

$$\int_{\Omega^k} J\boldsymbol{g}_h\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi} = \int_{\partial\Omega^k}\boldsymbol{\tau}_h\cdot\tilde{\boldsymbol{F}}^*\cdot\hat{\boldsymbol{n}}d\xi - \int_{\Omega^k}\tilde{\nabla}\boldsymbol{\tau}_h : \tilde{\boldsymbol{F}}_h d\boldsymbol{\xi}, \tag{20}$$

where $\tilde{\boldsymbol{F}}^*$ is the solution numerical flux, $\hat{\boldsymbol{n}}$ is the normal vector in the local basis (i.e. pointing outwards in the reference domain), and ":" denotes the tensor double inner product (i.e. $\boldsymbol{A}:\boldsymbol{B}=\sum_{\forall i,j}A_{ij}B_{ij}$). Subsequently, the integrals in (20) are replaced by numerical quadrature rules based on Gauss-Lobatto points. The discrete integral is denoted $\int^N$. Next, (20) is summed across all mesh elements:

$$\int_\Omega^N J\boldsymbol{g}_h\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi}=\int_{\mathcal{E}}^N\big(\{\!\{\boldsymbol{\tau}_h\}\!\}\cdot[\![\tilde{\boldsymbol{F}}^*]\!]_{\hat{\boldsymbol{n}}}+\{\!\{\tilde{\boldsymbol{F}}^*\}\!\}:[\![\boldsymbol{\tau}_h]\!]_{\hat{\boldsymbol{n}}}\big)d\xi-\int_\Omega^N\tilde{\nabla}\boldsymbol{\tau}_h:\tilde{\boldsymbol{F}}_h d\boldsymbol{\xi},\tag{21}$$

where the surface integral is obtained as a result of each two adjacent element contributions to a single face (see [10, 28]). Furthermore, $\mathcal{E}$ refers to the set of interior and boundary faces in which the domain Ω is divided. The DG average and jump operators, (6), have been adapted to work with tensors:

$$\begin{aligned}\{\!\{\boldsymbol{\tau}\}\!\}&=\frac{\boldsymbol{\tau}^++\boldsymbol{\tau}^-}{2},\ [\![\boldsymbol{\tau}]\!]_{\hat{\boldsymbol{n}}}=\boldsymbol{\tau}^+\otimes\hat{\boldsymbol{n}}^++\boldsymbol{\tau}^-\otimes\hat{\boldsymbol{n}}^-,\\ \{\!\{\tilde{\boldsymbol{F}}\}\!\}&=\frac{\tilde{\boldsymbol{F}}^++\tilde{\boldsymbol{F}}^-}{2},\ [\![\tilde{\boldsymbol{F}}]\!]_{\hat{\boldsymbol{n}}}=\tilde{\boldsymbol{F}}^+\cdot\hat{\boldsymbol{n}}^++\tilde{\boldsymbol{F}}^-\cdot\hat{\boldsymbol{n}}^-.\end{aligned}\tag{22}$$

Using the summation-by-parts property (see [2, 29, 30]), (21) can be discretely integrated by parts again, to obtain:

$$\int_\Omega^N J\boldsymbol{g}_h\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi}=\int_{\mathcal{E}}^N\big(\{\!\{\boldsymbol{\tau}_h\}\!\}\cdot[\![\tilde{\boldsymbol{F}}^*-\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}}+\{\!\{\tilde{\boldsymbol{F}}^*-\tilde{\boldsymbol{F}}_h\}\!\}:[\![\boldsymbol{\tau}_h]\!]_{\hat{\boldsymbol{n}}}\big)d\xi+\int_\Omega^N\big(\tilde{\nabla}\cdot\tilde{\boldsymbol{F}}_h\big)\boldsymbol{\tau}_h d\xi.\tag{23}$$

Both BR1 and SIP schemes use the averaged solution as the interface flux (i.e. $\tilde{\boldsymbol{F}}^*=\{\!\{\tilde{\boldsymbol{F}}_h\}\!\}$). Hence, the term $[\![\tilde{\boldsymbol{F}}^*]\!]_{\hat{\boldsymbol{n}}}$ vanishes, and so does the second term in the surface integral, $\{\!\{\tilde{\boldsymbol{F}}^*-\tilde{\boldsymbol{F}}_h\}\!\}$, such that:

$$\int_\Omega^N J\boldsymbol{g}_h\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi}=-\int_{\mathcal{E}}^N\{\!\{\boldsymbol{\tau}_h\}\!\}\cdot[\![\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}}d\xi+\int_\Omega^N\big(\tilde{\nabla}\cdot\tilde{\boldsymbol{F}}_h\big)\boldsymbol{\tau}_h d\xi.\tag{24}$$

The contribution of every element, Ω^k, to (24) can be expressed as:

$$\int_{\Omega^k}^N J\boldsymbol{g}_h\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi}=-\frac{1}{2}\int_{\partial\Omega^k}^N\boldsymbol{\tau}_h\cdot[\![\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}}d\xi+\int_{\Omega^k}^N\big(\tilde{\nabla}\cdot\tilde{\boldsymbol{F}}_h\big)\boldsymbol{\tau}_h d\xi.\tag{25}$$

Since (25) must hold for any $\boldsymbol{\tau}_h$, it must hold independently for the basis functions in each of the Gauss-Lobatto points. Therefore, it is possible to replace the test function by the product of the Lagrange polynomials $\boldsymbol{\tau}_h=l_i(\xi)l_j(\eta)\boldsymbol{e}_d$ (being $\boldsymbol{e}_d$ the d unitary Cartesian direction). After applying the numerical quadrature, the contribution of every Gauss-Lobatto point to the volume integrals yields:

$$\int_{\Omega^k}^N J\boldsymbol{g}_h^k\cdot\boldsymbol{\tau}_h d\boldsymbol{\xi}=w_iw_jJ_{ij}^k\boldsymbol{g}_{ij}^k,\ \int_{\Omega^k}^N\big(\tilde{\nabla}\cdot\tilde{\boldsymbol{F}}_h^k\big)\boldsymbol{\tau}_h d\xi=w_iw_jJ_{ij}^k\nabla u_{h,ij}^k.\tag{26}$$

On the other hand, since the product $\tilde{\boldsymbol{F}}\cdot\hat{\boldsymbol{n}}$ defines the solution multiplied by the normal vector, we can write $[\![\tilde{\boldsymbol{F}}]\!]_{\hat{\boldsymbol{n}}}=\big([\![u]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)\boldsymbol{S}$, being $\boldsymbol{S}$ one of the surface Jacobian defined in (17). As a consequence, the contribution of every Gauss-Lobatto point to the surface integral yields:

$$\begin{aligned}\int_{\partial\Omega^k}^N\{\!\{\boldsymbol{\tau}_h\}\!\}\cdot[\![\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}}d\xi=&\frac{1}{2}\int_{\partial\Omega^k}^N\big([\![u_h^k]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)\boldsymbol{\tau}_h\cdot\boldsymbol{S}^k d\xi\\ =&\ \frac{1}{2}\big([\![u_h]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)_i^{k,B}\delta_{j0}w_i\boldsymbol{S}_i^{k,B}+\frac{1}{2}\big([\![u_h]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)_i^{k,T}\delta_{jN}w_i\boldsymbol{S}_i^{k,T}\\ &+\frac{1}{2}\big([\![u_h]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)_j^{k,L}\delta_{i0}w_j\boldsymbol{S}_j^{k,L}+\frac{1}{2}\big([\![u_h]\!]_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\big)_j^{k,R}\delta_{iN}w_j\boldsymbol{S}_j^{k,R},\end{aligned}\tag{27}$$

where the assumption of a *well-constructed* mesh allows to replace the prolongation of $J\boldsymbol{a}^{\xi_j}$ by the appropriate surface Jacobian (17).

To compute the second equation in (13), we proceed as in (20), i.e. the weak formulation is constructed multiplying by a test function v and integrating by parts once in an element:

$$\int_{\Omega^k} Jfvd\boldsymbol{\xi} = \int_{\partial\Omega^k} \boldsymbol{g^*}\cdot\boldsymbol{S}vd\xi - \int_{\Omega^k} \tilde{\boldsymbol{g}}\cdot\tilde{\nabla}vd\boldsymbol{\xi}, \tag{28}$$

where the surface integral has been rewritten using the equivalence between local and physical coordinates, $\tilde{\boldsymbol{g}}^*\cdot\hat{\boldsymbol{n}} = \boldsymbol{g}^*\cdot\boldsymbol{S}$. Following (28), the study of BR1 and SIP schemes requires the prolongation of $\boldsymbol{g}$ to the edges to obtain the numerical flux $\boldsymbol{g^*}$. Notice that the volume term is common in both techniques. For the sake of simplicity, and without loss of generality, only the left face of element k is considered. Due to the Gauss-Lobatto interpolation, this is achieved evaluating one nodal value (i.e. setting $i=0$). Hence, we replace (26) and (27) in (24) and evaluate $g_{L,j}$:

$$\begin{aligned}\boldsymbol{g}^k_{L,j} = \boldsymbol{g}^k_{0j} = \nabla u^k_{h,0j} &- \frac{1}{2J^k_{0j}w_0}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^L_j\boldsymbol{S}^L_j \\ &- \frac{1}{2}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^B_0\frac{\delta_{j0}}{J^k_{0j}w_j}\boldsymbol{S}^B_0 - \frac{1}{2}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^T_0\frac{\delta_{jN}}{J^k_{0j}w_j}\boldsymbol{S}^T_0.\end{aligned} \tag{29}$$

Notice that, for interior edge points (i.e. non-corner points, $j\neq 0, N$), (29) reduces to:

$$\boldsymbol{g}^k_{L,j} = \nabla u^k_{h,0j} - \frac{1}{2J^k_{0j}w_0}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^L_j\boldsymbol{S}^L_j, \tag{30}$$

whilst at the corners (e.g. $j=0$), the jumps from the two adjacent faces are added to (30):

$$\boldsymbol{g}^k_{L,0} = \nabla u^k_{h,00} - \frac{1}{2J^k_{00}w_0}\left[\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^L_0\boldsymbol{S}^L_0 + \frac{1}{2}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^B_0\boldsymbol{S}^B_0\right]. \tag{31}$$

From the gradients prolongation to the faces, we can write the BR1 flux. Inserting (30) into (5), for an edge-interior point (empty squares in Fig. 1(a)), we obtain:

$$\boldsymbol{g}^*_{BR1}\cdot\boldsymbol{S} = \{\!\{\nabla u_h\}\!\}\cdot\boldsymbol{S} - \frac{\|\boldsymbol{S}\|^2}{2w_0}\{\!\{J^{-1}\}\!\}\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right). \tag{32}$$

Similarly, following (5), the SIP flux, $\boldsymbol{g}^*_{SIP}$, is:

$$\boldsymbol{g}^*_{SIP}\cdot\boldsymbol{S} = \{\!\{\nabla u_h\}\!\}\cdot\boldsymbol{S} - \sigma\|\boldsymbol{S}\|\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right), \tag{33}$$

where we used (19) to relate the jump operator in the physical and computational domains. Finally, we find that, for face-interior points, the BR1 and SIP formulations are equivalent when the penalty parameter σ is:

$$\sigma = \sigma^\star_{BR1} = \frac{N(N+1)}{2}\|\boldsymbol{S}\|\{\!\{J^{-1}\}\!\}, \tag{34}$$

which only depends on the mesh geometry and the polynomial order, since the quadrature weight $w_0 = w_N = 1/N(N+1)$ for Gauss-Lobatto points in $[0,1]$. For the edge end-points (i.e. corners), we take (31) and consider the contribution of the faces shared by the node. As depicted in Fig. 1(b), naming Γ^+ and Γ^- the two faces shared by the elements Ω^+ and Ω^- we find:

$$\boldsymbol{g^*}_{BR1}\cdot\boldsymbol{S}^e = \{\!\{\nabla u\}\!\}\cdot\boldsymbol{S}^e - \frac{\|\boldsymbol{S}^e\|^2}{2w_0}\{\!\{J^{-1}\}\!\}^e\left[\left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^e + \left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^{\Gamma^+}\frac{\boldsymbol{S}^e\cdot\boldsymbol{S}^{\Gamma^+}}{2\|\boldsymbol{S}^e\|^2} + \left(\llbracket u_h\rrbracket_{\hat{\boldsymbol{n}}}\cdot\hat{\boldsymbol{n}}\right)^{\Gamma^-}\frac{\boldsymbol{S}^e\cdot\boldsymbol{S}^{\Gamma^-}}{2\|\boldsymbol{S}^e\|^2}\right]. \tag{35}$$

We conclude that the penalty parameter remains the same as in (34), but the jumps are modified. The SIP method considers the jumps across shared faces, whilst the BR1 method adds the projection to the edge of the jumps across both Γ^+ and Γ^-, as described in Fig. 1(b). Let us note that these extra terms are responsible for the non-compactness of the BR1 method. These terms vanish if the faces cross at 90^o, such that $\boldsymbol{S}^e \cdot \boldsymbol{S}^{\Gamma^\pm} = 0$. This is true for Cartesian meshes, and also for one-dimensional discretisations. If these conditions are satisfied, the BR1 and SIP schemes are algebraically identical as long as the penalty parameter for the SIP method is chosen following (34). The results found are summarised in Table 1.

Table 1: Comparison of the architecture of BR1 and SIP formulations with Gauss-Lobatto points. The estimate for the penalty parameter from Shahbazi [1] is included for comparison. Note that Shahbazi's estimate holds only for straight-sided elements, whilst the penalty parameter (with associated numerical fluxes) derived here suits elements with arbitrary curvilinear geometries.

Face-interior	SIP	BR1	(Shahbazi's [1] estimation)
Penalty parameter σ	σ	$\frac{N(N+1)}{2}\|\boldsymbol{S}\|\{\!\{J^{-1}\}\!\}$	$\frac{(N+1)(N+2)}{2}\|\boldsymbol{S}\|\max(J^{-1,+}, J^{-1,-}))$
Jump operator $[\![\bullet]\!]_{\hat{\boldsymbol{n}}}$	$[\![u]\!]_{\hat{\boldsymbol{n}}}$	$[\![u]\!]_{\hat{\boldsymbol{n}}}$	

Face end-point	SIP	BR1	(Shahbazi [1] estimation)
Penalty parameter σ	σ	$\frac{N(N+1)}{2}\|\boldsymbol{S}\|\{\!\{J^{-1}\}\!\}$	$\frac{(N+1)(N+2)}{2}\|\boldsymbol{S}\|\max(J^{-1,+}, J^{-1,-}))$
Jump operator $[\![\bullet]\!]_{\hat{\boldsymbol{n}}}$	$[\![u]\!]^e_{\hat{\boldsymbol{n}}}$	$[\![u]\!]^e_{\hat{\boldsymbol{n}}} + [\![u_h]\!]^{\Gamma^+}_{\hat{\boldsymbol{n}}} \frac{\boldsymbol{S}^e \cdot \boldsymbol{S}^{\Gamma^+}}{2\|\boldsymbol{S}^e\|^2} + [\![u_h]\!]^{\Gamma^-}_{\hat{\boldsymbol{n}}} \frac{\boldsymbol{S}^e \cdot \boldsymbol{S}^{\Gamma^-}}{2\|\boldsymbol{S}^e\|^2}$	

5. Numerical experiments

In this section we investigate the accuracy of the previous theoretical findings. We study the difference between BR1 and SIP operators, applied to the two-dimensional Poisson equation with Dirichlet boundary conditions:

$$\begin{aligned} \nabla^2 u &= 0, \quad (x, y) \in \Omega([0,1]^2) \\ u(\partial\Omega) &= u_D. \end{aligned} \tag{36}$$

The study performed in Section 4 only referred to interior faces. Nevertheless, the study of Dirichlet boundary conditions follows similar steps and is included in Appendix Appendix A. We solve (36) in a two-dimensional structured mesh with 4^2 straight-sided elements and polynomial order $N = 3$. For this configuration and following Table 1, we obtain $\sigma_{BR1} = 24$ and $\sigma_{SHAH} = 40$. To show the similarities of both methods, we compare the analytical Jacobian of the discrete Laplacian operators. Figure 2 shows the relative error between the BR1 and the SIP Jacobian matrices, measured using the L^2 norm. When considering the penalty parameter (σ_{BR1}) as proposed in this work, we observe that both methods are identical to machine precision. Furthermore, to provide further insight into the effect of the penalty parameter in the Jacobian matrix, we show the eigenvalue spectra in Figure 3. Note that, since both discretisations are

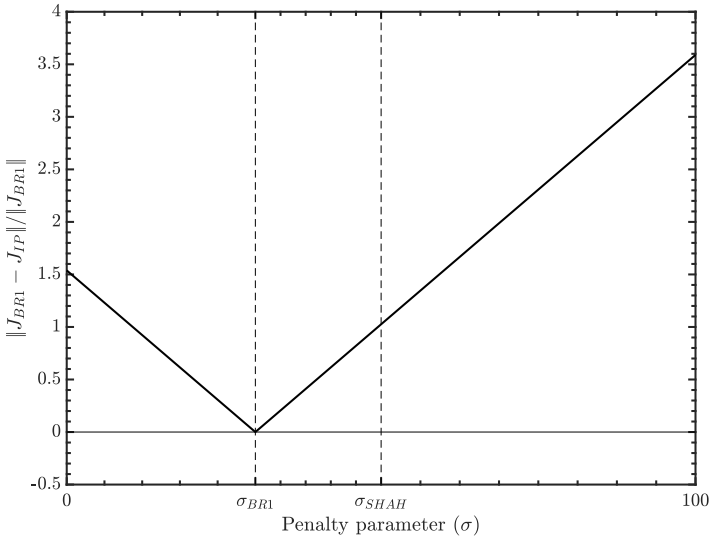


Figure 2: L^2 norm of the error of the BR1 and the SIP Jacobian matrices when varying the penalty parameter (σ). We have highlighted our estimate (σ_{BR1}) and Shahbazi's estimate (σ_{SHAH}), confirming that both techniques are equivalent when the penalty parameter is σ_{BR1}.

symmetric, the eigenvalues are strictly real (i.e. without imaginary part associated to dispersion errors). We have considered the SIP without stabilization ($\sigma = 0$), which yields positive eigenvalues, confirming that the SIP is unstable when under-penalized, as described in [31]. Additionally, we observe that BR1 and SIP (with σ_{BR1}) have identical eigenvalues. The SIP method with Shahbazi's estimate for the penalty parameter is also represented, as an example of an over-penalized SIP approach. We note that increasing the penalty parameter above our estimate (σ_{BR1}) tends to increase the condition number of the Jacobian, without affecting the stability of the method. These results are complemented by the evolution of the largest SIP eigenvalue, when increasing the penalty parameter (see Figure 4). This eigenvalue is positive (unstable discretisation) for penalty parameters below our estimate σ_{BR1}, and negative (stable discretisation) for penalty parameters above the estimate (e.g. Shahbazi's estimate).

6. Summary and conclusions

The main conclusions are:

- The Bassi-Rebay 1 (BR1) scheme can be regarded as a particular version of the Symmetric Interior Penalty (SIP) method since the numerical fluxes are computed as the average of the solution gradient, $\{\!\{\nabla u_h\}\!\}$, plus a penalty term that depends on the interface jumps. This equivalence is consistent with the identical rates of convergence observed for both methods ([10]).
- The estimation of the minimum penalty parameter required to obtain a coercive (and stable) SIP scheme is sharper than the one provided by Shahbazi [1]. However, the bound provided by Shahbazi is general, since no assumption is made regarding the node distribution, whilst our work is limited to Gauss-Lobatto points (and satisfy the summation by part property).
- Gassner et al. [9] proved that a stable method can be derived by using a DG-BR1 discretisation of a general advection-diffusion equation. In this note, we have shown that the SIP method is an over-penalised BR1, for a particular set of points and type of elements. Combining these two results, it may be concluded that a DG-SIP discretisation, of the linear advection-diffusion system, is also stable with Gauss-Lobatto points, and Cartesian meshes.

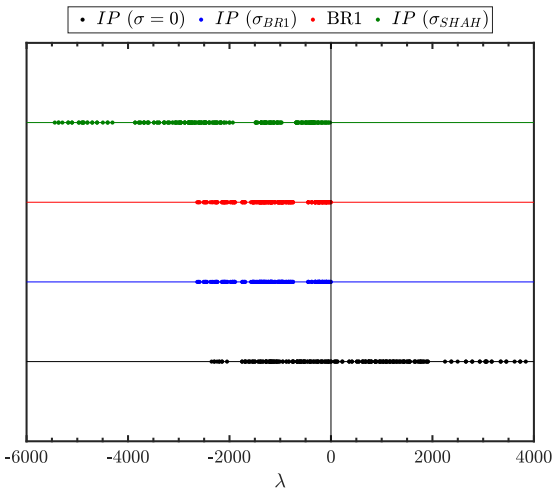


Figure 3: Eigenvalue spectra of the BR1 and SIP Jacobian matrices. The penalty parameters considered for the SIP are $\sigma = 0, \sigma_{BR1}$ and σ_{SHAH}. Note that all eigenvalues lay on the real axis (i.e. the discretisation does not introduce any dispersion error). We show that the SIP method with $\sigma = 0$ is unstable, that both BR1 and SIP (with σ_{BR1}) eigenvalues are identical and that Shahbazi's estimate decreases the minimum eigenvalue significantly, which increases the condition number of the Jacobian.

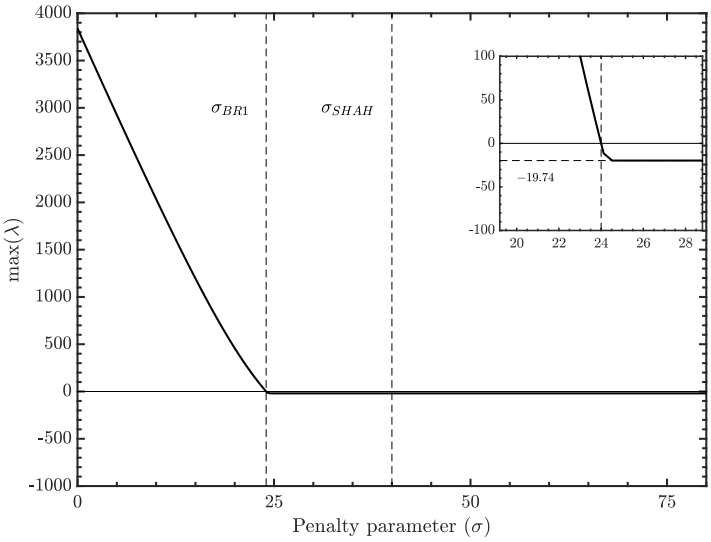


Figure 4: Maximum eigenvalue obtained from the SIP Jacobian as a function of σ. We show that for $\sigma < \sigma_{BR1}$ the scheme is unstable and stable for $\sigma > \sigma_{BR1}$.

- Since the BR1 method is a particular version of the Symmetric Interior Penalty formulation, this work proposes an alternative way to implement the BR1 method using a SIP formulation.

Acknowledgements

The author acknowledges the computer resources and technical assistance provided by the Centro de Supercomputación y Visualización de Madrid (CeSViMa).

Appendix A. Penalty parameter in Dirichlet boundary conditions faces

In this section we derive the SIP penalty parameter σ that mimics the BR1 scheme at Dirichlet boundary faces. For the sake of brevity, we will only consider Cartesian elements (i.e. without corner cross-terms), but the generalisation for curvilinear elements follows the reasoning of Section 4. In both BR1 and SIP schemes, the solution numerical flux $u^\star$ for faces with Dirichlet boundary conditions coincides with the boundary value u_D:

$$u^\star_{BR1} = u^\star_{IP} = u_D. \tag{A.1}$$

Hence, we can consider the gradient equation in all elements, (21) with the contribution of Dirichlet boundary condition faces:

$$\int_\Omega^N J\boldsymbol{g}_h \cdot \boldsymbol{\tau}_h d\boldsymbol{\xi} = \int_\mathcal{E}^N \big(\{\!\{\boldsymbol{\tau}_h\}\!\} \cdot [\![\tilde{\boldsymbol{F}}^*]\!]_{\hat{\boldsymbol{n}}} + \{\!\{\tilde{\boldsymbol{F}}^*\}\!\} : [\![\boldsymbol{\tau}_h]\!]_{\hat{\boldsymbol{n}}}\big) d\xi + \int_{\partial\Omega_D}^N \boldsymbol{\tau} \cdot \boldsymbol{F}_D \cdot \hat{\boldsymbol{n}} d\xi - \int_\Omega^N \tilde{\nabla}\boldsymbol{\tau}_h : \tilde{\boldsymbol{F}}_h d\boldsymbol{\xi}, \tag{A.2}$$

where $\boldsymbol{F}_D = u_D[J\boldsymbol{a}^\xi, J\boldsymbol{a}^\eta]$ results from the imposition of Dirichlet boundary conditions (A.1). Following (24), we integrate (A.2) by parts to obtain:

$$\int_\Omega^N J\boldsymbol{g}_h \cdot \boldsymbol{\tau}_h d\boldsymbol{\xi} = -\int_\mathcal{E}^N \{\!\{\boldsymbol{\tau}_h\}\!\} \cdot [\![\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}} d\xi - \int_{\partial\Omega_D}^N \boldsymbol{\tau}_h \cdot \big(\tilde{\boldsymbol{F}}_h - \tilde{\boldsymbol{F}}_D\big) \cdot \hat{\boldsymbol{n}} d\xi + \int_\Omega^N \big(\tilde{\nabla} \cdot \tilde{\boldsymbol{F}}_h\big)\boldsymbol{\tau}_h d\xi. \tag{A.3}$$

Setting (A.3) for a particular boundary element Ω_k such that $\partial\Omega_k \cap \partial\Omega_D \neq \varnothing$:

$$\int_{\Omega_k}^N J\boldsymbol{g}_h \cdot \boldsymbol{\tau}_h d\boldsymbol{\xi} = -\frac{1}{2}\int_{\partial\Omega_k \setminus \{\partial\Omega_k \cap \partial\Omega_D\}}^N \boldsymbol{\tau}_h \cdot [\![\tilde{\boldsymbol{F}}_h]\!]_{\hat{\boldsymbol{n}}} d\xi - \int_{\partial\Omega_k \cap \partial\Omega_D}^N \boldsymbol{\tau}_h \cdot \big(\tilde{\boldsymbol{F}}_h - \tilde{\boldsymbol{F}}_D\big) \cdot \hat{\boldsymbol{n}} d\xi + \int_\Omega^N \big(\tilde{\nabla} \cdot \tilde{\boldsymbol{F}}_h\big)\boldsymbol{\tau}_h d\xi, \tag{A.4}$$

we find that interior faces are weighted with 1/2 whilst external boundary faces are not. Following (29), we prolong the lifted gradient $\boldsymbol{g}$ to a Dirichlet boundary face (e.g. a left face) to find:

$$\boldsymbol{g}^k_{L,j} = \nabla u^k_{h,0j} - \frac{1}{J^k_{0j} w_0} \big(u_{0j} - u_D\big)^L_j \boldsymbol{S}^L_j. \tag{A.5}$$

Regarding the Poisson equation, gradient fluxes (5) for Dirichlet boundary conditions become:

$$\boldsymbol{g}^*_{BR1} = \boldsymbol{g}_h, \quad \boldsymbol{g}^*_{SIP} = \nabla u_h - \sigma(u_h - u_D)\boldsymbol{n}. \tag{A.6}$$

Replacing (A.5) in the BR1 flux, we obtain:

$$\boldsymbol{g^*}_{BR1} \cdot \boldsymbol{S} = \{\!\{\nabla u_h\}\!\} \cdot \boldsymbol{S} - \frac{\|\boldsymbol{S}\|^2}{w_0} \{\!\{J^{-1}\}\!\}\big(u_h - u_D\big), \tag{A.7}$$

and

$$\boldsymbol{g^*}_{SIP} \cdot \boldsymbol{S} = \{\!\{\nabla u_h\}\!\} \cdot \boldsymbol{S} - \sigma\|\boldsymbol{S}\|\big(u_h - u_D\big). \tag{A.8}$$

Comparing (A.7) and (A.8), we find that the penalty parameter for Dirichlet boundary condition faces is the double that of interior faces:

$$\sigma^{D}_{BR1} = 2\sigma_{BR1} = N(N+1)\|\boldsymbol{S}\|\{\!\{J^{-1}\}\!\}. \tag{A.9}$$

The requirement to double the penalty parameter at Dirichlet boundaries, in the SIP method, was already introduced in [32, 31] and references therein.

References

References

[1] K. Shahbazi, Short note: An explicit expression for the penalty parameter of the interior penalty method, Journal of Computational Physics 205 (2) (2005) 401–407.

[2] G.J. Gassner, A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods, SIAM Journal on Scientific Computing 35 (3) (2013) 1233–1256.

[3] G. J. Gassner, A kinetic energy preserving nodal discontinuous Galerkin spectral element method, International Journal for Numerical Methods in Fluids 76 (1) (2014) 28–50.

[4] D.A. Kopriva and G.J. Gassner, An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems, SIAM Journal on Scientific Computing 36 (4) (2014) A2076–A2099.

[5] D. A. Kopriva, A. R. Winters, M. Bohm, G. J. Gassner, A provably stable discontinuous Galerkin spectral element approximation for moving hexahedral meshes, Computers & Fluids 139 (Supplement C) (2016) 148 – 160, 13th USNCCM International Symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 60th birthday of Professor David Kopriva.

[6] J. Manzanero, G. Rubio, E. Ferrer, and E. Valero, Dispersion-dissipation analysis for advection problems with non-constant coefficients: Application to discontinuous Galerkin formulations, In press, SIAM Journal of Scientific Computing (2017).

[7] J. Manzanero, G. Rubio, E. Ferrer, E. Valero, D.A. Kopriva, Insights on aliasing driven instabilities for advection equations with application to Gauss-Lobatto discontinuous Galerkin methods, Journal of Scientific Computing (2017).doi:https://doi.org/10.1007/s10915-017-0585-6.

[8] G. J. Gassner, A. R. Winters, D. A. Kopriva, Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations, Journal of Computational Physics 327 (Supplement C) (2016) 39 – 66.

[9] G.J. Gassner, A.R. Winters, F.J. Hindenlang, and D.A. Kopriva, The BR1 scheme is stable for the compressible Navier-Stokes equations, arXiv preprint arXiv:1704.03646.

[10] D.N. Arnold, F. Brezzi, B. Cockburn, and L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM Journal on Numerical Analysis 39 (5) (2001) 1749–1779.

[11] F. Bassi and S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, Journal of Computational Physics 131 (2) (1997) 267 – 279.

[12] M. F. Wheeler, An Elliptic Collocation-Finite Element Method with Interior Penalties, SIAM Journal on Numerical Analysis 15 (1) (1978) 152–161.

[13] G. J. Gassner and A. D. Beck, On the accuracy of high-order discretizations for underresolved turbulence simulations, A.D. Theor. Comput. Fluid Dyn. (2013) 27–221.

[14] M. Kompenhans, G. Rubio, E. Ferrer, and E. Valero, Adaptation strategies for high order discontinuous galerkin methods based on tau-estimation, Journal of Computational Physics 306 (2016) 216 – 236.

[15] M. Kompenhans, G. Rubio, E. Ferrer, and E. Valero, Comparisons of p–adaptation strategies based on truncation– and discretisation–errors for high order discontinuous Galerkin methods, Computers & Fluids 139 (2016) 36 – 46, 13th {USNCCM} International Symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 60th birthday of Professor David Kopriva.

[16] B. Riviere, Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.

[17] B. Cockburn, G. Kanschat, and D. Schötzau, An equal-order DG method for the incompressible Navier–Stokes equations, Journal of Scientific Computing 40 (2009) 188–210.

[18] E. Ferrer and R.H.J. Willden, A high order discontinuous Galerkin finite element solver for the incompressible Navier–Stokes equations, Computers & Fluids 46 (1) (2011) 224–230.

[19] E. Ferrer and R. H.J. Willden, A high order discontinuous Galerkin - Fourier incompressible 3D Navier-Stokes solver with rotating sliding meshes, Journal of Computational Physics 231 (21) (2012) 7037–7056.

[20] E. Ferrer, D. Moxey, R. Willden, S. Sherwin, Stability of projection methods for incompressible flows using high order pressure-velocity pairs of same degree: Continuous and discontinuous galerkin formulations, Communications in Computational Physics 16 (3) (2014) 817–840.

[21] E. Ferrer, An interior penalty stabilised incompressible Discontinuous Galerkin - Fourier solver for implicit Large Eddy Simulations, Journal of Computational Physics 348 (2017) 754–775.

[22] R. Hartmann, P. Houston, Symmetric Interior Penalty DG Methods for the Compressible Navier-Stokes Equations I: Method Formulation, Nottingham e-prints.

[23] K. Black, A conservative spectral element method for the approximation of compressible fluid flow, Kybernetika 35 (1) (1999) 133–146.

[24] D. A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, Journal of Scientific Computing 26 (3) (2006) 301.

[25] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods: Fundamentals in Single Domains (Scientific Computation), 1st Edition, Scientific computation, Springer, 2011.

[26] D.A. Kopriva, Implementing spectral methods for partial differential equations, Springer Netherlands, 2009.

[27] D. A. Kopriva, S. L. Woodruff, M. Y. Hussaini, Computation of electromagnetic scattering with a non-conforming discontinuous spectral element method, International Journal for Numerical Methods in Engineering 53 (1) (2002) 105–122.

[28] J. Peraire, P.-O. Persson, The Compact Discontinuous Galerkin (CDG) Method for Elliptic Problems, SIAM Journal on Scientific Computing 30 (4) (2008) 1806–1824.

[29] J. Hicken, D. Zingg, Summation-by-parts operators and high-order quadrature, Journal of Computational and Applied Mathematics 237 (1) (2013) 111 – 125.

[30] D. A. Kopriva, A Polynomial Spectral Calculus for Analysis of DG Spectral Element Methods, Springer International Publishing, Cham, 2017, pp. 21–40.

[31] B. Rivière, Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations, Society for Industrial and Applied Mathematics, 2008.

[32] Y. Epshteyn, B. Rivière, Estimation of penalty parameters for symmetric interior penalty Galerkin methods, Journal of Computational and Applied Mathematics 206 (2) (2007) 843 – 872.