Department of Applied Mathematics
Faculty of EEMCS

University of Twente
The Netherlands

P.O. Box 217
7500 AE Enschede
The Netherlands
Phone: +31-53-4893400
Fax: +31-53-4893114
Email: memo@math.utwente.nl
www.math.utwente.nl/publications

Memorandum No. 1690

A study on discontinuous Galerkin finite element methods for elliptic problems

J.J. Sudirham, J.J.W. van der Vegt
and R.M.J. van Damme

September, 2003

ISSN 0169-2690

A Study on Discontinuous Galerkin Finite Element Methods for Elliptic Problems

J.J. Sudirham, J.J.W. van der Vegt, and R.M.J. van Damme
University of Twente, Department of Applied Mathematics,
The Netherlands

September 15, 2003

Abstract

In this report we study several approaches of the discontinuous Galerkin finite element methods for elliptic problems. An important aspect in these formulations is the use of a lifting operator, for which we present an efficient numerical approximation technique. Numerical experiments for two different discontinuous Galerkin methods are presented for one dimensional problems and compared with exact results. In addition, the theoretical order of accuracy is verified numerically.

Keywords: discontinuous Galerkin finite element methods, elliptic problems.
Mathematics Subject Classification: 65N30, 76M10, 35J05

1 Introduction

The Discontinuous Galerkin Finite Element Method (DGFEM) is rather widely used in recent years for the numerical solution of partial differential equations. This is stimulated by the computational convenience of the method due to its high degree of locality, which is beneficial for *hp*-adaptation, and provides a good efficiency on parallel computers. An overview about the DG method is discussed in [10].

There are extensive developments of the DG method with discontinuous discretizations for first, second, and higher-order partial differential equations. In particular, the DG method for second-order elliptic problems has been studied in Bassi and Rebay [5], Brezzi et al. [7], and Cockburn et al. [11]. In [1], Arnold et al. started to collect and analyze all approaches that have been developed. In their second paper [2], they gave a unified analysis and comparison for most of the methods that have been developed over the past thirty years.

In [2], the authors give the weak formulation of the DG method for elliptic problems formulated for homogenous boundary conditions. We choose two approaches described in [2] and derive the bilinear form of these approaches for general boundary conditions. We perform numerical experiments with these approaches and use an elliptic problem with homogenous boundary conditions as a standard test case. We choose method developed in [5] as an example for method with stabilization term and an approach developed by

Baumann and Oden in [13] for a method without stabilization term. We compare the two methods based on the numerical experiments. Our conclusions are used as background information for the development of space-time DG method for time-dependent second-order parabolic partial differential equations, see [15].

The organizaation of the report is as follows. In Section 2, we present the general formulation of DG methods and derive the bilinear form of Baumann-Oden method and Bassi-Rebay method. In Section 3, we discuss in detail efficient numerical approximation technique for local lifting operator. In Section 4, we perform numerical experiments for one dimensional elliptic problems using both methods. We compare the results based on these experiments. We also study the order of accuracy of the Bass-Rebay method numerically. Finally, we end in Section 5 with some conclusions.

2 DG Methods for Elliptic Problems

In this section we cite the main results from Arnold et al. [2]. We define our problem in d dimensions. Let $\Omega \subset \mathbb{R}^d$ be a convex polygonal domain, with boundary $\partial\Omega$ partitioned as $\partial\Omega = \overline{\Gamma_D} \cup \overline{\Gamma_N}$, with $\Gamma_D \cap \Gamma_N = \emptyset$. We consider the following boundary value problem

$$
\begin{aligned}
-\Delta u &= f \quad && \text{in} \quad \Omega, && (2.1)\\
u &= g_D \quad && \text{on} \quad \Gamma_D, && (2.2)\\
\nabla u \cdot n &= g_N \quad && \text{on} \quad \Gamma_N, && (2.3)
\end{aligned}
$$

where f, g_D and g_N are given functions in $L^2(\Omega)$, and n the unit outward normal vector at $\partial\Omega$. Introducing the auxiliary variable $\sigma = \nabla u$, the problem is rewritten as a first-order system

$$
\begin{aligned}
\sigma &= \nabla u \quad && \text{in} \quad \Omega, && (2.4)\\
-\nabla \cdot \sigma &= f \quad && \text{in} \quad \Omega, && (2.5)\\
u &= g_D \quad && \text{on} \quad \Gamma_D, && (2.6)\\
\sigma \cdot n &= g_N \quad && \text{on} \quad \Gamma_N. && (2.7)
\end{aligned}
$$

Next we want to derive a weak formulation for this elliptic partial differential equation using the DG method. Before we do that, first we introduce the finite element spaces for this problem and some trace operators.

2.1 Finite Element Spaces and Trace Operators

In this section we introduce the definition of the finite element spaces for our formulation and define the necessary trace operators related to the discontinuity of the functions accross element faces. An approximation to Ω is defined as $\Omega_h = \{K\}$ with K a finite element, which is a subset of Ω. The tessellation $\mathcal{T}_h = \{K\}$ of Ω_h is defined as

$$\mathcal{T}_h := \{K_j \mid \bigcup_{j=1}^{N} K_j = \Omega_h \text{ and } K_j \cap K_{j'} = \emptyset \text{ if } j \neq j', \quad 1 \leq j, j' \leq N\},$$

such that $\Omega_h \to \Omega$ as $h \to 0$, with h the radius of the smallest sphere completely containing each element $K \in \mathcal{T}_h$, and N the total number of elements in Ω_h. Each element $K \in \mathcal{T}_h$ is an image of a fixed master element $\hat{K}$; i.e., $K = F_K(\hat{K})$ for all $K \in \mathcal{T}_h$, where $\hat{K}$ is either the open unit simplex or the open unit hypercube in $\mathbb{R}^d$. For a nonnegative integer k, we denote by $P_k(\hat{K})$ the set of polynomials of total degree k on $\hat{K}$. When $\hat{K}$ is the unit hypercube, we also consider $Q_k(\hat{K})$, the set of all tensor product polynomials on $\hat{K}$ of degree k in each coordinate direction. To each $K \in \mathcal{T}_h$ we assign a nonnegative integer p_k as local polynomial degree. The finite element spaces are defined as

$$V_h := \{v_h \in L^2(\Omega) : v \mid_K \circ F_K \in R_{p_k}(\hat{K}), \quad \forall K \in \mathcal{T}_h\}$$
$$\Sigma_h := \{\tau_h \in (L^2(\Omega))^d : \tau \mid_K \circ F_K \in (R_{p_k}(\hat{K}))^d, \quad \forall K \in \mathcal{T}_h\}$$

where R is either P or Q and we require that $\nabla(V_h) \subset \Sigma_h$. Each function $v(x) \in V_h$ in element K_j is defined as

$$v(x) = \sum_{m=0}^{p_k} \hat{V}_{m,j} \phi_{m,j}(x), \tag{2.8}$$

with p_k the polynomial degree in element K_j, $\phi_{m,j}(x) = \hat{\phi}_m(F_K^{-1}(x))$ the basis functions on element K_j, and $\hat{V}_{m,j}$ the expansion coefficients.

We introduce now an appropriate functional setting. We denote by $H^l(\mathcal{T}_h)$ the space of functions on Ω whose restriction to each element K belongs to the Sobolev space $H^l(K)$. The finite element spaces V_h and Σ_h are subsets of $H^l(\mathcal{T}_h)$ and $(H^l(\mathcal{T}_h))^d$, respectively, for any l. The traces of v and q at the element boundary ∂K are defined as

$$v_K = \lim_{\epsilon \downarrow 0} v(x - \epsilon n_K), \quad \forall v \in V_h, \tag{2.9}$$

$$q_K = \lim_{\epsilon \downarrow 0} q(x - \epsilon n_K), \quad \forall q \in \Sigma_h, \tag{2.10}$$

which means that v_K and q_K are restricted to element K, with n_K the unit outward normal vector at ∂K. The traces v_K and q_K belong to classes $T(\Gamma) := \prod_{K \in \mathcal{T}_h} L^2(\partial K)$ and $(T(\Gamma))^d$, where Γ denotes the union of the boundaries of the elements K of $\mathcal{T}_h$. The interior faces Γ_{int} are defined as $\Gamma_{\text{int}} := \Gamma \setminus \partial\Omega_h$. Next we define the *average* and *jump* operators. We define an internal face $e_{\text{int}} \in \Gamma_{\text{int}}$ shared by elements K_1 and K_2, and a boundary face $e_{\text{bnd}} \in (\partial K_1 \cap \partial\Omega_h)$. The functions $v \in T(\Gamma_{\text{int}})$ and $q \in (T(\Gamma_{int}))^d$ are multivalued on an internal face $e_{\text{int}} \in \Gamma_{\text{int}}$. The unit normal vectors n_{K_1} and n_{K_2} are defined on e_{int} and e_{bnd} pointing exterior to K_1 and K_2, respectively. Defining functions $v_i := v_{K_i}$, $q_i := q_{K_i}, n_i := n_{K_i}$, the average operator is defined as

$$\{v\} = \frac{1}{2}(v_1 + v_2), \quad \text{on} \quad e_{\text{int}}, \tag{2.11}$$
$$\{q\} = \frac{1}{2}(q_1 + q_2), \quad \text{on} \quad e_{\text{int}}, \tag{2.12}$$
$$\{v\} = v_1, \quad \text{on} \quad e_{\text{bnd}}, \tag{2.13}$$
$$\{q\} = q_1, \quad \text{on} \quad e_{\text{bnd}}, \tag{2.14}$$

and the jump operator is defined as

$$[v] = v_1 n_1 + v_2 n_2, \quad \text{on} \quad e_{\text{int}}, \tag{2.15}$$
$$[q] = q_1 \cdot n_1 + q_2 \cdot n_2, \quad \text{on} \quad e_{\text{int}}, \tag{2.16}$$
$$[v] = v_1 n_1, \quad \text{on} \quad e_{\text{bnd}}, \tag{2.17}$$
$$[q] = q_1 \cdot n_1, \quad \text{on} \quad e_{\text{bnd}}. \tag{2.18}$$

Notice that the jump $[v]$ of a scalar function v is a vector parallel to the normal, and the jump $[q]$ of a vector function q is a scalar quantity. In the next section we show main steps to obtain a weak formulation for DG methods for elliptic problems.

2.2 Weak Formulation for DG Methods

In this section we derive the weak formulation for (2.4) - (2.7) using a DG method. We start by multiplying (2.4) and (2.5) by test functions $\tau \in \Sigma_h$ and $v \in V_h$, respectively, and integrate by parts formally on an element K to obtain

$$\int_K \sigma \cdot \tau dx = -\int_K u \nabla \cdot \tau dx + \int_{\partial K} u n_K \cdot \tau ds, \quad \tau \in \Sigma_h, \tag{2.19}$$
$$\int_K \sigma \cdot \nabla v dx = \int_K f v dx + \int_{\partial K} \sigma \cdot n_K v ds, \quad v \in V_h. \tag{2.20}$$

The DG finite element discretization is obtained by approximating the functions u and σ in each element $K \in \mathcal{T}_h$ with $u_h \in V_h$ and $\sigma_h \in \Sigma_h$. Because of these choices, the functions u and σ in the element boundary integrals are replaced with linear *numerical fluxes* $\hat{u}_K = (\hat{u}_K)_{K \in \mathcal{T}_h}$ and $\hat{\sigma}_h = (\hat{\sigma}_K)_{K \in \mathcal{T}_h}$, which are the approximations at the boundary of K to u and σ, respectively. Choosing appropriate numerical fluxes is the main topic in many articles discussing the DG method, see for instance [2]. The general weak formulation can now be expressed as
Find a $u_h \in V_h$ and $\sigma_h \in \Sigma_h$ such that for all $K \in \mathcal{T}_h$ we have

$$\int_K \sigma_h \cdot \tau dx = -\int_K u_h \nabla \cdot \tau dx + \int_{\partial K} \hat{u}_K n_K \cdot \tau ds, \quad \forall \tau \in \Sigma_h, \tag{2.21}$$
$$\int_K \sigma_h \cdot \nabla v dx = \int_K f v dx + \int_{\partial K} \hat{\sigma}_K \cdot n_K v ds, \quad \forall v \in V_h. \tag{2.22}$$

If we sum (2.21) and (2.22) over all elements, we obtain

$$\int_\Omega \sigma_h \cdot \tau dx = -\int_\Omega u_h \nabla \cdot \tau dx + \sum_{K \in \mathcal{T}_h} \int_{\partial K} \hat{u}_K n_K \cdot \tau ds, \quad \forall \tau \in \Sigma_h, \tag{2.23}$$

$$\int_\Omega \sigma_h \cdot \nabla v dx = \int_\Omega f v dx + \sum_{K \in \mathcal{T}_h} \int_{\partial K} \hat{\sigma}_K \cdot n_K v ds, \quad \forall v \in V_h. \tag{2.24}$$

Following the derivation in Arnold et. al [2], for all $v \in T(\Gamma)$ and for all $q \in (T(\Gamma))^d$ we have the relation

$$\sum_{K \in \mathcal{T}_h} \int_{\partial K} v_K q_K \cdot n_K v ds = \int_\Gamma [v] \cdot \{q\} ds + \int_{\Gamma_{\text{int}}} \{v\}[q] ds. \tag{2.25}$$

Using this identity, we obtain

$$\int_\Omega \sigma_h \cdot \tau dx = -\int_\Omega u_h \nabla \cdot \tau dx + \int_\Gamma [\hat{u}] \cdot \{\tau\} ds + \int_{\Gamma_{\text{int}}} \{\hat{u}\}[\tau] ds, \quad \forall \tau \in \Sigma_h, \tag{2.26}$$

$$\int_\Omega \sigma_h \cdot \nabla v dx = \int_\Omega f v dx + \int_\Gamma \{\hat{\sigma}\} \cdot [v] ds + \int_{\Gamma_{\text{int}}} [\hat{\sigma}]\{v\} ds, \quad \forall v \in V_h. \tag{2.27}$$

Using integration by parts formula and (2.25), the equation for σ_h (2.26) can be transformed into

$$\int_\Omega \sigma_h \cdot \tau dx = \int_\Omega \nabla u_h \cdot \tau dx - \int_\Gamma [u_h - \hat{u}] \cdot \{\tau\} ds - \int_{\Gamma_{\text{int}}} \{u_h - \hat{u}\}[\tau] ds. \tag{2.28}$$

Define the lifting operators $r : (L^2(\Gamma))^d \to \Sigma_h$ and $l : L^2(\Gamma_{\text{int}}) \to \Sigma_h$ by

$$\int_\Omega r(q) \cdot \tau dx = -\int_\Gamma q \cdot \{\tau\} ds, \tag{2.29}$$

$$\int_\Omega l(v) \cdot \tau dx = -\int_{\Gamma_{\text{int}}} v[\tau] ds, \tag{2.30}$$

for all $\tau \in \Sigma_h$. Using the lifting operators, (2.28) can be written as

$$\int_\Omega \sigma_h \cdot \tau dx = \int_\Omega \nabla u_h \cdot \tau dx + \int_\Omega r([u_h - \hat{u}]) \cdot \tau dx + \int_\Omega l(\{u_h - \hat{u}\}) \cdot \tau dx. \tag{2.31}$$

From the last equation, we obtain

$$\sigma_h = \nabla u_h + r([u_h - \hat{u}]) + l(\{u_h - \hat{u}\}) \qquad \text{a.e.} \tag{2.32}$$

Inserting the last equation into (2.27), we obtain

Table 1: Some DG methods and their numerical fluxes.

	Method	$\hat{u}_K$	$\hat{\sigma}_K$	Reference
1.	Bassi-Rebay	$\{u_h\}$	$\{\sigma_h\}$	[4]
2.	Brezzi et al. 1	$\{u_h\}$	$\{\sigma_h\} - \alpha_r([u_h])$	[7]
3.	LDG	$\{u_h\} - \beta \cdot [u_h]$	$\{\sigma_h\} + \beta[\sigma_h] - \alpha_j([u_h])$	[11]
4.	IP	$\{u_h\}$	$\{\nabla u_h\} - \alpha_j([u_h])$	[12]
5.	Bassi et al. 2	$\{u_h\}$	$\{\nabla u_h\} - \alpha_r([u_h])$	[6]
6.	Baumann-Oden	$\{u_h\} + n_K \cdot [u_h]$	$\{\nabla u_h\}$	[13]
7.	NIPG	$\{u_h\} + n_K \cdot [u_h]$	$\{\nabla u_h\} - \alpha_j([u_h])$	[14]
8.	Babuska-Zlamal	$(u_h\|_K)\|_{\partial K}$	$-\alpha_j([u_h])$	[3]
9.	Brezzi et al. 2	$(u_h\|_K)\|_{\partial K}$	$-\alpha_r([u_h])$	[8]

$$\int_\Omega (\nabla u_h + r([u_h - \hat{u}]) + l(\{u_h - \hat{u}\})) \cdot \nabla v dx = \int_\Omega f v dx + \int_\Gamma \{\hat{\sigma}\} \cdot [v] ds + \int_{\Gamma_{\text{int}}} [\hat{\sigma}]\{v\} ds, \quad \forall v \in V_h. \tag{2.33}$$

The weak formulation for DG finite element discretizations for elliptic problems finally can be written as follows

$$B(u_h, v) = \int_\Omega f v dx, \quad \forall v \in V_h, \tag{2.34}$$

where

$$B(u_h, v) := \int_\Omega \nabla u_h \cdot \nabla v dx - \int_\Gamma ([u_h - \hat{u}] \cdot \{\nabla v\} + \{\hat{\sigma}\} \cdot [v]) ds - \int_{\Gamma_{\text{int}}} (\{u_h - \hat{u}\}[\nabla v] + [\hat{\sigma}]\{v\}) ds. \tag{2.35}$$

In [2] Arnold et al. have listed all the choices for the numerical fluxes used in (2.21) and (2.22) that have been proposed so far. The choices of $\hat{u}_K$ and $\hat{\sigma}_K$ for different approaches are summarized in Table 1. Note that in this table the last column contains the reference of each method. The choices of the numerical fluxes holds for interior elements or homogenous boundary conditions. For general boundary conditions, these choices can be different. These choices can be found in [13] for Baumann-Oden method, in [7] for Methods 1, 2, 5, and 9, while for LDG method, the numerical fluxes for general boundary conditions can be found in [9].

In Table 1, some numerical fluxes for $\hat{\sigma}_K$ contain the operators $\alpha_j([u_h])$ and $\alpha_r([u_h])$. Here we explain briefly the formulation for these operators, which are called local lifting operators.

- The operator $\alpha_j(\phi)$ is simply $\mu\phi$ with $\mu \in \mathbb{R}^+$. This operator comes from the interior penalty (IP) term

$$\alpha^j(w, v) = \int_\Gamma \mu[w] \cdot [v] ds \tag{2.36}$$

with the penalty weighting function $\mu : \Gamma \to \mathbb{R}^+$ given by $\eta_e h_e^{-1}\phi$ on e, with η_e a positive number.

- The operator $\alpha_r(\phi)$ is defined as $\alpha_r(\phi) = -\eta_e\{r_e(\phi)\}$ on a face $e \in \Gamma_{\text{int}}$ and as $\alpha_r(\phi) = -\eta_e r_{e,g_D}(\phi)$ on a face $e \in \Gamma_D$. For all $\tau \in \Sigma_h$, the local lifting operators $r_e : (L^1(e))^d \to \Sigma_h$ and $r_{e,g_D} : (L^1(e))^d \to \Sigma_h$ are given by

$$\int_\Omega r_e(\phi) \cdot \tau dx = -\int_e \phi \cdot \{\tau\} ds, \qquad \text{on } e \in \Gamma_{\text{int}}, \tag{2.37}$$

$$\int_\Omega r_{e,g_D}(\phi) \cdot \tau dx = -\int_e \phi \cdot \tau ds + \int_e g_D n \cdot \tau ds, \quad \text{on } e \in \Gamma_D. \tag{2.38}$$

Note that $r_e(\phi)$ vanishes outside the union of one or two elements containing the face e and that $r(\phi) = \sum_{e\in\partial K} r_e(\phi)$ for any $K \in \mathcal{T}_h$. In Section 3 we will explain one formulation to compute the lifting operator of this type.

In [2], it was concluded that Methods 2 to 5 in Table 1 are consistent, adjoint consistent and stable under certain condition on parameters μ and η. These methods have a local lifting operator in their formulation, either in the form of α_j or α_r. This fact indicates that the lifting operator gives an important contribution to the properties of the DG method. Most DG methods with the local lifting operators have optimal rates of convergence of h^k in $H^1(\mathcal{T}_h)$ and h^{k+1} in L^2, see [2]. It was also concluded that DG methods whose numerical fluxes $\hat{\sigma}_K$ are independent of σ_h (Methods 4 to 9 in Table 1) produce stiffness matrices with a smaller number of non-zero entries. This makes the matrices are more sparse than the others. In the next section we will choose some methods from Table 1 and discuss the weak formulation of each of these methods.

2.3 Weak Formulation for Several Approaches

In this section we derive the weak formulations for different DG finite element discretizations for elliptic problems in more detail.

- Baumann-Oden method (Method 6 in Table 1)

  This method uses

$$\hat{u}_K = \begin{cases} \{u_h\} + n_K \cdot [u_h], & \text{on } \Gamma_{\text{int}}, \\ n_K \cdot [u_h - g_D], & \text{on } \Gamma_D, \\ u_h, & \text{on } \Gamma_N, \end{cases}$$

and

$$\hat{\sigma}_K \cdot n_K = \begin{cases} \{\nabla u_h\} \cdot n, & \text{on } \Gamma_{\text{int}}, \\ \nabla u_h \cdot n, & \text{on } \Gamma_D, \\ g_N, & \text{on } \Gamma_N, \end{cases}$$

as their numerical fluxes. Substituting these fluxes into (2.35), we obtain

$$B(u_h, v) := \int_\Omega \nabla u_h \cdot \nabla v dx + \int_{\Gamma_{\text{int}} \cup \Gamma_D} \big([u_h] \cdot \{\nabla v\} - \{\nabla u_h\} \cdot [v]\big) ds \\ - \int_{\Gamma_N} g_N v ds - \int_{\Gamma_D} g_D n \cdot \nabla v ds. \quad (2.39)$$

- Bassi et al. method (Method 5 in Table 1)

  This method uses

$$\hat{u}_K = \begin{cases} \{u_h\}, & \text{on } \Gamma_{\text{int}}, \\ g_D, & \text{on } \Gamma_D, \\ u_h, & \text{on } \Gamma_N, \end{cases}$$

and

$$\hat{\sigma}_K \cdot n_K = \begin{cases} (\{\nabla u_h\} + \eta_e \{r_e([u_h])\}) \cdot n, & \text{on } \Gamma_{\text{int}}, \\ (\nabla u_h + \eta_e \{r_{e,g_D}([u_h])\}) \cdot n, & \text{on } \Gamma_D, \\ g_N, & \text{on } \Gamma_N, \end{cases}$$

as their numerical fluxes. Substituting these fluxes into (2.35), we obtain

$$B(u_h, v) := \int_\Omega \nabla u_h \cdot \nabla v dx - \int_{\Gamma_{\text{int}} \cup \Gamma_D} ([u_h] \cdot \{\nabla v\} + \{\nabla u_h\} \cdot [v]) ds \\ - \sum_{e \in \Gamma_{\text{int}}} \eta_e \int_e \{\hat{r}_e([u_h])\} \cdot [v] ds - \sum_{e \in \Gamma_D} \eta_e \int_e \hat{r}_{e,g_D}([u_h]) \cdot v n ds \\ + \int_{\Gamma_D} g_D n \cdot \nabla v ds - \int_{\Gamma_N} g_N v ds. \quad (2.40)$$

In order to have a stable method, [7] and [2] suggested to take the parameter $\eta_e > \mathcal{F}$ with $\mathcal{F}$ the number of element faces.

3 Local Lifting Operator

In this section we derive a way to compute the local lifting operator. There is considerable freedom in computing the local lifting operator, the paper from Bassi and Rebay [5] gives one example. Since we use a local lifting operator of α_r type for the Bassi-Rebay method, we explain in this report how to formulate this operator in terms of the expansion coefficients in (2.8).

The local lifting operators $r_e : (L^1(e))^d \to \Sigma_h$ and $r_{e,g_D} : (L^1(e))^d \to \Sigma_h$ for $[u_h]$ can be written as

$$\int_\Omega r_e([u_h]) \cdot \tau dx = -\int_e [u_h] \cdot \{\tau\} ds, \qquad \forall \tau \in \Sigma_h, \quad \text{for} \quad e \in \Gamma_{\text{int}}, \tag{3.1}$$

$$\int_\Omega r_{e,g_D}([u_h]) \cdot \tau dx = -\int_e u_h n \cdot \tau ds + \int_e g_D n \cdot \tau ds, \quad \forall \tau \in \Sigma_h, \quad \text{for} \quad e \in \Gamma_D. \tag{3.2}$$

One possibility for the operator r_e is to express it as polynomial expansion as in (2.8)

$$r_e([u_h]) = \sum_{p=0}^{p_k} \hat{R}_{p,j} \phi_{p,j}(x), \tag{3.3}$$

with coefficients $\hat{R}_{p,j} \in \mathbb{R}^d$. Techniques for computing the local lifting operators in (3.1) and (3.2) will be explained separately in the next sections.

3.1 Lifting operator on an internal face

In this section we consider the local lifting operator defined in (3.1) on an internal face $e \in \Gamma_{\text{int}}$, where two elements K_i and K_j share this face. Using (3.3) in the weak formulation for the lifting operator (3.1), we obtain

$$\int_{K_i} r_{e,i}([u_h]) \cdot \tau_i dx + \int_{K_j} r_{e,j}([u_h]) \cdot \tau_j dx = -\frac{1}{2} \int_e (u_{h,i} n_i + u_{h,j} n_j) \cdot (\tau_i + \tau_j) ds, \quad \forall \tau_i, \tau_j \in \Sigma_h, \tag{3.4}$$

as the operator $r_e(u_h)$ vanishes outside the union of the two elements containing the face e, and hence the left-hand side in the equation only contains the contribution from elements K_i and K_j. Here we expand the jump operator of u_h (2.15) and the average operator for τ (2.12). Comparing the left-hand and right-hand sides in (3.4), the operator $r_{e,i}$ in an element K_i can be written as

$$\int_{K_i} r_{e,i}([u_h]) \cdot \tau_i dx = -\frac{1}{2} \int_e u_{h,i} n_i \cdot \tau_i - \frac{1}{2} \int_e u_{h,j} n_j \cdot \tau_i ds, \quad \forall \tau_i \in \Sigma_h. \tag{3.5}$$

Substituting the expansions (3.3) into (3.5), and using the argument that (3.5) holds for any $\tau_i \in \Sigma_h$, we obtain

$$\sum_{n=0}^{p_i} \hat{R}_{n,i} \int_{K_i} \phi_{l,i}(x)\phi_{n,i}(x)dx = -\frac{1}{2}\sum_{m=0}^{p_i} \hat{U}_{m,i} \int_e \phi_{l,i}(x_i)\phi_{m,i}(x_i)n_i ds$$
$$-\frac{1}{2}\sum_{p=0}^{p_j} \hat{U}_{p,j} \int_e \phi_{l,i}(x_i)\phi_{p,j}(x_j)n_j ds, \quad l = 0,\dots,p_i, \quad (3.6)$$

with p_i, p_j denote the local polynomial degree of element K_i and K_j, respectively. If we define the matrices $A_i \in \mathbb{R}^{(p_i+1)\times(p_i+1)}$ as

$$A_i = \int_{K_i} \phi_{l,i}(x)\phi_{n,i}(x)dx,$$

and the right-hand side in (3.6) as $P(\hat{U}_i, \hat{U}_j) \in \mathbb{R}^{(p_i+1)\times d}$, the linear system for coefficients $\hat{R}_i \in \mathbb{R}^{(p_i+1)\times d}$ is obtained

$$A_i \hat{R}_i = P(\hat{U}_i, \hat{U}_j). \quad (3.7)$$

We can solve this linear system using a linear solver or express the coefficients $\hat{R}_i$ directly in terms of $\hat{U}_i$ and $\hat{U}_j$. The lifting operator $r_{e,i}$ in element K_i which shares a face e with element K_j can now be represented as

$$r_{e,i}([u_h]) = \sum_{n=0}^{p_i} A_i^{-1} P(\hat{U}_i, \hat{U}_j)\phi_{n,i}(x). \quad (3.8)$$

3.2 Lifting operator on a Dirichlet boundary face

In this section we shows how to compute the lifting operator on a Dirichlet boundary face $e \in \Gamma_D$ in terms of approximate functions u_h and the boundary condition g_D. The lifting operator $r_{e,g_D} : (L^1(e))^d \to \Sigma_h$ in a boundary element K_j is given by

$$\int_{K_j} r_{e,g_D,j}([u_h]) \cdot \tau_j dx = -\int_e u_{h,j} n_j \cdot \tau_j ds + \int_e g_D n_j \cdot \tau_j dS, \quad \forall \tau_j \in \Sigma_h. \quad (3.9)$$

Analogous to the previous section we substitute the expansions (3.3) into (3.9) and using the argument that (3.9) holds for any $\tau_j \in \Sigma_h$, we obtain

$$\sum_{n=0}^{p_j} \hat{R}_{n,j} \int_{K_j} \phi_{l,j}(x)\phi_{n,j}(x)dx = -\sum_{m=0}^{p_j} \hat{U}_{m,j} \int_e \phi_{l,j}(x_j)\phi_{m,j}(x_j)n_j ds$$
$$+\int_e \phi_{l,j}(x_j) g_D n_j ds, \quad l = 0,\dots,p_j. \quad (3.10)$$

Defining the right-hand side of (3.10) as $P(\hat{U}_j, g_D) \in \mathbb{R}^{(p_j+1)\times d}$, the linear system for the coefficients $\hat{R}_j \in \mathbb{R}^{(p_j+1)\times d}$ is obtained

$$A_j \hat{R}_j = P(\hat{U}_j, g_D). \quad (3.11)$$

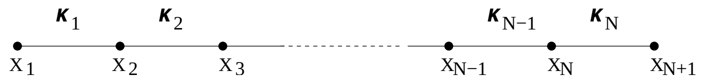


Figure 1: 1D space elements

The local lifting operator $r_{e,g_D,j}$ in element K_j can be expressed as

$$r_{e,g_D,j}([u_h]) = \sum_{n=0}^{p_j} A_j^{-1} P(\hat{U}_j, g_D)\phi_{n,j}(x). \tag{3.12}$$

This completes the description of the formulation of local lifting operator. In the next section we will discuss numerical experiments of DG methods for elliptic problems in one dimension.

4 Numerical Experiments

In this section we present the numerical discretization and solutions obtained with the two methods discussed in Section 2.3 for a one dimensional problem with homogenous boundary condition

$$\begin{aligned} -u_{xx} &= f(x), \quad 0 \le x \le 1, \\ u(0) &= 0, \\ u(1) &= 0. \end{aligned}$$

The interval $(0,1)$ is partitioned into N elements K_j, $j = 1, \ldots, N$ (Figure 1). The end points of element K_j are denoted by x_j and x_{j+1}. Each element K_j has two boundaries which are the end points of the element and we denote these boundaries S_j and S_{j+1}. Each element K is related to the master element $\hat{K} = (-1,1)$ through the parametrization (see [16])

$$x = F_{K_j}(\xi) = \frac{1}{2}(1-\xi)x_j + \frac{1}{2}(1+\xi)x_{j+1}.$$

The basis functions $\phi_{m,j}$ on element K_j and the basis functions $\hat{\phi}_m$ on the master element $\hat{K}$ have the following relation

$$\hat{\phi}_m(\xi_1) = \hat{\phi}_m(F_{K_j}^{-1}(\xi_1)) = \phi_{m,j}(x)$$

with $\xi \in \hat{K}$ and $\hat{\phi}_m = \xi^m$. In the next sections we will discuss the numerical discretization for the 1D elliptic problem in detail.

4.1 Baumann-Oden Method for the 1D Problem

In this section we discuss the numerical discretization of Baumann-Oden method in detail. For a one dimensional problem with homogenous boundary condition, the formulation of Baumann-Oden method (2.39) gives the discrete formulation

$$\sum_{j=1}^{N}\int_{K_j}\frac{du_h}{dx}\frac{dv}{dx}dx+\sum_{i=1}^{N+1}\Big([u_h]\{\frac{dv}{dx}\}\big|_{S_i}\Big)-\sum_{i=1}^{N+1}\Big(\{\frac{du_h}{dx}\}[v]\big|_{S_i}\Big)=\sum_{j=1}^{N}\int_{K_j}fvdx.$$

After substituting the average and jump operators for u_h, for each interior element $K_j, j = 2,\dots,N-1$ we obtain

$$\begin{aligned}&\int_{K_j}\frac{du_j(x)}{dx}\frac{dv_j(x)}{dx}dx\\&+\frac{1}{2}\Big((u_j(x_{j+1}^-)n^-+u_{j+1}(x_{j+1}^+)n^+)\frac{dv_j(x_{j+1}^-)}{dx}\Big)+\frac{1}{2}\Big((u_j(x_j^+)n^-+u_{j-1}(x_j^-)n^+)\frac{dv_j(x_j^+)}{dx}\Big)\\&-\frac{1}{2}\Big((\frac{du_j(x_{j+1}^-)}{dx}+\frac{du_{j+1}(x_{j+1}^+)}{dx})v_j(x_{j+1}^-)n^-\Big)-\frac{1}{2}\Big((\frac{du_j(x_j^+)}{dx}+\frac{du_{j-1}(x_j^-)}{dx})v_j(x_j^+)n^-\Big)\\&=\int_{K_j}fv_j(x)dx,\end{aligned}\quad(4.1)$$

where n^- denotes the unit outward normal vector at ∂K_j, n^+ the unit outward normal vector of elements connected to element K_j, $(n^+=-n^-)$, and $x_i^{\pm}$ is defined as $\lim_{\epsilon\to 0}(x_i\pm\epsilon)$. For elements at the domain boundary (K_1 and K_N), we substitute the average and jump operators defined at the boundary ((2.13), (2.14), (2.17), and (2.18)). At the boundary S_{j+1} the unit normal vectors are defined as $n^-=1, n^+=-1$ while at S_j, we have $n^-=-1, n^+=1$.

If we substitute polynomial expansions for u and v into (4.1), the numerical discretizations for the coefficients $\hat{U}_{m,j}$ is obtained

$$\mathcal{M}_1\hat{U}_{j-1}+\mathcal{M}_2\hat{U}_j+\mathcal{M}_3\hat{U}_{j+1}=F_j,\quad(4.2)$$

with

$$\begin{aligned}\mathcal{M}_1&=\frac{1}{2}C_{j,j-1}(x_j^+,x_j^-)+\frac{1}{2}B_{j,j-1}(x_j^+,x_j^-),\\\mathcal{M}_2&=D_j+\frac{1}{2}C_{j,j}(x_{j+1}^-,x_{j+1}^-)-\frac{1}{2}C_{j,j}(x_j^+,x_j^+)-\frac{1}{2}B_{j,j}(x_{j+1}^-,x_{j+1}^-)+\frac{1}{2}B_{j,j}(x_j^+,x_j^+),\\\mathcal{M}_3&=-\frac{1}{2}C_{j,j+1}(x_{j+1}^-,x_{j+1}^+)-\frac{1}{2}B_{j,j+1}(x_{j+1}^-,x_{j+1}^+).\end{aligned}$$

The matrices $D_j\in\mathbb{R}^{(p_j+1)\times(p_j+1)}, B_{i,j}\in\mathbb{R}^{(p_i+1)\times(p_j+1)}, C_{i,j}\in\mathbb{R}^{(p_i+1)\times(p_j+1)}$ and vector $F_j\in\mathbb{R}^{(p_j+1)}$ are defined as

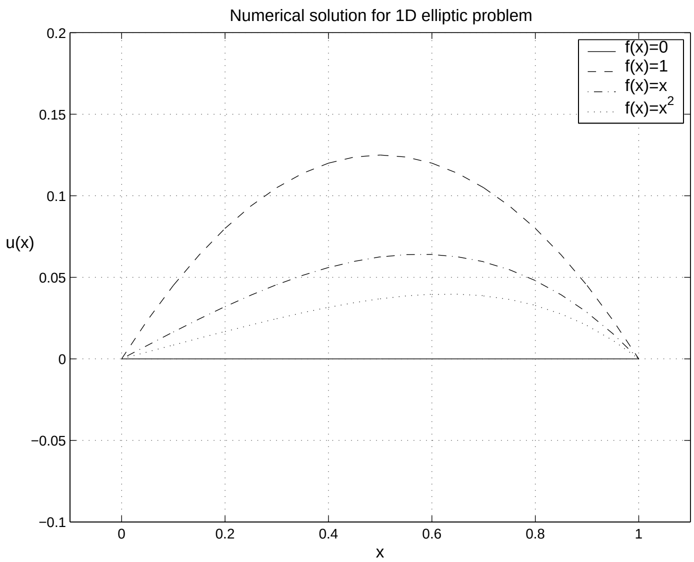


Figure 2: Results with Baumann-Oden method using quadratic basis functions

$$
\begin{aligned}
D_j &= \int_{K_j} \frac{d\phi_{n,j}(x)}{dx} \frac{d\phi_{m,j}(x)}{dx} dx, \\
B_{i,j}(x_1, x_2) &= \phi_{n,i}(x_1) \frac{d\phi_{m,j}(x_2)}{dx}, \\
C_{i,j}(x_1, x_2) &= \frac{d\phi_{n,i}(x_1)}{dx} \phi_{m,j}(x_2), \\
F_j &= \int_{K_j} \phi_{n,j}(x) f(x) dx.
\end{aligned}
$$

The matrix structure obtained with the Baumann-Oden method is a compact stencil, as it only contains the contributions from the element and its direct neighbours. We perform simulations using linear, quadratic, and cubic functions as the basis functions $\hat{\phi}_m$. We choose the function $f(x)$ to be $0, 1, x, x^2$ so that the problem has the analytical solution $u(x)$ equal to $0, -x^2/2 + x/2, -x^3/6 + x/6, -x^4/12 + x/12$, respectively. For linear basis functions, the stiffness matrix is singular as can be expected theoretically from [2], for the higher order basis functions we obtain good approximations of analytical solution. An example of the result using quadratic basis functions and ten uniform-length elements is shown in Figure 2. As the stiffness matrix is singular for linear basis functions, Baumann-Oden method is not suitable for space-time DG finite element method and will not be considered any further.

4.2 Bassi-Rebay Method for the 1D Problem

In this section we describe the numerical discretization and some results from numerical experiments using the Bassi-Rebay method. We use the same one dimensional problem as in the previous section.

The DG method proposed by Bassi and Rebay in (2.40) has formulation

$$\sum_{j=1}^{N}\int_{K_j}\frac{du_h}{dx}\frac{dv}{dx}dx-\sum_{i=1}^{N+1}\Big([u_h]\{\frac{dv^-}{dx}\}\|_{S_i}\Big)-\sum_{i=1}^{N+1}\Big(\{\frac{du_h}{dx}\}[v]\|_{S_i}\Big)$$
$$-\sum_{i=1}^{N+1}\Big(\eta_e\{r_e([u_h])\}[v]\|_{S_i}\Big)=\sum_{j=1}^{N}\int_{K_j}fvdx.$$

For each interior element $K_j, j=2,\dots,N-1$, the discretization is of the form

$$\begin{aligned}&\int_{K_j}\frac{du_j(x)}{dx}\frac{dv_j(x)}{dx}dx\\ &-\frac{1}{2}\big(u_j(x_{j+1}^-)n^-+u_{j+1}(x_{j+1}^+)n^+\big)\frac{dv_j(x_{j+1}^-)}{dx}-\frac{1}{2}\big(u_j(x_j^+)n^-+u_{j-1}(x_j^-)n^+\big)\frac{dv_j(x_j^+)}{dx}\\ &-\frac{1}{2}\Big(\frac{du_j(x_{j+1}^-)}{dx}+\frac{du_{j+1}(x_{j+1}^+)}{dx}\Big)v_j(x_{j+1}^-)n^--\frac{1}{2}\Big(\frac{du_j(x_j^+)}{dx}+\frac{du_{j-1}(x_j^-)}{dx}\Big)v_j(x_j^+)n^-\\ &\quad-\eta_e\{re([u])\}|_{S_{j+1}}v_j(x_{j+1}^-)n^--\eta_e\{re([u])\}|_{S_j}v_j(x_j^+)n^-=\int_{K_j}f(x)v_j(x)dx.\end{aligned}\quad(4.3)$$

If we subtitute the polynomial expansions into (4.3), the following equations for coefficients $\hat{U}_{m,j}$ is obtained

$$\mathcal{N}_1\hat{U}_{j-1}+\mathcal{N}_2\hat{U}_j+\mathcal{N}_3\hat{U}_{j+1}=F_j,\quad(4.4)$$

with

$$\begin{aligned}\mathcal{N}_1=&-\frac{1}{2}C_{j,j-1}(x_j^+,x_j^-)-\frac{1}{4}\eta L_{j,j-1}(x_j^+,x_j^-)A_{j-1}^{-1}L_{j-1,j-1}(x_j^-,x_j^-)\\ &+\frac{1}{2}B_{j,j-1}(x_j^+,x_j^-)-\frac{1}{4}\eta L_{j,j}(x_j^+,x_j^+)A_j^{-1}L_{j,j-1}(x_j^+,x_j^-),\\ \mathcal{N}_2=&D_j-\frac{1}{2}C_{j,j}(x_{j+1}^-,x_{j+1}^-)+\frac{1}{2}C_{j,j}(x_j^+,x_j^+)+\frac{1}{4}\eta L_{j,j}(x_{j+1}^-,x_{j+1}^-)A_j^{-1}L_{j,j}(x_{j+1}^-,x_{j+1}^-)\\ &-\frac{1}{2}B_{j,j}(x_{j+1}^-,x_{j+1}^-)+\frac{1}{2}B_{j,j}(x_j^+,x_j^+)+\frac{1}{4}\eta L_{j,j+1}(x_{j+1}^-,x_{j+1}^+)A_{j+1}^{-1}L_{j+1,j}(x_{j+1}^+,x_{j+1}^-)\\ &+\frac{1}{4}\eta L_{j,j-1}(x_j^+,x_j^-)A_{j-1}^{-1}L_{j-1,j}(x_j^-,x_j^+)+\frac{1}{4}\eta L_{j,j}^-(x_j^+,x_j^+)A_j^{-1}L_{j,j}(x_j^+,x_j^+),\\ \mathcal{N}_3=&\frac{1}{2}C_{j,j+1}(x_{j+1}^-,x_{j+1}^+)-\frac{1}{4}\eta L_{j,j}(x_{j+1}^-,x_{j+1}^-)A_j^{-1}L_{j,j+1}(x_{j+1}^-,x_{j+1}^+)\\ &-\frac{1}{2}B_{j,j+1}(x_{j+1}^-,x_{j+1}^+)-\frac{1}{4}\eta L_{j,j+1}(x_{j+1}^-,x_{j+1}^+)A_{j+1}^{-1}L_{j+1,j+1}(x_{j+1}^+,x_{j+1}^+),\end{aligned}$$

and $\eta\equiv\inf_e\eta_e$. The matrix $P(\hat{U}_i,\hat{U}_j)$ in (3.8) is defined as

$$P(\hat{U}_i,\hat{U}_j)=-\frac{1}{2}n_iL_{i,i}\hat{U}_i-\frac{1}{2}n_jL_{i,j}\hat{U}_j,$$

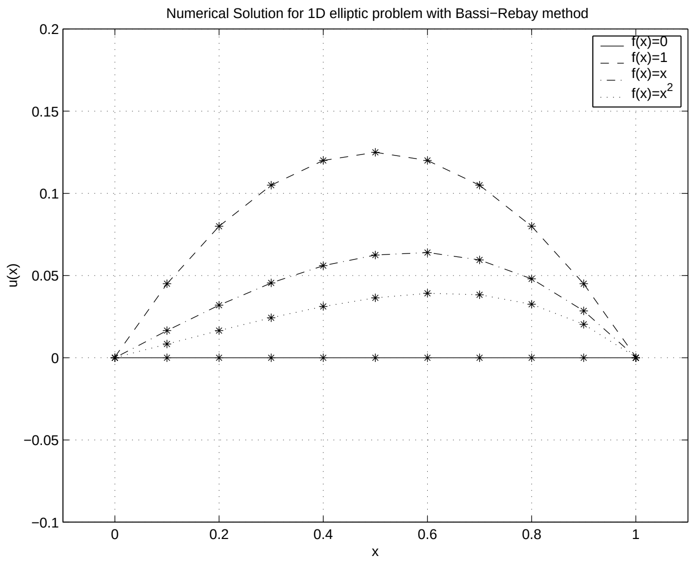


Figure 3: Results with Bassi-Rebay method with linear basis functions

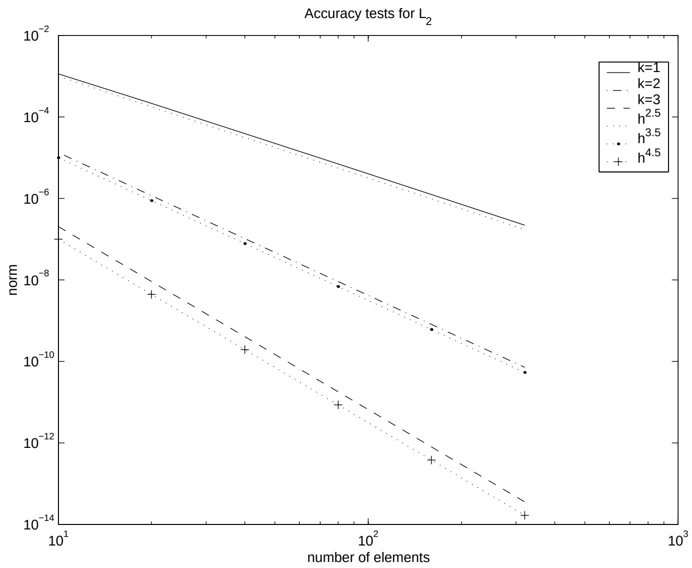
with matrix $L_{i,j} \in \mathbb{R}^{(p_i+1)\times(p_j+1)}$ defined as

$$L_{i,j}(x_1, x_2) = \phi_{n,i}(x_1)\phi_{m,j}(x_2).$$

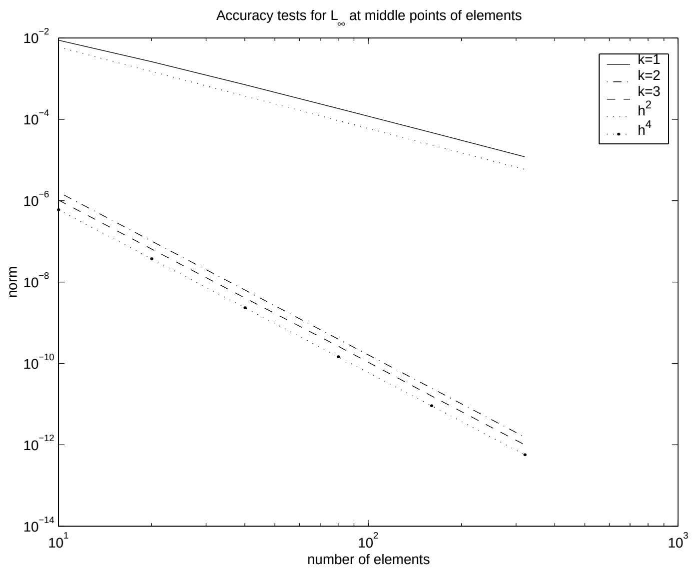
The definition of the matrices $D_j, B_{i,j}, C_{i,j}$, and vector F_j is the same as in the previous section. The stencil of the Bassi-Rebay DG discretization is also compact. For one-dimensional problems, each element is connected to two neighbours, hence a block tridiagonal matrix is obtained.

First we perform the simulation of the 1D model problem using linear basis functions. We choose the same functions $f(x)$ as in Baumann-Oden method and hence have the same analytical solution. The plot of the numerical solution using 10 uniform-length elements is presented in Figure 3.

Next we want to analyze the order of accuracy of the method. For $u(x) = -x^4/12 + x/12$ we perform simulations for linear, quadratic and cubic basis functions using an increasing number of elements. Plots of the order of accuracy in the L^2 and L^∞ norms are presented in Figure 4. We approximate the L^2 norm by computing the differences between the numerical and analytical solutions at several points on the elements, while for the L^∞norm we analyze the maximum values of all element middle points. For the L^2 norm, it is shown that the order of accuracy is higher than what we expected, that is $h^{k+1.5}$. This can be caused by the approximations we make in the compution of the L^2 norm or by the choice of the elliptic problem. The order of accuracy h^{k+1} is obtained in the L^∞norm for linear and cubic basis functions, for quadratic basis functions, the order of accuracy is h^{k+2}. The same results are obtained when we choose the solution to be $u(x) = \sin(\pi x)/\pi^2$ (Figure 5) and $u(x) = \sin(2\pi x)/\pi^2 + \sin(\pi x)/\pi^2$ (Figure 6).

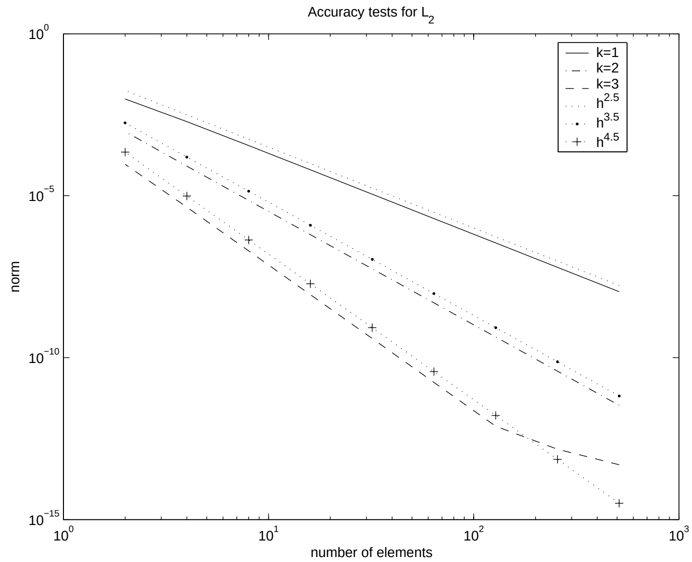


(a) L^2 norm

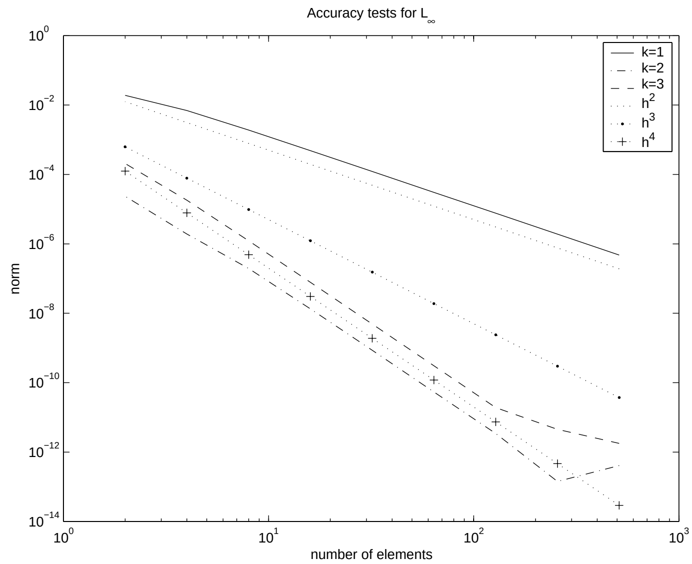


(b) L^∞ norm

Figure 4: analytical solution $u(x) = -x^4/12 + x/12$

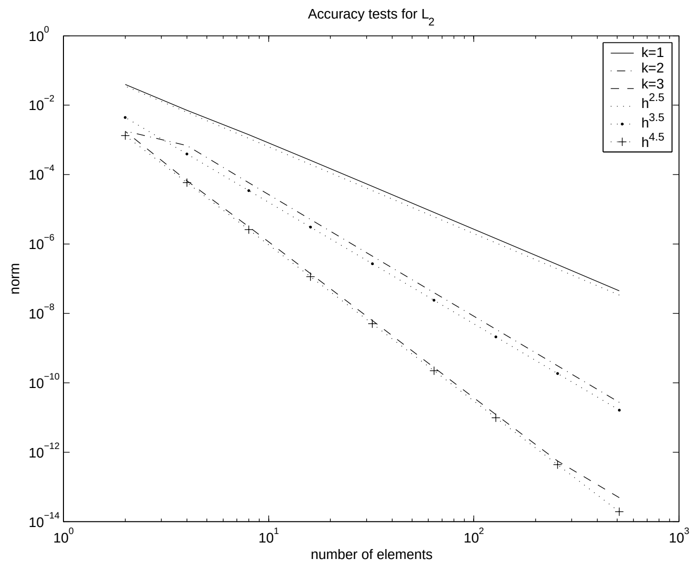


(a) L^2 norm

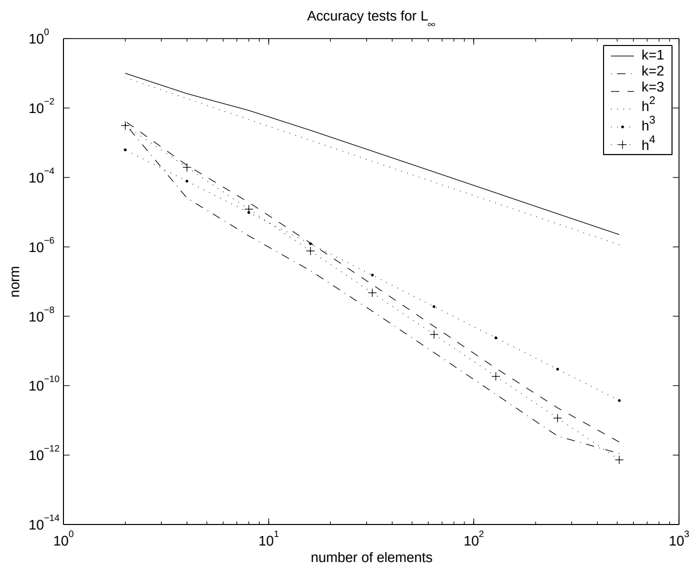


(b) L^∞ norm

Figure 5: analytical solution $u(x) = \sin(\pi x)/\pi^2$



(a) L^2 norm



(b) L^∞ norm

Figure 6: analytical solution $u(x) = \sin(2\pi x)/\pi^2 + \sin(\pi x)/\pi^2$

5 Conclusions

In this report we derive the weak formulation of two DG methods for the elliptic problem with general boundary conditions, which is a generalization of the weak formulation derived in [2].

The local lifting operator plays an important role in the stability of a DG method, but presently there is no paper available which discusses in detail how to compute this operator. In this report we derive one formulation to compute them.

We have chosen several different approaches and perform numerical experiments for the one dimensional Poisson equation with homogenous boundary conditions. As expected theoretically [2], our numerical experiments show that the Baumann-Oden approach is unstable for linear basis functions, as it gives a singular matrix for the numerical discretization. For higher order polynomials, the numerical experiments give stable solutions. This implies that this method is not suitable for computation of multidimensional problems and also for space-time DG method, where we use linear basis functions in time and space.

The Bassi-Rebay method gives a stable method with compact stencil and has rates of convergences both in the L^2 and L^∞ norms, equivalent with what we expected theoretically, in some cases even higher rates of convergence.

Acknowledgement

The first author acknowledges the financial support by the Dutch Technology Foundation (STW). This research is part of STW project TWI.5453 entitled *Analysis and Control of Transport Phenomena in Wet-Chemical Etching Processes.*

References

[1] D.N. Arnold, F. Brezzi, B. Cockburn, and L.D. Marini. *Discontinuous Galerkin Methods for Elliptic Problems*, pages 89–101. In Cockburn et al. [10], 2000.

[2] D.N. Arnold, F. Brezzi, B. Cockburn, and L.D. Marini. Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. *SIAM J. Num. Anal.*, 39(5):1749–1779, 2002.

[3] I. Babuska and M. Zlamal. Nonconforming Elements in the Finite Element Method with Penalty. *SIAM J. Num. Anal.*, 10:863–875, 1973.

[4] F. Bassi and S. Rebay. A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations. *J. Comput. Phys.*, 131:267–279, 1997.

[5] F. Bassi and S. Rebay. Numerical Evaluation of Two Discontinuous Galerkin Methods for the Compressible Navier-Stokes Equations. *Int. J. Numer. Meth. Fluids*, 40:197–207, 2002.

[6] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, and M. Savini. A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In *Proceedings of 2nd European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics*, pages 99–108, Antwerpen, Belgium, 1997. Technologisch Instituut.

[7] F. Brezzi, G. Manzini, D. Marini, P. Pietra, , and A. Russo. Discontinuous Galerkin Approximations for Elliptic Problems. *Numer. Methods Partial Differential Eq.*, 16:365–378, 2000.

[8] F. Brezzi, G. Manzini, D. Marini, P. Pietra, and A. Russo. Discontinuous galerkin approximations for diffusion problems. In *Atti Convegno in onore di F. Brioschi (Milan, 1997)*, pages 197–217, Milan, Italy, 1999. Instituto Lombardo, Accademia di Scienze e Lettere.

[9] P. Castillo, B. Cockburn, I. Perugia, and D. Schotzau. Local Discontinuous Galerkin Methods for Elliptic Problems. *Commun. Numer. Meth. Engng.*, 18:69–75, 2002.

[10] B. Cockburn, G.E. Karniadakis, and C.W. Shu, editors. *Discontinuous Galerkin Methods. Theory, Computation and Applications, Lecture Notes in Computational Science and Engineering, Vol. 11.* Springer–Verlag, New York, 2000.

[11] B. Cockburn and C.W. Shu. The Local Discontinuous Galerkin Method for Time-Dependent Convection-Diffusion Systems. *SIAM J. Num. Anal.*, 35(6):2440–2463, 1998.

[12] J. Douglas Jr. and T. Dupont. *Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods*, volume 58. Springer–Verlag, Berlin, 1976.

[13] J.T. Oden, I. Babuška, and C.E. Baumann. A Discontinuous *hp* Finite Element Method for Diffusion Problems. *J. Comput. Phys.*, 146:491–519, 1998.

[14] B. Riviere, M.F. Wheeler, and V. Girault. Improved Energy Estimates for Interior Penalty, Constrained and Discontinuous Galerkin Methods for Elliptic Problems I. *Comput. Geosci.*, 3:337–360, 1999.

[15] J.J. Sudirham, J.J.W. van der Vegt, and R.M.J. van Damme. *Space-Time Discontinuous Galerkin Method for Parabolic Problems in Moving Domain.* Dept. Applied Mathematics, Univ. Twente, The Netherlands, 2003. in preparation.

[16] J.J.W. van der Vegt. Discontinuous galerkin finite element methods. In *Lecture notes J.M. BurgersCentrum PhD Course: Computational Fluid Dynamics III.* J.M. Burgerscentrum, 2001.