Numer Algor (2012) 63:753–777
DOI 10.1007/s11075-012-9651-1

ORIGINAL PAPER

A computational study of the weak Galerkin method for second-order elliptic equations

Lin Mu · Junping Wang · Yanqiu Wang · Xiu Ye

Received: 20 October 2011 / Accepted: 18 September 2012 / Published online: 6 October 2012

Abstract The weak Galerkin finite element method is a novel numerical method that was first proposed and analyzed by Wang and Ye (2011) for general second order elliptic problems on triangular meshes. The goal of this paper is to conduct a computational investigation for the weak Galerkin method for various model problems with more general finite element partitions. The numerical results confirm the theory established in Wang and Ye (2011). The results also indicate that the weak Galerkin method is efficient, robust, and reliable in scientific computing.

The research of J. Wang was supported by the NSF IR/D program, while working at the National Science Foundation. However, any opinion, finding, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

The research of X. Ye was supported in part by the National Science Foundation under Grant No. DMS-1115097.

L. Mu
Department of Applied Science, University of Arkansas at Little Rock, Little Rock, AR 72204, USA
e-mail: lxmu@ualr.edu

J. Wang
Division of Mathematical Sciences, National Science Foundation, Arlington, VA 22230, USA
e-mail: jwang@nsf.gov

Y. Wang
Department of Mathematics, Oklahoma State University, Stillwater, OK 74078, USA
e-mail: yqwang@math.okstate.edu

X. Ye (✉)
Department of Mathematics, University of Arkansas at Little Rock, Little Rock, AR 72204, USA
e-mail: xxye@ualr.edu

Keywords Finite element methods · Weak Galerkin methods · Elliptic equations

Mathematics Subject Classifications (2010) Primary 65N30; Secondary 65N50

1 Introduction

In this paper, we are concerned with computation and numerical accuracy issues for the *weak Galerkin* method that was recently introduced in [13] for second order elliptic equations. The weak Galerkin method is an extension of the standard Galerkin finite element method where classical derivatives were substituted by weakly defined derivatives on functions with discontinuity. The weak Galerkin method is also related to the standard mixed finite element method in that the two methods are identical for simple model problems (such as the Poisson problem) using certain finite element discretizations (for example, Raviart–Thomas [12] and Brezzi–Douglas–Marini elements [4]). But they have fundamental differences for general second order elliptic equations and finite element discretizations, as we shall explain in details later. The goal of this paper is to numerically demonstrate the efficiency and accuracy of the weak Galerkin method in scientific computing. In addition, we shall extend the weak Galerkin method of [13] from triangular and tetrahedral elements to rectangular and cubic elements.

For simplicity, we take the linear second order elliptic equation as our model problem. More precisely, let Ω be an open bounded domain in $\mathbb{R}^d$, $d = 2, 3$ with Lipschitz continuous boundary $\partial\Omega$. The model problem seeks an unknown function $u = u(x)$ satisfying

$$\begin{aligned} -\nabla \cdot (\mathcal{A}\nabla u) + \boldsymbol{\beta} \cdot \nabla u + \gamma u &= f \qquad \text{in } \Omega, \\ u &= g \qquad \text{on } \partial\Omega, \end{aligned} \tag{1.1}$$

where $\mathcal{A} \in [L^\infty(\Omega)]^{d\times d}$, $\boldsymbol{\beta} \in [L^\infty(\Omega)]^d$, and $\gamma \in L^\infty(\Omega)$ are vector- and scalar-valued functions, as appropriate. Furthermore, assume that $\mathcal{A}$ is a symmetric and uniformly positive definite matrix function on Ω and the problem (1.1) has one and only one weak solution in the usual Sobolev space $H^1(\Omega)$ consisting of square integrable derivatives up to order one. f and g are given functions that ensure the desired solvability of (1.1).

Since the weak Galerkin method is a brand new method, we would like to comment on its relation with some existing methods in literature. Due to the "discontinuous" nature of the functions in the weak Galerkin finite element space, it is sensible to make comparisons with methods employing discontinuous functions, such as the mixed finite element method [5], the discontinuous Galerkin method (see the survey paper [2]), and the discontinuous Petrov–Galerkin method [6–8].

The original idea of the weak Galerkin method is motivated by the hybrid mixed finite element method. In fact, it is not hard to show that, for the Poisson equation $-\Delta u = f$, the weak Galerkin method with Σ_h defined using

Raviart–Thomas or Brezzi–Douglas–Marini elements is equivalent to the hybrid mixed finite element method [1, 5] using the same elements. In this case, one can indeed prove that $\nabla_d u_h$ is in $H(div, \Omega)$ and is equal to the discrete dual variable in the mixed finite element approximation. However, the introduction of the weak gradient operator ∇_d makes the weak Galerkin method fundamentally different from the mixed method for general second order elliptic equations. Why and how are they different? First, for second order elliptic equations with variable coefficients, these two methods give different numerical approximations. More precisely, for the equation $-\nabla \cdot (\mathcal{A}\nabla u + \boldsymbol{\beta} u) = f$ where $\mathcal{A}$ and $\boldsymbol{\beta}$ are variable matrix- and vector-coefficients, the mixed finite element method defines the dual variable $\sigma = \mathcal{A}\nabla u + \boldsymbol{\beta} u \in H(div)$ with a finite element discretization σ_h in the same space. However, in the weak Galerkin finite element method, the corresponding numerical flux is given by $\mathcal{A}(\nabla_d u_h) + \boldsymbol{\beta} u_h$ which is in general not a function in $H(div)$, though $\nabla_d u$ is locally in $H(div)$ space on each element. Second, the weak Galerkin method allows a variety of choices for S_h and Σ_h, as long as they satisfy certain conditions so that a weak gradient operator ∇_d can be defined with a certain approximation property. In this paper, we only consider the most "natural" and obvious choices based on the Raviart–Thomas and the Brezzi–Douglas–Marini elements. More general schemes and elements defined on arbitrary polygon/polyhedron have been introduced in [14] and [10].

The introduction of the weak gradient operator ∇_d allows one to freely approximate u and ∇u by using totally discontinuous functions. Another well-known numerical method which employs totally discontinuous functions is the discontinuous Galerkin method. To compare these two methods, we look at the two-dimensional problem with the lowest order elements for each of them. This amounts to the discontinuous Galerkin method using piecewise linear functions on a triangular mesh, and the weak Galerkin method defined on the same mesh using $(P_0(K_0),\ P_0(F),\ RT_0(K))$. In other words, the weak Galerkin method uses piecewise constants on both the triangles and the edges. These two elements give comparable approximation errors, as they both have $O(h)$ in the energy norm of the error and $O(h^2)$ in the L^2 norm, assuming full regularity of the solution. The total degrees of freedom for the discontinuous Galerkin method is $3 \times$ (*the number of triangles*), and for the weak Galerkin method it is (*the number of triangles*) + (*the number of edges*). Clearly, the weak Galerkin method generates a smaller linear system to solve than the discontinuous Galerkin method. As to implementation, the weak formulation of the discontinuous Galerkin method involves jumps of the primal variable and averages of the flux on mesh edges. Therefore, the computation of local stiffness matrices are no longer confined within one triangle, as it must exchange information with all neighboring triangles, through jumps and averages on edges. On the contrary, the computation of the local stiffness matrix for weak Galerkin methods is completely localized, and involves only the degrees of freedom on a given triangle and its three edges. Note that the communication with neighboring triangles is conveniently done through the introduction of edge-based degrees of freedom. Once the local

stiffness matrices are computed, one can assemble the global matrix in the same way as treating a usual primal formulation. It should be pointed out that the computation of the local stiffness matrix for weak Galerkin methods can be formulated in a standard or routine way, as to be discussed in Section 3. Explicit formulas for the local stiffness matrix for several lowest order weak Galerkin elements will be presented in Section 3.

The discontinuous Petrov–Galerkin method has recently been proposed for transport equations [6, 7], as well as for second order elliptic equations [8]. Different from the usual discontinuous Galerkin methods, the Petrov–Galerkin method uses different trial and test spaces, where the trial space is piecewisely defined using polynomials on both mesh elements and edges. For elliptic equations, the discontinuous Petrov–Galerkin method no longer needs a stabilization term, as is required in the interior penalty discontinuous Galerkin methods. This eliminates the trouble of choosing stabilization parameters that was commonly questioned to the discontinuous Galerkin method. Like the Petrov–Galerkin method, the weak Galerkin method also does not require any selection of stabilization parameters. The weak Galerkin method is quite different from the discontinuous Petrov–Galerkin method. First, the weak Galerkin method is a Ritz-Galerkin method that uses the same trial and test spaces. Second, the trial space for the weak Galerkin method only contains the primal variable, while the trial space for the discontinuous Petrov–Galerkin method contains both the primal and the dual variables. In addition, to achieve the same order of approximation accuracy, the weak Galerkin method uses much less degrees of freedom than the discontinuous Petrov–Galerkin method.

One well-known feature of the discontinuous Galerkin method or the discontinuous Petrov–Galerkin method is their ability in dealing with hybrid and even non-conformal meshes. This nice feature provides an added degree of freedom in mesh generation and refinement. It should be pointed out that the weak Galerkin method, as formulated in [10, 14], enjoys the same feature with a well developed convergence theory. Nevertheless, the weak Galerkin is still at its very early stage of development, and there are a lot outstanding issues that remain to be explored. For example, it is not clear whether the weak Galerkin method shall provide improved numerical solution to convection-dominant problems. How the idea of weak Galerkin can be used to yield more robust numerical schemes for high order of partial differential equations? What are the fast solution techniques for the discrete system arising from the weak Galerkin methods? The list can certainly go much longer. The most profound part of the weak Galerkin method is the novelty of the idea of using discrete weak gradient operators ∇_d in numerical partial differential equations. This idea can be generalized to more complicated differential equations including those with *div* and *curl* operators, as one can similarly define weak divergence and weak curl. The implementation of all these possible extensions are all based on the computation of these weak operators. The purpose of this paper is to provide a standard formula and a detailed algorithm to efficiently compute the weak gradient ∇_d, and also to validate the accuracy

and robustness of the weak Galerkin method with numerical examples. Once a standard formula for computing ∇_d has been established, one can derive similar formulas for weak divergence and weak curl in other applications.

Throughout the paper, we use $\|\cdot\|$ to denote the standard L^2 norm over the domain Ω, and use bold face Latin characters to denote vectors or vector-valued functions.

The paper is organized as follows. In Section 2, the weak Galerkin method is introduced and an abstract theory is given. In particular, we prove that certain rectangular elements satisfy the assumptions in the abstract theory, and thus establish a well-posedness and error estimate for the corresponding weak Galerkin method with rectangular meshes. In Section 3, we present some implementation details for the weak Galerkin elements. Finally in Section 4, we report some numerical results for various test problems. The numerical experiments not only confirm the theoretical predictions as given in the original paper [13], but also reveal new results that have not yet been theoretically investigated.

2 The weak Galerkin method

Let $\mathcal{T}_h$ be a shape-regular, quasi-uniform mesh of the domain Ω, with characteristic mesh size h. In two-dimension, we consider triangular and rectangular meshes, and in three-dimension, we mainly consider tetrahedral and hexahedral meshes. For each element $K \in \mathcal{T}_h$, denote by K_0 and ∂K the interior and the boundary of K, respectively. Here, K can be a triangle, a rectangle, a tetrahedron or a hexahedron. The boundary ∂K consists of several "sides", which are edges in two-dimension or faces/polygons in three-dimension. Denote by $\mathcal{F}_h$ the collection of all edges/faces in $\mathcal{T}_h$.

On each $K \in \mathcal{T}_h$, let $P_j(K_0)$ be the set of polynomials on K_0 with degree less than or equal to j, and $Q_j(K)$ be the set of polynomials on K_0 with degree of each variable less than or equal to j. Likewise, on each $F \in \mathcal{F}_h$, $P_l(F)$ and $Q_l(F)$ are defined analogously. Now, define a weak discrete space on mesh $\mathcal{T}_h$ by

$$S_h = \{v : \ v|_{K_0} \in P_j(K_0) \text{ or } Q_j(K_0) \text{ for all } K \in \mathcal{T}_h,$$
$$v|_F \in P_l(F) \text{ or } Q_l(F) \text{ for all } F \in \mathcal{F}_h\}.$$

Observe that the definition of S_h does not require any form of continuity across element or edge/face interfaces. A function in S_h is characterized by its value on the interior of each element plus its value on the edges/faces. Therefore, it is convenient to represent functions in S_h with two components, $v = \{v_0, v_b\}$, where v_0 denotes the value of v on all K_0s and v_b denotes the value of v on $\mathcal{F}_h$.

We further define an L^2 projection from $H^1(\Omega)$ onto S_h by setting $Q_h v \equiv \{Q_0 v,\ Q_b v\}$, where $Q_0 v|_K$ is the local L^2 projection of v in $P_j(K_0)$, for $K \in \mathcal{T}_h$, and $Q_b v|_F$ is the local L^2 projection in $P_l(F)$, for $F \in \mathcal{F}_h$.

The idea of the weak Galerkin method is to seek an approximate solution to (1.1) in the weak discrete space S_h. To this end, we need to introduce a discrete gradient operator on S_h. Indeed, this will be done locally on each element K. Let $V_r(K)$ be a space of polynomials on K such that $[P_r(K)]^d \subset V_r(K)$; details of $V_r(K)$ will be given later. Let

$$\Sigma_h = \{\mathbf{q} \in [L^2(\Omega)]^d : \mathbf{q}|_K \in V_r(K) \text{ for all } K \in \mathcal{T}_h\}.$$

A discrete gradient of $v_h = \{v_0, v_b\} \in S_h$ is defined to be a function $\nabla_d v_h \in \Sigma_h$ such that on each $K \in \mathcal{T}_h$,

$$\int_K \nabla_d v_h \cdot \mathbf{q}\, dx = -\int_K v_0 \nabla \cdot \mathbf{q}\, dx + \int_{\partial K} v_b \mathbf{q} \cdot \mathbf{n}\, ds, \quad \text{for all } \mathbf{q} \in V_r(K), \tag{2.1}$$

where $\mathbf{n}$ is the unit outward normal on ∂K. Clearly, such a discrete gradient is always well-defined.

Denote by $(\cdot, \cdot)$ the standard L^2-inner product on Ω. Let S_h^0 be a subset of S_h consisting of functions with vanishing boundary values. Now we can write the weak Galerkin formulation for (1.1) as follows: find $u_h = \{u_0, u_b\} \in S_h$ such that $u_b = Q_b g$ on each edge/face $F \subset \partial\Omega$ and

$$(\mathcal{A}\nabla_d u_h, \nabla_d v_h) + (\boldsymbol{\beta} \cdot \nabla_d u_h, v_0) + (\gamma u_0, v_0) = (f, v_0) \tag{2.2}$$

for all $v_h = \{v_0, v_b\} \in S_h^0$. For simplicity of notation, we introduce the following bilinear form

$$a(u_h, v_h) \triangleq (\mathcal{A}\nabla_d u_h, \nabla_d v_h) + (\boldsymbol{\beta} \cdot \nabla_d u_h, v_0) + (\gamma u_0, v_0). \tag{2.3}$$

The spaces S_h and Σ_h can not be chosen arbitrarily. There are certain criteria they need to follow, in order to guarantee that (2.2) provides a good approximation to the solution of (1.1). For example, Σ_h has to be rich enough to prevent from the loss of information in the process of taking discrete gradients, while it should remain to be sufficiently small for its computational cost. Hence, we would like to impose the following conditions upon S_h and Σ_h:

(P1) For any $v_h \in S_h$ and $K \in \mathcal{T}_h$, $\nabla_d v_h|_K = 0$ if and only if $v_0 = v_b = \textit{constant}$ on K.

(P2) For any $w \in H^{m+1}(\Omega)$, where $0 \le m \le j+1$, we have

$$\|\nabla_d(Q_h w) - \nabla w\| \le C h^m \|w\|_{m+1},$$

where and in what follows of this paper, C denotes a generic constant independent of the mesh size h.

Under the above two assumptions, it has been proved in [13] that (2.2) has a unique solution as long as the mesh size h is moderately small and the dual of (1.1) has an H^{1+s}-regularity with some $s > 0$. Furthermore, one has the following error estimate:

$$\begin{aligned} \|\nabla_d(u_h - Q_h u)\| &\le C\left(h^{1+s}\|f - Q_0 f\| + h^m \|u\|_{m+1}\right), \\ \|u_0 - Q_0 u\| &\le C\left(h^{1+s}\|f - Q_0 f\| + h^{m+s}\|u\|_{m+1}\right), \end{aligned} \tag{2.4}$$

for any $0 \le m \le j+1$, and $s > 0$ is the largest number such that the dual of (1.1) has an H^{1+s}-regularity.

There are several possible combinations of S_h and Σ_h that satisfy Assumptions **(P1)** and **(P2)**. Two examples of triangular elements have been given in [13], which are

1. Triangular element $(P_j(K_0),\ P_j(F),\ RT_j(K))$ for $j \ge 0$. That is, in the definition of S_h, we set $l = j$. And in the definition of Σ_h, we set $r = j$ and $V_r(K)$ to be the jth order Raviart–Thomas element $RT_j(K)$ [12].
2. Triangular element $(P_j(K_0),\ P_{j+1}(F),\ (P_{j+1}(K))^d)$ for $j \ge 0$. That is, in the definition of S_h, we set $l = j+1$. And in the definition of Σ_h, we set $r = j+1$ and $V_r(K) = (P_{j+1}(K))^d$, or in other words, the $(j+1)$st order Brezzi–Douglas–Marini element [4].

Next, we shall extend this result to rectangular elements. An extension to three-dimensional tetrahedral and hexahedral elements is straightforward.

2.1 Weak Galerkin on rectangular meshes

Consider the following two type of rectangular elements:

1. Rectangular element $(Q_j(K_0),\ Q_j(F),\ RT_j(K))$ for $j \ge 0$. That is, in the definition of S_h, we set $l = j$, and in the definition of Σ_h, we set $r = j$ and $V_r(K)$ to be the jth order Raviart–Thomas element $RT_j(K)$ on rectangle K.
2. Rectangular element $(P_j(K_0),\ P_{j+1}(F),\ BDM_{j+1}(K))$ for $j \ge 0$. That is, in the definition of S_h, we set $l = j+1$, and in the definition of Σ_h, we set $r = j+1$ and $V_r(K)$ to be the $(j+1)$st order Brezzi–Douglas–Marini element $BDM_{j+1}(K)$ on rectangle K.

Denote by $Q_{i,j}(K)$ the space of polynomials with degree in x and y less than or equal to i and j, respectively, and $\mathbf{curl} = \begin{bmatrix} -\partial/\partial y \\ \partial/\partial x \end{bmatrix}$. It is known that

$$RT_j(K) = \begin{bmatrix} Q_{j+1,j}(K) \\ Q_{j,j+1}(K) \end{bmatrix},$$

$$BDM_{j+1}(K) = \begin{bmatrix} P_{j+1}(K) \\ P_{j+1}(K) \end{bmatrix} + span\left\{\mathbf{curl}\,(x^{j+2}y),\ \mathbf{curl}\,(xy^{j+2})\right\},$$

and $\dim(RT_j(K)) = 2(j+1)(j+2)$, $\dim(BDM_{j+1}(K)) = (j+2)(j+3)+2$. The degrees of freedom for $RT_j(K)$ are:

$$\int_F (\mathbf{q}\cdot\mathbf{n})w\,ds, \qquad \text{for all } w \in Q_j(F),\ F \in \mathcal{F} \cap \partial K,$$

$$\int_K \mathbf{q}\cdot\mathbf{p}\,dx, \qquad \text{for all } \mathbf{p} \in Q_{j-1,j}(K) \times Q_{j,j-1}(K).$$

The degrees of freedom for $BDM_{j+1}(K)$ are

$$\int_F (\mathbf{q}\cdot\mathbf{n})w\,ds, \qquad \text{for all } w\in P_{j+1}(F),\ F\in\mathcal{F}\cap\partial K,$$

$$\int_K \mathbf{q}\cdot\mathbf{p}\,dx, \qquad \text{for all } \mathbf{p}\in[P_{j-1}(K)]^2.$$

It is also well-known that on each rectangle $K\in\mathcal{T}_h$ and each edge $F\in\mathcal{F}_h\cap\partial K$,

$$\begin{aligned} \nabla\cdot RT_j(K) &= Q_j(K_0), & RT_j(K)\cdot\mathbf{n}|_F &= Q_j(F),\\ \nabla\cdot BDM_{j+1}(K) &= P_j(K_0), & BDM_{j+1}(K)\cdot\mathbf{n}|_F &= P_{j+1}(F). \end{aligned} \tag{2.5}$$

Next, we show that the two set of elements defined as above satisfy Assumptions **(P1)** and **(P2)**.

Lemma 2.1 *For the two type of rectangular elements given in this subsection, the Assumption* **(P1)** *holds true.*

Proof If $v_0=v_b=$ *constant* on K, then clearly $\nabla_d v_h|_K$ vanishes since the right-hand side of (2.1) is zero from the divergence theorem. Now let us assume that $\nabla_d v_h|_K=0$. By (2.1) and using integration by parts, we have for all $\mathbf{q}\in RT_j(K)$ or $BDM_{j+1}(K)$,

$$\begin{aligned} 0 &= -\int_K v_0\nabla\cdot\mathbf{q}\,dx+\int_{\partial K} v_b\mathbf{q}\cdot\mathbf{n}\,ds\\ &= \int_{\partial K}(v_b-v_0)\mathbf{q}\cdot\mathbf{n}\,ds+\int_K(\nabla v_0)\cdot\mathbf{q}\,dx. \end{aligned} \tag{2.6}$$

We first consider the element $(Q_j(K_0),\ Q_j(F),\ RT_j(K))$. If $j=0$, then v_0 is a constant on K_0 and clearly $\nabla v_0=\mathbf{0}$. If $j>0$, take $\mathbf{q}$ such that $\int_F(\mathbf{q}\cdot\mathbf{n})w\,ds=0$ for all $w\in Q_j(F)$ and let it traverse through all degrees of freedom defined by $\int_K\mathbf{q}\cdot\mathbf{p}\,dx$, for $\mathbf{p}\in Q_{j-1,j}(K)\times Q_{j,j-1}(K)$. Since $(v_b-v_0)|_F\in Q_j(F)$ and $\nabla v_0\in Q_{j-1,j}(K)\times Q_{j,j-1}(K)$, (2.6) gives $\nabla v_0=\mathbf{0}$, which implies that v_0 is a constant on K_0. Now (2.6) reduces into

$$\int_{\partial K}(v_b-v_0)\mathbf{q}\cdot\mathbf{n}\,ds=0, \qquad \text{for all } \mathbf{q}\in RT_j(K).$$

Next, since $(v_b-v_0)|_F\in Q_j(F)=RT_j(K)\cdot\mathbf{n}|_F$ for all $F\in\mathcal{F}_h\cap\partial K$, by letting $\mathbf{q}$ traverse through all degrees of freedom on ∂K, we have $v_b-v_0=0$ on all F. This implies $v_b=v_0=$ *constant* in K.

For the $(P_j(K_0),\ P_{j+1}(F),\ BDM_{j+1}(K))$ element, using the same argument as in the previous case, and noticing that $\nabla v_0 \in (P_{j-1}(K))^2$, $(v_b - v_0)|_F \in P_{j+1}(F) = BDM_{j+1}(K) \cdot \mathbf{n}|_F$ for all $F \in \mathcal{F}_h \cap \partial K$, we can similarly prove that $v_b = v_0 = constant$ in K. □

Lemma 2.2 *For the two type of rectangular elements given in this subsection, the Assumption* **(P2)** *holds true.*

Proof Let $w \in H^m(\Omega), 1 \le m \le j+1$. For any $\mathbf{q} \in \Sigma_h$ and $K \in \mathcal{T}_h$, by (2.5) and the definition of L^2 projections, we have

$$\begin{aligned}\int_K (\nabla_d Q_h w) \cdot \mathbf{q}\, dx &= -\int_K (Q_0 w)(\nabla \cdot \mathbf{q})\, dx + \int_{\partial K} Q_b w (\mathbf{q} \cdot \mathbf{n})\, ds \\ &= -\int_K w(\nabla \cdot \mathbf{q})\, dx + \int_{\partial K} w(\mathbf{q} \cdot \mathbf{n})\, ds \\ &= \int_K (\nabla w) \cdot \mathbf{q}\, dx.\end{aligned}$$

In other words, on each $K \in \mathcal{T}_h$, $\nabla_d Q_h w$ is the L^2 projection of ∇w onto $RT_j(K)$ or $BDM_{j+1}(K)$. Thus, the Assumption **(P2)** follows immediately from the approximation properties of the L^2 projection, and the fact that both $RT_j(K)$ and $BDM_{j+1}(K)$ contains the entire polynomial space $[P_j(K)]^2$. □

Using Lemmas 2.1 and 2.2, one can derive the error estimate (2.4) for the rectangular elements by following the argument presented in [13]. Details are left to interested readers as an exercise.

3 Computation of local stiffness matrices

Similar to the standard Galerkin finite element method, the weak Galerkin method (2.2) can be implemented as a matrix problem where the matrix is given as the sum of local stiffness matrices on each element $K \in \mathcal{T}_h$. Thus, a key step in the computer implementation of the weak Galerkin is to compute element stiffness matrices. The goal of this section is to demonstrate ways of computing element stiffness matrices for various elements introduced in the previous sections.

For a given element $K \in \mathcal{T}_h$, let $\phi_{0,i}, i = 1, \ldots, N_0$, be a set of basis functions for $P_j(K_0)$ or $Q_j(K_0)$, and $\phi_{b,i}, i = 1, \ldots, N_b$, be a set of basis functions for $\sum_{F \in \partial K \cap \mathcal{F}_h} P_l(F)$ or $\sum_{F \in \partial K \cap \mathcal{F}_h} Q_l(F)$. Note that $\{\phi_{b,i}\}$ is the union of basis functions from all edges/faces of element K. Then every $v_h = \{v_0, v_b\} \in S_h$ has the following representation in K:

$$v_h|_K = \left\{ \sum_{i=1}^{N_0} v_{0,i}\phi_{0,i}, \sum_{i=1}^{N_b} v_{b,i}\phi_{b,i} \right\}.$$

On each K, the local stiffness matrix M_K for (2.2) can thus be written as a block matrix

$$M_K = \begin{bmatrix} M_{0,0} & M_{0,b} \\ M_{b,0} & M_{b,b} \end{bmatrix} \tag{3.1}$$

where $M_{0,0}$ is an $N_0 \times N_0$ matrix, $M_{0,b}$ is an $N_0 \times N_b$ matrix, $M_{b,0}$ is an $N_b \times N_0$ matrix, and $M_{b,b}$ is an $N_b \times N_b$ matrix. These matrices are defined, respectively, by

$$M_{0,0} = \left[a(\phi_{0,j}, \phi_{0,i})_K\right]_{i,j}, \qquad M_{0,b} = \left[a(\phi_{b,j}, \phi_{0,i})_K\right]_{i,j},$$
$$M_{b,0} = \left[a(\phi_{0,j}, \phi_{b,i})_K\right]_{i,j}, \qquad M_{b,b} = \left[a(\phi_{b,j}, \phi_{b,i})_K\right]_{i,j},$$

where the bilinear form $a(\cdot,\cdot)$ is defined as in (2.3), and i, j are the row and column indices, respectively.

To compute each block of M_K, we first need to calculate the discrete gradient operator ∇_d. For convenience, denote the local vector representation of $v_h|_K$ by

$$\underline{\mathbf{v}}_0 = \begin{bmatrix} v_{0,1} \\ v_{0,2} \\ \vdots \\ v_{0,N_0} \end{bmatrix}, \qquad \underline{\mathbf{v}}_b = \begin{bmatrix} v_{b,1} \\ v_{b,2} \\ \vdots \\ v_{b,N_b} \end{bmatrix}.$$

Let $\boldsymbol{\chi}_i$, $i = 1, \ldots, N_V$, be a set of basis functions for $V_r(K)$. Then, for every $\mathbf{q}_h \in \Sigma_h$, its value on K can be expressed as

$$\mathbf{q}_h|_K = \sum_{i=1}^{N_V} q_i \boldsymbol{\chi}_i.$$

Similarly, we denote the local vector representation of $\mathbf{q}_h|_K$ by

$$\underline{\mathbf{q}} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{N_V} \end{bmatrix},$$

Then, by the definition of the discrete gradient (2.1), given $v_h|_K$, we can compute the vector form of $\nabla_d v_h$ on K by

$$D_K(\underline{\nabla_d v_h}) = -Z_K \underline{\mathbf{v}}_0 + T_K \underline{\mathbf{v}}_b, \tag{3.2}$$

where the $N_V \times N_V$ matrix D_K, the $N_V \times N_0$ matrix Z_K, and the $N_V \times N_b$ matrix T_K are defined, respectively, by

$$D_K = \begin{bmatrix} \int_K \boldsymbol{\chi}_1 \cdot \boldsymbol{\chi}_1 \, dx & \cdots & \int_K \boldsymbol{\chi}_1 \cdot \boldsymbol{\chi}_{N_V} \, dx \\ \cdots & \cdots & \cdots \\ \int_K \boldsymbol{\chi}_{N_V} \cdot \boldsymbol{\chi}_1 \, dx & \cdots & \int_K \boldsymbol{\chi}_{N_V} \cdot \boldsymbol{\chi}_{N_V} \, dx \end{bmatrix}, \tag{3.3}$$

$$Z_K = \begin{bmatrix} \int_K (\nabla \cdot \boldsymbol{\chi}_1)\phi_{0,1}\,dx & \cdots & \int_K (\nabla \cdot \boldsymbol{\chi}_1)\phi_{0,N_0}\,dx \\ \cdots & \cdots & \cdots \\ \int_K (\nabla \cdot \boldsymbol{\chi}_{N_V})\phi_{0,1}\,dx & \cdots & \int_K (\nabla \cdot \boldsymbol{\chi}_{N_V})\phi_{0,N_0}\,dx \end{bmatrix},$$

and

$$T_K = \begin{bmatrix} \int_{\partial K} (\boldsymbol{\chi}_1 \cdot \mathbf{n})\phi_{b,1}\,ds & \cdots & \int_{\partial K} (\boldsymbol{\chi}_1 \cdot \mathbf{n})\phi_{b,N_b}\,ds \\ \cdots & \cdots & \cdots \\ \int_{\partial K} (\boldsymbol{\chi}_{N_V} \cdot \mathbf{n})\phi_{b,1}\,ds & \cdots & \int_{\partial K} (\boldsymbol{\chi}_{N_V} \cdot \mathbf{n})\phi_{b,N_b}\,ds \end{bmatrix}.$$

Notice that D_K is a symmetric matrix.

Once the matrices D_K, Z_K and T_K are computed, we can use (3.2) to calculate the weak gradient of basis functions $\phi_{0,i}$ and $\phi_{b,i}$ on K. It is not hard to see that

$$\underline{(\nabla_d \phi_{0,i})} = -D_K^{-1} Z_K \underline{\mathbf{e}}_i^{N_0}, \qquad \underline{(\nabla_d \phi_{b,i})} = D_K^{-1} T_K \underline{\mathbf{e}}_i^{N_b}, \tag{3.4}$$

where $\underline{\mathbf{e}}_i^{N_0}$ and $\underline{\mathbf{e}}_i^{N_b}$ are the standard basis for the Euclidean spaces $\mathbb{R}^{N_0}$ and $\mathbb{R}^{N_b}$, respectively, such that its i-th entry is 1 and all other entries are 0.

Define matrices

$$\begin{aligned} A_K &= \left[(\mathcal{A}\boldsymbol{\chi}_j, \boldsymbol{\chi}_i)_K\right]_{i,j}, \\ B_K &= \left[(\boldsymbol{\beta} \cdot \boldsymbol{\chi}_j, \phi_{0,i})_K\right]_{i,j}, \\ C_K &= \left[(\gamma\phi_{0,j}, \phi_{0,i})_K\right]_{i,j}, \end{aligned} \tag{3.5}$$

where $(\cdot,\cdot)_K$ denote the standard inner-product on $L^2(K)$ or $[L^2(K)]^d$, as appropriate. Clearly, A_K is an $N_V \times N_V$ matrix, B_K is an $N_0 \times N_V$ matrix, and C_K is an $N_0 \times N_0$ matrix. Then, an elementary matrix calculation shows that the local stiffness matrix M_K for (2.2) can be expressed in a way as specified in the following lemma.

Lemma 3.1 *The local stiffness matrix M_K defined in (3.1) can be computed by using the following formula*

$$\begin{aligned} M_{0,0} &= Z_K^t D_K^{-t} A_K D_K^{-1} Z_K - B_K D_K^{-1} Z_K + C_K, \\ M_{0,b} &= -Z_K^t D_K^{-t} A_K D_K^{-1} T_K + B_K D_K^{-1} T_K, \\ M_{b,0} &= -T_K^t D_K^{-t} A_K D_K^{-1} Z_K + T_K^t D_K^{-t} B_K^t, \\ M_{b,b} &= T_K^t D_K^{-t} A_K D_K^{-1} T_K, \end{aligned} \tag{3.6}$$

where the superscript t stands for the standard matrix transpose.

For the Poisson equation $-\Delta u = f$, we clearly have $A_K = D_K$ and $B_K = 0$, $C_K = 0$. Since D_K is symmetric, the local stiffness matrix becomes

$$M_K = \begin{bmatrix} Z_K^t D_K^{-1} Z_K & -Z_K^t D_K^{-1} T_K \\ -T_K^t D_K^{-1} Z_K & T_K^t D_K^{-1} T_K \end{bmatrix}. \tag{3.7}$$

In the rest of this section, we shall demonstrate the computation of the element stiffness matrix M_K with two concrete examples.

3.1 For the triangular element ($\mathbf{P_0(K)}$, $\mathbf{P_0(F)}$, $\mathbf{RT_0(K)}$)

Let K be a triangular element in $\mathcal{T}_h$. We consider the case when $j = l = 0$ and $V_r(K)$ being the lowest order Raviart–Thomas element. In other words, the discrete space S_h consists of piecewise constants on the triangles, and piecewise constants on the edges of the mesh. In this case, the discrete gradient is defined by using the lowest order Raviart–Thomas element on the triangle K. Clearly, we have $N_0 = 1$, $N_b = 3$ and $N_V = 3$.

Let $v_i = (x_i, y_i)$, $i = 1, 2, 3$, be the vertices of the triangle K and e_i be the edge opposite to the vertex v_i. Denote by $|e_i|$ the length of edge e_i and $|K|$ the area of the triangle K. We also denote by $\mathbf{n}_i$ and $\mathbf{t}_i$ the unit outward normal and unit tangential vectors on e_i, respectively. Here $\mathbf{t}_i$ should be in the positive (counterclockwise) orientation. If edge e_i goes from vertex v_j to v_k and K stays on the left when one travels from v_j to v_k, then it is not hard to see that

$$\mathbf{t}_i = \begin{bmatrix} t_{i,1} \\ t_{i,2} \end{bmatrix} = \frac{1}{|e_i|} \begin{bmatrix} x_k - x_j \\ y_k - y_j \end{bmatrix}, \qquad \mathbf{n}_i = \begin{bmatrix} n_{i,1} \\ n_{i,2} \end{bmatrix} = \frac{1}{|e_i|} \begin{bmatrix} y_k - y_j \\ -(x_k - x_j) \end{bmatrix}.$$

3.1.1 Approach I

One may use the following set of basis functions for the weak discrete functions on K:

$$\phi_{0,1} = 1, \qquad \phi_{b,i} = \begin{cases} 1 & \text{on } e_i \\ 0 & \text{otherwise} \end{cases} \qquad \text{for } i = 1, 2, 3, \tag{3.8}$$

and

$$\boldsymbol{\chi}_i = \frac{|e_i|}{2|K|} \begin{bmatrix} x - x_i \\ y - y_i \end{bmatrix}, \quad \text{for } i = 1, 2, 3. \tag{3.9}$$

Notice that $\boldsymbol{\chi}_i$ forms the standard basis for the lowest order Raviart–Thomas element, for which the degrees of freedom are taken to be the normal component on edges. Indeed, $\boldsymbol{\chi}_i$ satisfies

$$\boldsymbol{\chi}_i \cdot \mathbf{n}_j|_{e_j} = \begin{cases} 1 & \text{for } i = j, \\ 0 & \text{for } i \neq j. \end{cases}$$

It is straight forward to compute that, for the above defined basis functions,

$$Z_K = \begin{bmatrix} |e_1| \\ |e_2| \\ |e_3| \end{bmatrix}, \qquad T_K = \begin{bmatrix} |e_1| & 0 & 0 \\ 0 & |e_2| & 0 \\ 0 & 0 & |e_3| \end{bmatrix}.$$

The computation of D_K is slightly more complicated, but it can still be done without much difficulty, especially with the help of symbolic computing tools provided in existing software packages such as Maple and Mathematica. For simplicity of notation, denote

$$\begin{aligned} l_i &= |e_i|^2 && \text{for } 1 \le i \le 3, \\ l_{ij} &= |e_i|^2 + |e_j|^2 && \text{for } 1 \le i, j \le 3 \text{ and } i \ne j, \\ l_{123} &= |e_1|^2 + |e_2|^2 + |e_3|^2. \end{aligned}$$

Then, it can be verified that

$$\begin{aligned} D_K &= \frac{1}{48|K|} \begin{bmatrix} |e_1|^2 (3l_{23} - l_1) & |e_1||e_2| (l_{12} - 3l_3) & |e_1||e_3| (l_{13} - 3l_2) \\ |e_1||e_2| (l_{12} - 3l_3) & |e_2|^2 (3l_{13} - l_2) & |e_2||e_3| (l_{23} - 3l_1) \\ |e_1||e_3| (l_{13} - 3l_2) & |e_2||e_3| (l_{23} - 3l_1) & |e_3|^2 (3l_{12} - l_3) \end{bmatrix} \\ &= \frac{1}{48|K|} T_K \begin{bmatrix} 3l_{23} - l_1 & l_{12} - 3l_3 & l_{13} - 3l_2 \\ l_{12} - 3l_3 & 3l_{13} - l_2 & l_{23} - 3l_1 \\ l_{13} - 3l_2 & l_{23} - 3l_1 & 3l_{12} - l_3 \end{bmatrix} T_K^t. \end{aligned} \tag{3.10}$$

We point out that, the value of D_K given as in (3.10) agrees with the one presented in [3]. A verification of the formula (3.10) can be carried out by using the following fact

$$|K| = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix},$$

$$|e_i|^2 = (x_j - x_k)^2 + (y_j - y_k)^2,$$
$$i = 1, 2, 3, \quad j \ne k, \quad j \text{ and } k \text{ different from } i.$$

In computer implementation, it is convenient to use a form for the local matrix that can be expressed by using only edge lengths, as the one given by (3.10).

In addition, using symbolic computing tools and the law of sines and cosines, we can write D_K^{-1} as follows:

$$D_K^{-1} = T_K^{-t} \left(\frac{16|K|}{l_{123}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + \frac{1}{2|K|} \begin{bmatrix} 2l_1 & l_3 - l_{12} & l_2 - l_{13} \\ l_3 - l_{12} & 2l_2 & l_1 - l_{23} \\ l_2 - l_{13} & l_1 - l_{23} & 2l_3 \end{bmatrix} \right) T_K^{-1}.$$

Thus, to compute the local stiffness matrix M_K, it suffices to calculate A_K, B_K and C_K as given in (3.5), and then apply Lemma 3.1. Notice that these three

matrices depend on the coefficients $\mathcal{A}$, $\boldsymbol{\beta}$ and γ, and quadrature rules may be employed in the calculation. However, for the simple case of the Poisson equation $-\Delta u = f$, we see from (3.7) that

$$M_{00} = \left[\frac{144|K|}{l_{123}}\right], \qquad M_{0b} = M_{b0}^t = \left[\frac{-48|K|}{l_{123}} \ \frac{-48|K|}{l_{123}} \ \frac{-48|K|}{l_{123}}\right],$$

$$M_{bb} = \frac{16|K|}{l_{123}}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} + \frac{1}{2|K|}\begin{bmatrix}2l_1 & l_3-l_{12} & l_2-l_{13}\\ l_3-l_{12} & 2l_2 & l_1-l_{23}\\ l_2-l_{13} & l_1-l_{23} & 2l_3\end{bmatrix}.$$

3.1.2 Approach II

We would like to present another approach for computing the local stiffness matrix M_K in the rest of this subsection. Observe that a set of basis functions for the local space $V_r(K)$ can be chosen as follows

$$\boldsymbol{\chi}_1 = \begin{bmatrix}1\\0\end{bmatrix}, \qquad \boldsymbol{\chi}_2 = \begin{bmatrix}0\\1\end{bmatrix}, \qquad \boldsymbol{\chi}_3 = \begin{bmatrix}x-\bar{x}\\ y-\bar{y}\end{bmatrix}, \tag{3.11}$$

where $(\bar{x} = (x_1+x_2+x_3)/3, \bar{y} = (y_1+y_2+y_3)/3)$ is the coordinate of the barycenter of K. Note that both components of $\boldsymbol{\chi}_3$ have mean value zero on K. For the weak discrete function on K, we use the same set of basis functions as given in (3.8). It is not hard to see that

$$D_K = |K|\begin{bmatrix}1&0&0\\0&1&0\\0&0&\frac{l_{123}}{36}\end{bmatrix}, \qquad Z_K = \begin{bmatrix}0\\0\\2|K|\end{bmatrix},$$

and

$$T_K = \begin{bmatrix}y_3-y_2 & y_1-y_3 & y_2-y_1\\ x_2-x_3 & x_3-x_1 & x_1-x_2\\ \frac{2|K|}{3} & \frac{2|K|}{3} & \frac{2|K|}{3}\end{bmatrix}.$$

Next, we use the formula (3.5) to calculate the matrices A_K, B_K and C_K for the new basis (3.11). Finally, we calculate the local stiffness matrix M_K by using the formula provided in Lemma 3.1.

Since the set of basis functions for the weak discrete space are the same in Approaches I and II, the resulting local stiffness matrix M_K would remain unchanged from Approaches I and II. The set of basis functions (3.11) is advantageous over the set (3.9) in that the matrix D_K is a diagonal one whose inverse in trivial to compute.

3.2 For the cubic element ($\mathbf{Q_0(K)}$, $\mathbf{Q_0(F)}$, $\mathbf{RT_0(K)}$)

Let $K = [0, a] \times [0, b] \times [0, c]$ be a rectangular box where a, b, c are positive real numbers. We consider the three-dimensional cubic element, for which the discrete space S_h consists of piecewise constants on K_0 and piecewise constants on the faces of K. The space for the discrete gradient is the lowest order Raviart–Thomas element on K. We clearly have $N_0 = 1$, $N_b = 6$ and $N_V = 6$.

Denote the six faces $F_i, i = 1, \ldots, 6$ by

$$F_1 : x = 0, \qquad F_2 : x = a,$$
$$F_3 : y = 0, \qquad F_4 : y = b,$$
$$F_5 : z = 0, \qquad F_6 : z = c.$$

Note that the volume of K is given by $|K| = abc$ and the normal direction to each face is given by

$$\mathbf{n}_1 = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}, \ \mathbf{n}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \ \mathbf{n}_3 = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, \ \mathbf{n}_4 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \ \mathbf{n}_5 = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \ \mathbf{n}_6 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

We adopt the following set of basis functions for the weak discrete space on K

$$\phi_{0,1} = 1, \qquad \phi_{b,i} = \begin{cases} 1 & \text{on } F_i \\ 0 & \text{otherwise} \end{cases} \qquad \text{for } i = 1, \ldots, 6,$$

and

$$\boldsymbol{\chi}_1 = \begin{bmatrix} \frac{x}{a} - 1 \\ 0 \\ 0 \end{bmatrix}, \ \boldsymbol{\chi}_2 = \begin{bmatrix} \frac{x}{a} \\ 0 \\ 0 \end{bmatrix}, \ \boldsymbol{\chi}_3 = \begin{bmatrix} 0 \\ \frac{y}{b} - 1 \\ 0 \end{bmatrix},$$

$$\boldsymbol{\chi}_4 = \begin{bmatrix} 0 \\ \frac{y}{b} \\ 0 \end{bmatrix}, \ \boldsymbol{\chi}_5 = \begin{bmatrix} 0 \\ 0 \\ \frac{z}{c} - 1 \end{bmatrix}, \ \boldsymbol{\chi}_6 = \begin{bmatrix} 0 \\ 0 \\ \frac{z}{c} \end{bmatrix}.$$

Clearly, each $\boldsymbol{\chi}_i$ satisfies

$$\boldsymbol{\chi}_i \cdot \mathbf{n}_j|_{F_j} = \begin{cases} 1 & \text{for } i = j, \\ 0 & \text{for } i \neq j. \end{cases}$$

It is not hard to compute that

$$D_K = \frac{|K|}{6} \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & 2 \end{bmatrix}, \qquad D_K^{-1} = \frac{2}{|K|} \begin{bmatrix} 2 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 \end{bmatrix},$$

and

$$Z_K = \begin{bmatrix} bc \\ bc \\ ac \\ ac \\ ab \\ ab \end{bmatrix}, \qquad T_K = \begin{bmatrix} bc & 0 & 0 & 0 & 0 & 0 \\ 0 & bc & 0 & 0 & 0 & 0 \\ 0 & 0 & ac & 0 & 0 & 0 \\ 0 & 0 & 0 & ac & 0 & 0 \\ 0 & 0 & 0 & 0 & ab & 0 \\ 0 & 0 & 0 & 0 & 0 & ab \end{bmatrix}$$

Then, the local stiffness matrix M_K can be computed using the formula presented in Lemma 3.1.

4 Numerical experiments

In this section, we shall report some numerical results for the weak Galerkin finite element method on a variety of testing problems, with different mesh and finite elements. To this end, let $u_h = \{u_0, u_b\}$ and u be the solution to the weak Galerkin equation (2.2) and the original equation (1.1), respectively. Define the error by $e_h = u_h - Q_h u = \{e_0,\ e_b\}$ where $Q_h u$ is the L^2 projection of u onto appropriately defined spaces. Let us introduce the following norms:

$$H^1 \text{ semi-norm:} \qquad \|\nabla_d e_h\| = \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla_d e_h|^2 \, dx \right)^{1/2},$$

$$\text{Element-based } L^2 \text{ norm :} \qquad \|e_0\| = \left(\sum_{K \in \mathcal{T}_h} \int_K |e_0|^2 \, dx \right)^{1/2},$$

$$\text{Edge/Face-based } L^2 \text{ norm :} \qquad \|e_b\| = \left(\sum_{F \in \mathcal{F}_h} h_K \int_F |e_b|^2 \, ds \right)^{1/2},$$

where in the definition of $\|e_b\|$, h_K stands for the size of the element K that takes F as an edge/face. We shall also compute the error in the following norms

$$\|\nabla_d u_h - \nabla u\| = \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla_d u_h - \nabla u|^2 \, dx \right)^{1/2},$$

$$\|u_h - u\| = \left(\sum_{K \in \mathcal{T}_h} \int_K |u_0 - u|^2 \, dx \right)^{1/2},$$

$$\|e_0\|_\infty = \sup_{\substack{x \in K_0 \\ K \in \mathcal{T}_h}} |e_0(x)|.$$

Here the maximum norm $\|e_0\|_\infty$ is computed over all Gaussian points, and all other integrals are calculated with a Gaussian quadrature rule that is of high order of accuracy so that the error from the numerical integration can be virtually ignored.

4.1 Case 1: model problems with various boundary conditions

First, we consider the Laplace equation with nonhomogeneous Dirichlet boundary condition:

$$u = g \qquad \text{on } \partial\Omega. \tag{4.1}$$

We introduce a discrete Dirichlet boundary data g_h, which is either the usual nodal value interpolation, or the L^2 projection of $u = g$ on the boundary. Let $\Gamma \subset \partial\Omega$ and define

$$\begin{aligned} S_{g_h,\Gamma,h} = \{v :\ & v|_{K_0} \in P_j(K_0) \text{ or } Q_j(K_0) \text{ for all } K \in \mathcal{T}_h, \\ & v|_F \in P_l(F) \text{ or } Q_l(F) \text{ for all } F \in \mathcal{F}_h, \\ & v = g_h \text{ on } \mathcal{F}_h \cap \Gamma\}. \end{aligned}$$

When $\Gamma = \partial\Omega$, we simply denote $S_{g_h,\partial\Omega,h}$ by $S_{g_h,h}$. The discrete Galerkin formulation for the nonhomogeneous Dirichlet boundary value problem can be written as: find $u_h \in S_{g_h,h}$ such that for all $v_h \in S_h^0$,

$$(\mathcal{A}\nabla_d u_h, \nabla_d v_h) + (\boldsymbol{\beta} \cdot \nabla_d u_h, v_0) + (\gamma u_0,\ v_0) = (f,\ v_0).$$

We would like to see how the weak Galerkin approximation might be affected when the boundary data $u = g$ is approximated with different schemes (nodal interpolation verses L^2 projection). To this end, we use a two dimensional test problem with domain $\Omega = (0, 1) \times (0, 1)$ and exact solution given by $u = \sin(2\pi x + \pi/2)\sin(2\pi y + \pi/2)$. A uniform triangular mesh and the element $(P_0(K),\ P_0(F),\ RT_0(K))$ is used in the weak Galerkin discretization. The results are reported in Tables 1 and 2. It can be seen that both approximations of the Dirichlet boundary data give optimal order of convergence for the weak Galerkin method, while the L^2 projection method yields a slightly smaller error in $\|e_0\|$ and $\|e_b\|$.

Table 1 Case 1. Numerical results with Dirichlet data being approximated by the usual nodal point interpolation

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	7.14e−01	2.16e−02	4.05e−02	1.01e+0	1.30e−01	4.43e−02
1/16	3.56e−01	5.61e−03	1.01e−02	5.04e−01	6.53e−02	1.12e−02
1/32	1.78e−01	1.41e−03	2.53e−03	2.51e−01	3.27e−02	2.86e−03
1/64	8.90e−02	3.55e−04	6.32e−04	1.25e−01	1.63e−02	7.15e−04
1/128	4.45e−02	8.88e−05	1.57e−04	6.29e−02	8.18e−03	1.79e−04
$O(h^r), r =$	1.0012	1.9837	2.0014	1.0024	0.9984	1.9879

Table 2 Case 1. Numerical results with Dirichlet data being approximated by L^2 projection

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	7.10e−01	1.75e−02	3.08e−02	1.01e+0	1.29e−01	3.68e−02
1/16	3.55e−01	4.59e−03	7.69e−03	5.04e−01	6.52e−02	9.54e−03
1/32	1.78e−01	1.16e−03	1.92e−03	2.51e−01	3.27e−02	2.39e−03
1/64	8.90e−02	2.90e−04	4.81e−04	1.25e−01	1.63e−02	6.01e−04
1/128	4.45e−02	7.27e−05	1.20e−04	6.29e−02	8.18e−03	1.50e−04
$O(h^r), r =$	0.9993	1.9808	1.9999	1.0015	0.9968	1.9861

Next, we consider a mixed boundary condition:

$$\begin{cases} u = g^D & \text{on } \Gamma_D, \\ (\mathcal{A}\nabla u)\cdot \mathbf{n} + \alpha u = g^R & \text{on } \Gamma_R, \end{cases}$$

where g^D is the Dirichlet boundary data, g^R is the Robin type boundary data, $\alpha \geq 0$, and $\Gamma_D \cap \Gamma_R = \emptyset$, $\Gamma_D \cup \Gamma_R = \partial\Omega$. When $\alpha = 0$, the Robin type boundary condition becomes the Neumann type boundary condition.

For the mixed boundary condition, it is not hard to see that the weak formulation can be written as: find $u_h \in S_{g_h^D, \Gamma_D, h}$ such that for all $v_h \in S_{0,\Gamma_D,h}$,

$$\begin{aligned} (\mathcal{A}\nabla_d u_h, \nabla_d v_h) \ &+ \langle \alpha u_b, v_b\rangle_{\Gamma_R} + (\boldsymbol{\beta}\cdot\nabla_d u_h, v_0) + (\gamma u_0, v_0) \\ &= (f, v_0) + \langle g^R, v_b\rangle_{\Gamma_R}, \end{aligned}$$

where $\langle\cdot,\cdot\rangle_{\Gamma_R}$ denotes the L^2 inner-product on Γ_R. We tested a two-dimensional problem with $\mathcal{A}$ to be an identity matrix and $\Omega = (0,1)^2$ with a uniform triangular mesh. The exact solution is chosen to be $u = \sin(\pi y)e^{-x}$. This function satisfies

$$\nabla u \cdot \mathbf{n} + u = 0$$

on the boundary segment $x = 1$. We use the Dirichlet boundary condition on all other boundary segments. The element $(P_0(K), P_0(F), RT_0(K))$ is used in the discretization. For the Dirichlet boundary data, the L^2 projection is used to approximate the boundary data g_h^D. The results are reported in Table 3. It shows optimal rates of convergence in all norms for the weak Galerkin approximation with mixed boundary conditions.

Table 3 Case 1. Numerical results for a test problem with mixed boundary conditions, where a Robin type boundary condition is imposed on part of the boundary

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	1.55e−01	3.18e−03	1.14e−02	1.95e−01	4.51e−02	1.12e−02
1/16	7.87e−02	8.20e−04	2.90e−03	9.82e−02	2.25e−02	3.18e−03
1/32	3.94e−02	2.06e−04	7.29e−04	4.92e−02	1.12e−02	8.40e−04
1/64	1.97e−02	5.17e−05	1.82e−04	2.46e−02	5.64e−03	2.15e−04
1/128	9.87e−03	1.29e−05	4.56e−05	1.23e−02	2.82e−03	5.46e−05
$O(h^r), r =$	0.9958	1.9876	1.9926	0.9971	1.0001	1.9262

Table 4 Case 2. Numerical results for a test problem with degenerate diffusion $\mathcal{A}$ in the domain

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	5.61e−02	3.32e−03	6.60e−03	5.75e−02	5.48e−03	1.27e−02
1/16	4.03e−02	1.38e−03	2.81e−03	4.09e−02	2.59e−03	4.90e−03
1/32	2.95e−02	5.68e−04	1.16e−03	2.96e−02	1.23e−03	2.21e−03
1/64	2.15e−02	2.35e−04	4.83e−04	2.15e−02	5.97e−04	1.16e−03
1/128	1.55e−02	9.93e−05	2.02e−04	1.55e−02	2.91e−04	5.99e−04
$O(h^r)$, $r =$	0.4614	1.2687	1.2594	0.4697	1.0579	1.0912

4.2 Case 2: a model problem with degenerate diffusion

We consider a test problem where the diffusion coefficient $\mathcal{A}$ is singular at some points of the domain. Note that in this case, the usual mixed finite element method may not be applicable due to the degeneracy of the coefficient. But the primary variable based formulations, including the weak Galerkin method, can still be employed for a numerical approximation.

More precisely, we consider the following two-dimensional problem

$$\begin{aligned} -\nabla \cdot (xy\,\nabla u) &= f \qquad \text{in } \Omega, \\ u &= 0 \qquad \text{on } \partial\Omega, \end{aligned}$$

where $\Omega = (0, 1)^2$. Notice that the diffusion coefficient $\mathcal{A} = xy$ vanishes at the origin. We set the exact solution to be $u = x(1 - x)y(1 - y)$. The configuration for the finite element partitions is the same as in test Case 1. We tested the weak Galerkin method on this problem, and the results are presented in Table 4 and Fig. 1.

Since the diffusion coefficient $\mathcal{A}$ is not uniformly positive definite on Ω, we have no anticipation that the weak Galerkin approximation has any optimal rate of convergence, though the exact solution is smooth. It should be pointed out that the usual Lax-Milgram theorem is not applicable to such problems in order to have a result on the solution existence and uniqueness. However, one can prove that the discrete problem always has a unique solution when

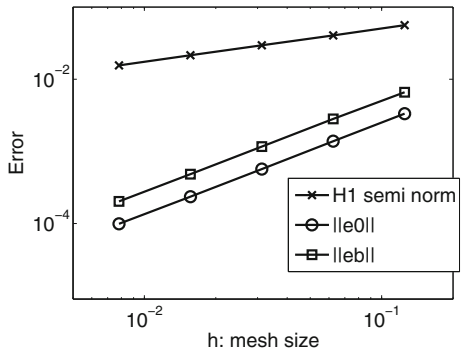

Fig. 1 Case 2. Convergence rate of $\|\nabla_d e_h\|$, $\|e_0\|$ and $\|e_b\|$ for the case of degenerate diffusion

Table 5 Case 3. Convergence rates for a problem with corner singularity ($\gamma = 0.5$)

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	1.88e−01	6.40e−03	1.47e−02	2.54e−01	1.49e−02	4.30e−02
1/16	1.36e−01	2.20e−03	5.28e−03	1.84e−01	7.66e−03	3.01e−02
1/32	9.74e−02	7.62e−04	1.86e−03	1.32e−01	3.89e−03	2.12e−02
1/64	6.93e−02	2.65e−04	6.57e−04	9.42e−02	1.96e−03	1.49e−02
1/128	4.92e−02	9.33e−05	2.32e−04	6.69e−02	9.88e−04	1.05e−02
$O(h^r)$, $r =$	0.4852	1.5251	1.4992	0.4827	0.9805	0.5066

Gaussian quadratures are used in the numerical integration. Interestingly, the numerical experiments show that the weak Galerkin method converges with a rate of approximately $O(h^{0.5})$ in $\|\nabla_d e_h\|$, $O(h^{1.25})$ in $\|e_0\|$ and $\|e_b\|$. It is left for future research to explore a theoretical foundation of the observed convergence behavior.

4.3 Case 3: a model problem on a domain with corner singularity

We consider the Laplace equation on a two-dimensional domain for which the exact solution possesses a corner singularity. For simplicity, we take $\Omega = (0, 1)^2$ and let the exact solution be given by

$$u(x, y) = x(1 - x)y(1 - y)r^{-2+\gamma}, \tag{4.2}$$

where $r = \sqrt{x^2 + y^2}$ and $\gamma \in (0, 1]$ is a constant. Clearly, we have

$$u \in H_0^1(\Omega) \cap H^{1+\gamma-\varepsilon}(\Omega) \quad \text{and} \quad u \notin H^{1+\gamma}(\Omega),$$

where ε is any small, but positive number. Again, a uniform triangular mesh and the element $(P_0(K),\ P_0(F),\ RT_0(K))$ are used in the numerical discretization. Note that the weak Galerkin for this problem is exactly the same as the standard mixed finite element method.

This model problem was numerically tested with $\gamma = 0.5$ and $\gamma = 0.25$. The convergence rates are reported in Tables 5 and 6. Notice that $\|\nabla_d e_h\|$ and $\|e_0\|$ behaves in a way as predicted by theory (2.4); i.e., they converge with rates given by $O(h^\gamma)$ and $O(h^{1+\gamma})$, respectively. The result also shows that the approximation on the element edge/face has a rate of convergence $O(h^{1+\gamma})$.

Table 6 Case 3. Convergence rates for a problem with corner singularity ($\gamma = 0.25$)

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	4.93e−01	1.69e−02	3.58e−02	6.65e−01	2.56e−02	1.25e−01
1/16	4.18e−01	7.07e−03	1.52e−02	5.66e−01	1.31e−02	1.05e−01
1/32	3.53e−01	2.94e−03	6.39e−03	4.79e−01	6.72e−03	8.85e−02
1/64	2.98e−01	1.22e−03	2.68e−03	4.04e−01	3.42e−03	7.44e−02
1/128	2.51e−01	5.14e−04	1.12e−03	3.40e−01	1.73e−03	6.25e−02
$O(h^r)$, $r =$	0.2437	1.2613	1.2489	0.2417	0.9717	0.2505

4.4 Case 4: a model problem with intersecting interfaces

This test problem is taken from [9], which has also been tested by other researchers [11]. In two dimension, consider $\Omega = (-1, 1)^2$ and the following problem

$$-\nabla \cdot (\mathcal{A}\nabla u) = 0,$$

where $\mathcal{A} = K_1\mathbf{I}_2$ in the first and third quadrants, and $K_2\mathbf{I}_2$ in the second and forth quadrants. Here $\mathbf{I}_2$ is the 2×2 identity matrix and K_1, K_2 are two positive numbers. Consider an exact solution which takes the following form in polar coordinates:

$$u(x, y) = r^\gamma \mu(\theta),$$

where $\gamma \in (0, 1]$ and

$$\mu(\theta) = \begin{cases} \cos((\pi/2 - \sigma)\gamma)\cos((\theta - \pi/2 + \rho)\gamma), & \text{if } 0 \le \theta \le \pi/2, \\ \cos(\rho\gamma)\cos((\theta - \pi + \sigma)\gamma), & \text{if } \pi/2 \le \theta \le \pi, \\ \cos(\sigma\gamma)\cos((\theta - \pi - \rho)\gamma), & \text{if } \pi \le \theta \le 3\pi/2, \\ \cos((\pi/2 - \rho)\gamma)\cos((\theta - 3\pi/2 - \sigma)\gamma), & \text{if } 3\pi/2 \le \theta \le 2\pi. \end{cases} \tag{4.3}$$

The parameters γ, ρ, σ satisfy the following nonlinear relations

$$\begin{aligned} &R := K_1/K_2 = -\tan((\pi/2 - \sigma)\gamma)\cot(\rho\gamma), \\ &1/R = -\tan(\rho\gamma)\cot(\sigma\rho), \\ &R = -\tan(\rho\gamma)\cot((\pi/2 - \rho)\gamma), \\ &\max\{0, \pi\gamma - \pi\} < 2\gamma\rho < \min\{\pi\gamma, \pi\}, \\ &\max\{0, \pi - \pi\gamma\} < -2\gamma\sigma < \min\{\pi, 2\pi - \pi\gamma\}. \end{aligned} \tag{4.4}$$

The solution $u(r, \theta)$ is known to be in $H^{1+\gamma-\varepsilon}(\Omega)$ for any $\varepsilon > 0$, and has a singularity near the origin $(0, 0)$.

One choice for the coefficients is to take $\gamma = 0.1$, $R \approx 161.4476387975881$, $\rho \approx \pi/4$, $\sigma \approx -14.92256510455152$. We numerically solve this problem by

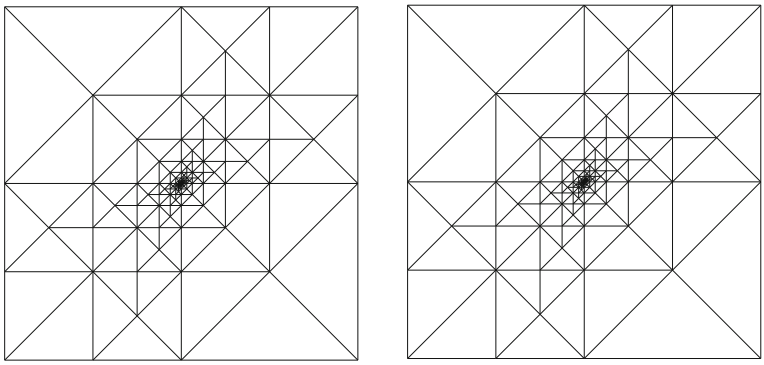

Fig. 2 Case 4. The initial triangular mesh for the intersecting interface problem, with 268 (*left*) and 300 (*right*) *triangles*

Table 7 Case 4. Convergence rate for the intersecting interface problem with an initial mesh containing 268 triangles

Level	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
0	1.07e−01	3.97e−03	9.95e−03	1.47e−01	2.60e−02	1.97e−02
1	9.76e−02	2.92e−03	6.44e−03	1.26e−01	1.33e−02	1.94e−02
2	9.30e−02	2.51e−03	5.11e−03	1.16e−01	7.01e−03	1.91e−02
3	9.12e−02	2.21e−03	4.44e−03	1.11e−01	3.95e−03	1.88e−02
4	8.98e−02	1.95e−03	3.91e−03	1.07e−01	2.55e−03	1.84e−02
$O(h^r), r =$	0.0604	0.2446	0.3229	0.1084	0.8461	0.0239

using the weak Galerkin method with element $(P_0(K),\ P_0(F),\ RT_0(K))$ on triangular meshes. It turns out that uniform triangular meshes are not good enough to handle the singularity in this problem. Indeed, we use a locally refined initial mesh, as shown in Fig. 2, which consists of 268 triangles. This mesh is then uniformly refined, by dividing each triangle into 4 subtriangles, to get a sequence of nested meshes. Although this can not be compared with an adaptive mesh refinement process, it does improve the accuracy of the numerical approximation, as shown in our numerical results reported in Table 7. Since the mesh is not quasi-uniform, we do not expect that the theoretical error estimation (2.4) apply for this problem. An interesting observation of Table 7 is that, the norm $\|u_0 - u\|$ appears to converge in a much faster rate than $\|e_0\| = \|u_0 - Q_0 u\|$, while the opposite has usually been observed for other test cases. We believe that this is due to the use of a locally refined initial mesh in our testing process. When the actual value of $\|u_0 - u\|$ reduces to the same level as the value of $\|e_0\|$, its convergence rate slows down to the same as $\|e_0\|$. Readers are also encouraged to derive their own conclusions from these numerical experiments.

We also observe that, when the initial mesh gets more refined near the origin, the convergence rates increase slightly. In Table 8, this trend is clearly shown. For each initial mesh, it is refined four times to get five levels of nested meshes. The convergence rates are computed based on these five nested

Table 8 Case 4. Convergence rate for the intersecting interface problem with different initial meshes, where the first column indicates the total number of triangles in the initial mesh

#	Convergence rates $O(h^r), r =$					
triangles	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
268	0.0604	0.2446	0.3229	0.1084	0.8461	0.0239
300	0.0750	0.2623	0.3489	0.1206	0.8699	0.0373
332	0.0888	0.2818	0.3772	0.1329	0.8912	0.0487
364	0.1020	0.3031	0.4079	0.1454	0.9099	0.0586
396	0.1148	0.3266	0.4411	0.1581	0.9260	0.0673
428	0.1273	0.3522	0.4766	0.1711	0.9396	0.0749
460	0.1396	0.3802	0.5145	0.1843	0.9509	0.0817
492	0.1519	0.4105	0.5548	0.1978	0.9602	0.0878
524	0.1641	0.4432	0.5972	0.2117	0.9678	0.0932

Table 9 Case 5. Convergence rate for the anisotropic problem with $k = 3$

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	1.48e+0	1.95e−02	4.61e−02	2.70e+0	1.29e−01	4.13e−02
1/16	7.39e−01	5.11e−03	1.16e−02	1.35e+0	6.53e−02	1.06e−02
1/32	3.69e−01	1.29e−03	2.92e−03	6.80e−01	3.27e−02	2.67e−03
1/64	1.84e−01	3.24e−04	7.33e−04	3.40e−01	1.63e−02	6.68e−04
1/128	9.23e−02	8.12e−05	1.83e−04	1.70e−01	8.18e−03	1.66e−04
$O(h^r), r =$	1.0010	1.9793	1.9942	0.9972	0.9975	1.9906

meshes. The initial meshes are generated by refining only those triangles near the origin. Two examples of initial meshes are shown in Fig. 2.

4.5 Case 5: an anisotropic problem

Consider a two dimensional anisotropic problem defined in the square domain $\Omega = (0, 1)^2$ as follows

$$-\nabla \cdot (\mathcal{A}\nabla u) = f,$$

where the diffusion coefficient is given by

$$\mathcal{A} = \begin{bmatrix} k^2 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \text{for } k \neq 0.$$

We chose a function f and a Dirichlet boundary condition so that the exact solution is given by $u(x, y) = \sin(2\pi x)\sin(2k\pi y)$. In applying the weak Galerkin method, we use an anisotropic triangular mesh that was constructed by first dividing the domain into $kn \times n$ sub-rectangles, and then splitting each rectangle into two triangles by connecting a diagonal line. The characteristic mesh size is $h = 1/n$. We tested two cases with $k = 3$ and $k = 9$. The results are reported in Tables 9 and 10. The tables show optimal rates of convergence for the weak Galerkin approximation in various norms. The numerical experiment indicates that the weak Galerkin method can handle anisotropic problems and meshes without any trouble.

Table 10 Case 5. Convergence rate for the anisotropic problem with $k = 9$

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/4	7.98e+0	6.80e−02	2.93e−01	1.58e+1	2.52e−01	1.49e−01
1/8	3.89e+0	2.07e−02	7.44e−02	8.18e+0	1.30e−01	4.22e−02
1/16	1.91e+0	5.43e−03	1.88e−02	4.12e+0	6.53e−02	1.09e−02
1/32	9.54e−01	1.37e−03	4.72e−03	2.06e+0	3.27e−02	2.74e−03
1/64	4.76e−01	3.44e−04	1.18e−03	1.03e+0	1.63e−02	6.84e−04
$O(h^r), r =$	1.0161	1.9160	1.9897	0.9857	0.9883	1.9492

Table 11 Case 6. Convergence rate for a 3D model problem with smooth solution

h	$\|\nabla_d e_h\|$	$\|e_0\|$	$\|e_b\|$	$\|\nabla_d u_h - \nabla u\|$	$\|u_0 - u\|$	$\|e_0\|_\infty$
1/8	1.85e−01	1.62e−02	4.27e−02	1.22e+00	1.34e−01	3.63e−02
1/12	8.53e−02	7.69e−03	1.94e−02	8.19e−01	9.14e−02	1.96e−02
1/16	4.86e−02	4.42e−03	1.10e−02	6.15e−01	6.89e−02	1.18e−02
1/20	3.13e−02	2.85e−03	7.07e−03	4.92e−01	5.52e−02	7.78e−03
$O(h^r)$, $r =$	1.9389	1.8984	1.9618	0.9914	0.9737	1.6779

4.6 Case 6: a three-dimensional model problem

The final test problem is a three dimensional Laplace equation defined on $\Omega = (0, 1)^3$, with a Dirichlet boundary condition and an exact solution given by $u = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z)$. The purpose of this test problem is to examine the convergence rate of the cubic $(Q_0(K),\ Q_0(F),\ RT_0(K))$ element. The results are reported in Table 11.

In addition to the optimal rates of convergence as shown in Table 11, on can also see a superconvergence for $\|\nabla_d e_h\|$. The same result is anticipated for 2D rectangular elements. It is left to interested readers for a further investigation, especially for model problems with variable coefficients.

References

1. Arnold, D.N., Brezzi, F.: Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates. RAIRO Model. Math. Anal. Numer. **19**, 7–32 (1985)
2. Arnold, D., Brezzi, F., Cockburn, B., Marini, D.: Unified analysis of discontinuous Galerkin methods for elliptic problems. SIAM J. Numer. Anal. **39**, 1749–1779 (2002)
3. Bahriawati, C., Carstensen, C.: Three Matlab implementations of the lowest order Raviart–Thomas MFEM with a posteriori error control. Comput. Methods Appl. Math. **5**, 333–361 (2005)
4. Brezzi, F., Douglas, J., Marini, L.: Two families of mixed finite elements for second order elliptic problems. Numer. Math. **47**, 217–235 (1985)
5. Brezzi, F., Fortin, M.: Mixed and Hybrid Finite Element Methods. Springer (1991)
6. Demkowicz, L., Gopalakrishnan, J.: A class of discontinuous Petrov–Galerkin methods. Part I: the transport equation. Comput. Methods Appl. Mech. Eng. **23–24**, 1558–1572 (2010)
7. Demkowicz, L., Gopalakrishnan, J.: A class of discontinuous Petrov–Galerkin methods. Part II: optimal test functions. Numer. Methods Parts D. E. **27**, 70–105 (2011)
8. Demkowicz, L., Gopalakrishnan, J.: Analysis of the DPG method for the Poisson problem. SIAM J. Numer. Anal. **49**, 1788–1809 (2011)
9. Kellog, R.B.: On the Poisson equation with intersecting interfaces. Appl. Anal., **4**, 101–129 (1976)
10. Lin, M., Wang, J., Ye, X.: Weak Galerkin finite element methods for second-order elliptic problems on polytopal meshes. arXiv:1204.3655v1 [math.NA]
11. Morin, P., Nochetto, R.H., Siebert, K.G.: Convergence of adaptive finite element methods. SIAM Rev. **44**, 631–658 (2002)

12. Raviart, P., Thomas, J.: A mixed finite element method for second order elliptic problems. In: Galligani, I., Magenes, E. (eds.) Mathematical Aspects of the Finite Element Method, Lectures Notes in Math. 606. Springer, New York (1977)
13. Wang, J., Ye, X.: A weak Galerkin finite element method for second-order elliptic problems. arXiv:1104.2897v1 [math.NA] (2011)
14. Wang, J., Ye, X.: A weak Galerkin mixed finite element method for second-order elliptic problems. arXiv:1104.2897v1 [math.NA]