Computers and Mathematics with Applications 71 (2016) 1568–1585

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Mortar formulation for a class of staggered discontinuous Galerkin methods

Hyea Hyun Kim [a,*], Eric T. Chung [b], Chi Yeung Lam [b]

[a] Department of Applied Mathematics, Kyung Hee University, Republic of Korea

[b] Department of Mathematics, The Chinese University of Hong Kong, Hong Kong Special Administrative Region

ARTICLE INFO

Article history:
Received 3 August 2015
Received in revised form 31 December 2015
Accepted 21 February 2016
Available online 19 March 2016

Keywords:
Mortar condition
Elliptic problems
Discontinuous Galerkin methods
Non-matching meshes
Staggered grids

ABSTRACT

A mortar formulation is developed and analyzed for a class of staggered discontinuous Galerkin (SDG) methods applied to second order elliptic problems in two dimensions. The computational domain consists of nonoverlapping subdomains and a triangulation is provided for each subdomain, which need not conform across subdomain interfaces. This feature allows a more flexible design of discrete models for problems with complicated geometries, shocks, or singular points. A mortar matching condition is enforced on the solutions across the subdomain interfaces by introducing a Lagrange multiplier space. Moreover, optimal convergence rates in both L^2 and discrete energy norms are proved. Numerical results are presented to show the performance of the method.

1. Introduction

The staggered discontinuous Galerkin (SDG) method was initially developed for wave propagation problems [1,2] giving a class of explicit, energy conserving and optimally convergent schemes, with significantly smaller dispersion errors [3,4]. The main idea of the method is the use of a pair of locally conforming finite element spaces, which are defined on a staggered grid and are constructed so that some inf–sup stability conditions are obtained. In addition, in contrast to classical DG schemes [5,6], SDG methods do not require the use of any numerical flux, so that the numerical solutions will satisfy some discrete physical laws [4,7,2,8] mimicking the continuous counterpart. The methodology is then designed for other types of problems, for example convection–diffusion equations [7], Stokes equations [9], and the Maxwell's equations [4,8,10] with good success. When applied to second order elliptic problems in the mixed formulation [11–13], the method gives optimal convergence for both the original scalar variable and the flux variable. Moreover, a local postprocessing technique can then be applied to obtain a superconvergent solution [14]. On the other hand, fast solvers have been developed for efficient iterative solutions. In particular, some overlapping Schwarz methods [11] and a BDDC (Balancing Domain Decomposition by Constraints) method [12] are developed for robust preconditioning. Optimal condition number bounds are also obtained.

In this paper, we consider a mortar formulation of the SDG method applied to second order elliptic problems in a mixed form. For many practical applications, it is desirable to use different triangulations in different regions of the computational domain. To allow flexibility in constructing the triangulations, the triangulations are in general non-matching across different regions. We will then need a way to couple the solutions in various regions. The difficulty in our formulation is that we have to deal with both nonmatching meshes and discontinuous vector flux variables across the subdomain interfaces compared to mortar formulation of the standard conforming case, see [15–17]. The key methodologies of this

* Corresponding author.
E-mail addresses: hhkim@hku.ac.kr, hyeahyun@gmail.com (H.H. Kim), tschung@math.cuhk.edu.hk (E.T. Chung), cylam@math.cuhk.edu.hk (C.Y. Lam).

http://dx.doi.org/10.1016/j.camwa.2016.02.035
0898-1221/

work are to use the mortar idea for coupling the nonmatching meshes as in the standard mortar methods [15–17] and to introduce interface flux variables by hybridization as in [14,18]. We refer to earlier works in [19,20] for mixed methods with hybridization. The interface flux variables are in fact related to Lagrange multipliers appearing in the mortar matching condition on scalar variables. In each subdomain, the discretization is obtained by the standard SDG method presented in [12,11]. Across subdomain interfaces, these local solutions are coupled through the use of Lagrange multipliers. We refer [21] for a mortar formulation to couple mixed finite element methods on nonmatching meshes where higher order polynomials are introduced to form the Lagrange multiplier space so as to obtain the optimal order of errors for both velocity and pressure approximations. Later, in [22] a similar idea is applied to slightly compressible flows in porous media. We refer to [23] for a different approach, where Nitsche's method is applied to deal with the non-matching meshes on an interface. Coupling of discontinuous Galerkin and mixed finite element methods was also developed in [24] by introducing a carefully designed flux condition. In that approach, the optimal order of errors for the velocity approximation is obtained while one less order is obtained for the pressure approximation.

For the proposed mortar SDG method, optimal convergence theory is developed. We have shown that the order of convergence is k in a discrete energy norm for the scalar variables and is $k+\frac{1}{2}$ in L^2-norm for the vector flux variables, when the scalar and vector flux variables, and Lagrange multipliers are approximated by polynomials of the same order up to $k \geq 0$. In fact, we have shown numerically that the order $k+\frac{1}{2}$ for the vector flux variable is indeed sharp. In our analysis, by using weighted norms depending on coefficients ρ_i of the subdomains, and certain assumptions on meshes and the choice of non-mortar/mortar subdomains we can obtain the error bounds to be independent of coefficient variations. For matching meshes, we prove that we recover the optimal convergence for the flux variable, namely, the convergence order is $k+1$ in L^2-norm. Furthermore, by a duality technique, we can prove that the convergence order is $k+1$ for the scalar variable in L^2-norm. All of these theoretical estimates are confirmed by numerical experiments.

The paper is organized as follows. In Section 2, we will introduce some notations and give a detailed construction of the mortar SDG method. The optimal convergence of the method is analyzed in Section 3. In Section 4, we present some numerical examples to show the performance of the method and confirm the theoretical estimates.

2. Mortar formulation of SDG method

In this section, we will describe the mortar formulation for a class of SDG methods. We will first present the subdomain partition and triangulations of subdomains as well as mesh notations. We emphasize that the triangulations of subdomains are not necessarily matching across subdomain interfaces. We will then derive the mortar SDG methods. The main idea is that, in each subdomain, the discretization is obtained by the classical SDG methods (see for example [11,12]) with a Neumann boundary condition, which is taken as an additional unknown corresponding to the flux and having a single value on subdomain interfaces. This additional unknown allows us to impose continuity of the solution across subdomain interfaces by a mortar matching condition, to be specified in the following sections. The resulting system is a saddle point problem for the original solution and the additional flux variable.

2.1. Model problem and triangulation

For a concrete illustration of the mortar formulation of SDG methods, we will consider the following second order elliptic problem in two dimensions:

$$\begin{aligned} -\nabla\cdot(\rho\,\nabla u) &= f, \quad \text{in } \Omega,\\ u &= 0, \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where Ω is the computational domain and $f(x)$ is a given source function. We divide the domain Ω into a set of N non-overlapping subdomains, $\overline{\Omega} = \bigcup_{i=1}^{N} \overline{\Omega}_i$. We assume, for simplicity, that $\{\Omega_i\}_{i=1}^{N}$ is a geometrically conforming partition of Ω but the algorithm and theory developed in this paper can be extended to geometrically nonconforming partitions. We further assume that $\rho(x)$ is piecewise constant and $\rho(x) = \rho_i > 0$ in each subdomain Ω_i. Every subdomain Ω_i is equipped with a quasi-uniform triangulation $\mathcal{T}_{h_i}$ with mesh size $h_i > 0$. The triangulations $\{\mathcal{T}_{h_i}\}_{i=1}^{N}$ can be non-matching across the subdomain interface $\Gamma = \bigcup \Gamma_{ij}$, where $\Gamma_{ij} (= \partial\Omega_i \bigcap \partial\Omega_j)$ is the interface shared by the two subdomains Ω_i and Ω_j. See Fig. 1 for an illustration of two subdomains.

As explained above, the discretization of (2.1) in each subdomain is based on the classical SDG methods. Thus, we will now present the construction of the SDG space for each Ω_i, and the construction follows the framework presented in [1,2]. We let $\mathcal{F}_{u,i}$ be the set of edges in the initial triangulation $\mathcal{T}_{h_i}$ and $\mathcal{F}_{u,i}^0 \subset \mathcal{F}_{u,i}$ be the set of interior edges. For each triangle $\tau \in \mathcal{T}_{h_i}$, we divide it into three subtriangles by connecting an interior point to the three vertices. We note that the interior point can be chosen as the centroid of the triangle to get a good regularity of the subdivided triangulation.

We denote by $\mathcal{S}\mathcal{T}_{h_i}$ the resulting finer triangulation and by $\mathcal{F}_{q,i}$ the set of edges generated by this subdivision process. Fig. 1 shows an illustration of this process with solid lines representing edges in $\mathcal{F}_{u,i}$ and dashed lines representing edges in $\mathcal{F}_{q,i}$. Let $k \geq 0$ be the order of polynomial used for the approximation and $P^k(\tau)$ be the set of polynomials with degree less than or equal to k defined on τ. We define the following SDG spaces

$$Q_{h_i} = \left\{ \boldsymbol{q} : \boldsymbol{q}|_\tau \in [P^k(\tau)]^2,\ \forall \tau \in \mathcal{S}\mathcal{T}_{h_i} \text{ and } [\boldsymbol{q}]|_e = 0,\ \forall e \in \mathcal{F}_{q,i} \right\} \tag{2.2}$$