Applied Mathematics and Computation 266 (2015) 838–849

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Scale separation in fast hierarchical solvers for discontinuous Galerkin methods

Vadym Aizinger[a,*], Dmitri Kuzmin[b], Lukas Korous[c]

[a] Applied Mathematics I, Friedrich-Alexander University Erlangen-Nürnberg, Cauerstr. 11, 91058 Erlangen, Germany
[b] Department of Mathematics, LS III, Dortmund University of Technology, Vogelpothsweg 87, D-44227 Dortmund, Germany
[c] Department of Theory of Electrical Engineering, Faculty of Electrical Engineering, University of West Bohemia, Univerzitni 26, 306 14 Pilsen, Czech Republic

ARTICLE INFO

Keywords:
Hyperbolic conservation laws
Convection–diffusion equation
Discontinuous Galerkin methods
Nonsymmetric interior penalty Galerkin method
p-multigrid method
Multiscale methods

ABSTRACT

We present a method for solution of linear systems resulting from discontinuous Galerkin (DG) approximations. The two-level algorithm is based on a hierarchical scale separation scheme (HSS) such that the linear system is solved globally only for the cell mean values which represent the coarse scales of the DG solution. The system matrix of this coarse-scale problem is exactly the same as in the cell-centered finite volume method. The higher order components of the solution (fine scales) are computed as corrections by solving small local problems. This technique is particularly efficient for DG schemes that employ hierarchical bases and leads to an unconditionally stable method for stationary and time-dependent hyperbolic and parabolic problems. Unlike *p*-multigrid schemes, only two levels are used for DG approximations of any order. The proposed method is conceptually simple and easy to implement. It compares favorably to *p*-multigrid in our numerical experiments. Numerical tests confirm the accuracy and robustness of the proposed algorithm.

1. Introduction

Discontinuous Galerkin (DG) methods [5] have been an active research area during the last decade and belong nowadays to the most widely used numerical tools in CFD. They combine some of the most attractive features of the finite element and finite volume methods, support meshes with very general element shapes and hanging nodes, and are well suited for all types of adaptivity. Compact approximation stencils and favorable computation to communication ratios of DG schemes enable efficient parallelization of codes on distributed memory clusters and GPUs. Their main drawback compared to the finite element/finite volume methods still remains a significantly larger number of degrees of freedom utilized by a DG approximation of the same order. This disadvantage becomes even more limiting if instead of explicit time-stepping algorithms one considers stationary and time-implicit DG discretizations that require solution of linear equation systems.

A promising approach to solving linear systems originating from DG discretizations was proposed by Fidkowski et al. [7] and further developed and analyzed by a number of other authors in [2,3,8,11–13,15]. All numerical methods investigated in the above publications are variations of the *p*-multigrid scheme motivated by a similar solution technique proposed for the spectral method by Ronquist and Patera in [16]. As opposed to the classical *h*-multigrid method that relies on coarse meshes to suppress the low wave number errors present in the fine-mesh solution, the *p*-multigrid method employs in this role lower order approximations

* Corresponding author. Tel.: +49 9131 85 67222.
E-mail addresses: aizinger@math.fau.de (V. Aizinger), kuzmin@math.uni-dortmund.de (D. Kuzmin), korous@rice.zcu.cz (L. Korous).

http://dx.doi.org/10.1016/j.amc.2015.05.047
0096-3003/

of the solution on the same mesh. Particularly efficient implementations of this scheme can be produced for DG discretizations utilizing hierarchical bases.

The hierarchical scale separation (HSS) technique explored in this work differs from the p-multigrid method in two key aspects: first, instead of using a full hierarchy of approximations as smoothers we only need two approximations for a DG scheme of any order: coarse and fine; second, our fine-scale smoothing does not affect any coarse degrees of freedom resulting in a more complete decoupling of the coarse- and fine-scale problems. The coarse-scale problem is formulated for cell mean values; it has the computational structure of a cell-centered finite volume method. The local fine-scale problems are solved using the fluxes that contain the results of the global coarse-scale solve. In contrast to the full p-multigrid algorithm, all fine-scale components associated with the same element are updated simultaneously. Whereas two-level simplifications of p-multigrid schemes are – however uncommon – treated in the DG literature [12], the second difference appears to be a unique feature of the HSS method presented here.

In [10], we introduced an iterative solver based on the HSS paradigm in the context of an implicit pseudo-time-stepping DG scheme for stationary hyperbolic conservation laws. The unconditionally stable implicit algorithm has a lot in common with variational multiscale (VMS) methods. As shown by Bochev et al. [4,9], the multiscale nature of DG approximations provides useful insights for the design of efficient algorithms. In particular, an additive decomposition of the finite element space into continuous coarse scales and discontinuous fine scales leads to a DG scheme with the computational structure of a continuous Galerkin method [4]. In the present paper, we extend HSS to DG discretizations of (both stationary and time-dependent) convection–diffusion equations. We also describe the traditional p-multigrid approach and discuss its relationship to the HSS-based solver.

The remainder of this paper is organized as follows. In the next section, we formulate the boundary value problem for a generic convection–diffusion equation and discretize it using a DG method. Section 3 details the hierarchical scale decomposition procedure and gives a brief outline of the p-multigrid scheme that was used as a comparison. Some numerical examples illustrating the performance of the proposed method for stationary and time-dependent problems are presented in Section 4.

2. DG discretization

2.1. Model problem

The model problem that we will use for the presentation of our scheme is the linear convection–diffusion equation

$$\frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{a}u - D\nabla u) = 0 \qquad \text{in } \Omega, \tag{1}$$

where Ω is a bounded domain, $u(\mathbf{x}, t)$ is a conserved scalar quantity, $\mathbf{a}(\mathbf{x}, t)$ is a (continuous) velocity field, and $D(\mathbf{x}, t)$ a diffusivity coefficient. The initial condition is given by

$$u(\cdot, 0) = u_0 \qquad \text{in } \Omega. \tag{2}$$

The inflow boundary is defined as $\partial\Omega_{\text{in}} = \{\mathbf{x} \in \partial\Omega \mid \mathbf{a}(\mathbf{x}, t) \cdot \mathbf{n}(\mathbf{x}) < 0\}$, where $\mathbf{n}$ denotes the unit outward normal to the boundary $\partial\Omega$. Similarly, we call $\partial\Omega_{\text{out}} = \{\mathbf{x} \in \partial\Omega \mid \mathbf{a}(\mathbf{x}, t) \cdot \mathbf{n}(\mathbf{x}) \geq 0\}$ the outflow boundary. The Dirichlet boundary condition

$$u = u_{\text{in}} \text{ on } \partial\Omega_{\text{in}} \tag{3}$$

is imposed for pure convection ($D = 0$) and convection–diffusion problems. If $D > 0$ we enforce, in addition,

$$u = u_{\text{out}} \text{ on } \partial\Omega_{\text{out}}. \tag{4}$$

In the present work, we consider several limiting cases of the above equation: pure convection, convection-dominated, and diffusion-dominated. In addition, the stationary problem ($\frac{\partial u}{\partial t} = 0$) was tested for all three settings.

2.2. Notation

We call $\mathcal{T}_h$ a (possibly unstructured) computational mesh and $\mathcal{S}_h$ will denote the set of all internal edges/faces in $\mathcal{T}_h$. On the interface between elements $K^+ \in \mathcal{T}_h$ and $K^- \in \mathcal{T}_h$, the traces are given by the one-sided limits

$$u^{\pm}(\mathbf{x}, t) = \lim_{\epsilon \to +0} u(\mathbf{x} - \epsilon\mathbf{n}^{\pm}, t), \quad \mathbf{x} \in \partial K,$$

where $\mathbf{n}^{\pm}$ are the external unit normals to $K^{\pm}$. The mean and the jump of a scalar u across the common boundary of K^+ and K^- are denoted by

$$\{u(\mathbf{x}, t)\} = \frac{1}{2}(u^+(\mathbf{x}, t) + u^-(\mathbf{x}, t))$$

and

$$[\![u(\mathbf{x}, t)]\!] = u^+(\mathbf{x}, t)\ \mathbf{n}^+(\mathbf{x}) + u^-(\mathbf{x}, t)\ \mathbf{n}^-(\mathbf{x})$$