Procedia Computer Science

Procedia Computer Science 101, 2016, Pages 8 – 17

YSC 2016. 5th International Young Scientist Conference on Computational Science

Fast parallel integration for three dimensional Discontinuous Petrov Galerkin method

Maciej Woźniak[1], Marcin Łoś[1], Maciej Paszyński[1], and Leszek Demkowicz[2]

[1] AGH University of Science and Technology Kraków, Poland
macwozni@agh.edu.pl, los@agh.edu.pl paszynsk@agh.edu.pl
[2] The University of Texas at Austin, Austin, Texas, U.S.A.
leszek@ices.utexas.edu

Abstract
Finite Element Method comes with a challenge of constructing test functions, that provide better stability. Discontinuous Petrov-Galerkin method constructs optimal test functions "on the fly". However this method comes with relatively high computational cost. In this paper we show a parallelization method to reduce computation time.

Keywords: Finite Element Method, Discontinuous Petrov-Galerkin, parallel, shared memory

1 Introduction

In this paper we present a parallelization of the algorithm for generation of the element matrices for Discontinuous Petrov Galerkin (DPG) method [3, 4, 5]. The DPG method is a new rapidly growing method for solving the numerical problems. It enables for automatic control of the stability of the numerical formulations. We have parallelized the element routines of the hp3d DPG code developed by the group of prof. Demkowicz. We are aware of other parallel FEM packages supporting adaptive computations for DPG, including CAMELIA [14] and DUNE-DPG [10]. However, the hp3d framework is unique in the following ways:

- It supports hexahedral, tetrahedral, prism and pyramid elements in 3D. To our best knowledge, CAMELIA supports only hexahedral elements, and DUNE supports triangular elements only.
- It enables for parallel anisotropic refinements over computational domain distributed with different kind of finite elements, including tetrahedral, hexahedra, prism and pyramid. The CAMELIA and DUNE packages do not allow for anisotropic refinements, and thus the exponential convergence of the numerical solution is not possible there.
- CAMELIA and DUNE do not support complex Hcurl discretization. Our framework will enable for parallel automatic hp-adaptive computations for different classes of problems, including H1, Hdiv, and Hcurl.

Peer-review under responsibility of organizing committee of the scientific committee of the 5th International Young Scientist Conference on Computational Science

doi:10.1016/j.procs.2016.11.003

Our preliminary work presented in this paper concerns the parallelization of the element matrices for the elliptic problem. However, our future work will involve parallelization of Hdiv and Hcurl element routines.

Let us focus on the model elliptic problem. In the Sobolev space

$$H_0^1(\Omega) = \{u \in L^2(\Omega) : D^\alpha u \in L^2\Omega,\ |\alpha| \leq 1,\ \mathrm{tr}\, u = 0 \text{ on } \partial\Omega\} \tag{1}$$

we introduce classical weak formulation for Poisson problem in $H_0^1(\Omega)$. We seek for $u \in H_0^1(\Omega)$.

$$\int_\Omega \nabla u \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \quad \forall v \in H_0^1(\Omega) \tag{2}$$

We may also express the above problem with abstract notation:

$$b(u,v) = l(v) \tag{3}$$

where in our model problem we have

$$b(u,v) = \int_\Omega \nabla u \nabla v \,\mathrm{d}x \tag{4}$$

$$l(v) = \int_\Omega f v \,\mathrm{d}x \tag{5}$$

We project the weak problem into the finite dimensional subspace $V_h \subset H_0^1(\Omega)$

$$\int_\Omega \nabla u_h \nabla v_h \,\mathrm{d}x = \int_\Omega f v_h \,\mathrm{d}x \quad \forall v_h \in V_h \subset H_0^1(\Omega) \tag{6}$$

The actual mathematical theory concerning the stability of the numerical method for general weak formulation (2) is based on the famous "Babuśka-Brezzi condition" (BBC) developed in years 1971-1974 at the same time by Ivo Babuśka, and Franco Brezzi [13, 12, 9] The condition states that the problem (2) is stable when

$$\sup_{v \in V} \frac{|b(u,v)|}{\|v\|_V} \geq \gamma \|u\|_U \forall u \in U \tag{7}$$

However, the inf-sup condition in the above form concerns the abstract formulation where we consider all the test functions from $v \in V$ and look for solution at $u \in U$ (e.g. $U = V$). The above condition is satisfied also if we restric to the space of trial functions $u_h \in U_h$

$$\sup_{v \in V} \frac{|b(u_h,v)|}{\|v\|_V} \geq \gamma \|u_h\|_{U_h} \tag{8}$$

However, if we use test functions from the finite dimensional test space $V_h = \mathrm{span}\{v_h\}$

$$\sup_{v_h \in V_h} \frac{|b(u_h,v_h)|}{\|v_h\|_{V_h}} \geq \gamma_h \|u_h\|_{U_h} \tag{9}$$

we do not have a guarantee that the supremum (9) will be equal to the original supremom (7), since we have restricted V to V_h. The optimality of the method depends on the quality