Hybridized Discontinuous Galerkin (HDG) Methods

Hybridized discontinuous Galerkin (HDG) methods can be considered as extensions to discontinuous Galerkin (DG) methods. Basic idea, in layman's terms, is to introduce trace functions which only live on the faces of elements and define fluxes using these new functions. We will consider interior penalty DG (IPDG) method for both elementwise unknowns and trace unknowns and we will call it IPDG-Hybridized (IPDG-H). IPDG-H has some desirable properties; resulting linear system is smaller than those from DG methods and it may even be smaller than linear systems resulting from continuous Galerkin (CG) discretizations if high order polynomials are needed. They are stable for advection dominated flows. Another good quality of IPDG-H is that it results in approximate velocity fields which are exactly divergence free which is desired in the solution of incompressible flow problems. Note that it is possible to use a different space for facet elements, for example if CG methods are used, the resulting method is called embedded DG (EDG) method and has the stability of DG methods and is spectrally equivalent to CG methods. But they do not give exactly divergence free velocity fields since divergence of a function in a DG space is not contained in a CG space in general.

Program:

- **Part I: Introduction to Hybridized Discontinuous Galerkin (HDG) Methods**
  December 21, 15:40-17:30, S212 IAM

In the first part, we will discuss the derivation of HDG methods, leaving the theory out (coercivity, continuity, interpolation error estimates) for interested to read, introduce a notation in the preparation for MATLAB implementation. Through this notation, we believe we are addressing the challenge of vectorization, which is crucial for writing fast MATLAB codes.

- **Part II: How to Implement Hybridized Discontinuous Galerkin (HDG) Methods Efficiently in MATLAB?**
  December 22, 14:40-16:30, S212 IAM

In second part, we will go through the MATLAB implementation, show the link between Mathematics and the code.

Note that, MATLAB code is not meant to be used for high fidelity simulations as MATLAB is quite inefficient in memory usage and quite limited in parallel (both shared and distributed) scalability. But still this implementation and MATLAB itself are critical tools in Applied Mathematics learning and prototyping. Hence if anyone is interested, we can direct them to software packages build upon C, C++ or Fortran which still retains high level of abstraction while delivering high speed and efficient memory usage.

Suggested Reading:

- B. Cockburn, J. Gopalakrishnan, and R.Lazarov, Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems, SIAM Journal on Numerical Analysis 47(2), 1319-1365, 2009.

- N.C. Nguyen, J. Peraire, B. Cockburn, An Implicit High-Order Hybridizable Discontinuous Galerkin Method for Linear Convection-Diffusion Equations, Journal of Computational Physics 228(9), 3232-3254, 2009.

Speaker:

Abdullah Ali Sivas

Research and Teaching Assistant, University of Waterloo, Canada

Email: aasivas@uwaterloo.ca

Web page: http://www.math.uwaterloo.ca/~aasivas