Homogeneous Multigrid for the Hybridizable Discontinuous Galekrin Method

Final Project for 16.930 – Spring 2023

Aditya Karthik Saravanakumar

Linear Systems

$$\underline{\underline{A}}\,\underline{x} = \underline{b}$$

- Linear systems of equations are ubiquitous in computational science and often arise from finite discretizations of partials differential equations.
- Consequentially, the ability to efficiently solve these systems is of great importance, and research in this direction has become a fundamental part of linear algebra in modern times.
- Techniques used to solve linear systems can be classified as <u>direct and iterative methods</u>.
- Most direct methods [1] can be interpreted as variations of the famous Gauss Elimination algorithm, which generates exact solutions at the cost of $O(n^3)$ for a dense $n \times n$ matrix.
- However, the matrices that arise from the discretization of physical systems often possess <u>special structures</u> that can be utilized to devise more efficient algorithms

1. Davis et al (2016)

Multigrid Algorithm

$$\underline{\underline{A}}\,\underline{x} = \underline{b}$$

- Developed over the last 25 years – Introduced in the 1970s [1,2]
- State of the art for linear elliptic problems
- Possibility of solving a linear system in a fixed number of iterations
- Good introduction to multigrid methods - Bramble *Multigrid Methods* 2019

What's a smoother?

An iterative method is called a smoother if the high frequency components of the error decay faster than low frequency components

Examples: Gauss Seidel, Weighted Jacobi, Successive Overrelaxation (SOR)

Fundamental Ideas

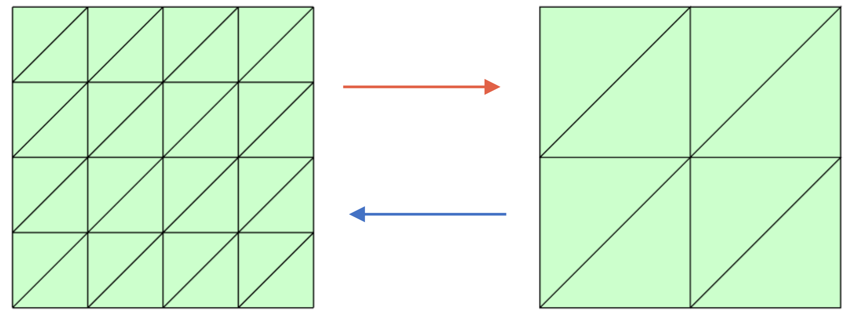
- High frequency components of the error are removed in a few iterations
- Switching to a coarse grid makes the previously low frequency components, high-frequency components ("smooth becomes rough")

1. Brandt 1973; 2. Brandt 1977

Two-Grid Algorithm

Key Ingredients :

1. Smoother
2. Restriction Operator
3. Prolongation Operator



$$\underline{\underline{A}}\,\underline{x} - \underline{\underline{A}}\,\underline{x}^{est} = \underline{b} - \underline{\underline{A}}\,\underline{x}^{est} \quad \Longrightarrow \quad \underline{\underline{A}}\,(\underline{x} - \underline{x}^{est}) = \underline{\underline{A}}\,\underline{e} = \underline{r}$$

Algorithm One cycle of a two-grid algorithm

1: Choose an initial guess $\underline{x}_h^0$

2: Relax ν_1 iterations of $\underline{\underline{A}}_h\,\underline{x}_h^0 = \underline{b}_h \rightarrow \underline{x}_h^{1/3}$ ▷ Pre-smoothing iterations

3: Compute the residual as $\underline{r}_h = \underline{b}_h - \underline{\underline{A}}_h\,\underline{x}_h^{1/3}$ ▷ Coarse grid correction

4: Restrict residual to coarse grid as $\underline{r}_{2h} = \underline{\underline{I}}_{2h}^{h}\,\underline{r}_h$

5: Compute coarse grid error as $\underline{\underline{A}}_{2h}\,\underline{e}_{2h} = \underline{r}_{2h} \rightarrow \underline{e}_{2h}$

6: Prolongate error to fine grid as $\underline{e}_h = \underline{\underline{I}}_{h}^{2h}\,\underline{e}_{2h}$

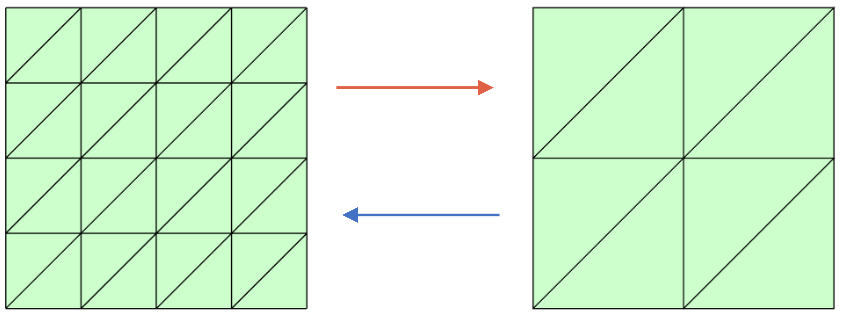
7: Correct $\underline{x}_h^{2/3} = \underline{x}_h^{1/3} + \underline{e}_h$

8: Relax ν_2 iterations of $\underline{\underline{A}}_h\,\underline{x}_h^{2/3} = \underline{b}_h \rightarrow \underline{x}_h^1$ ▷ Post-smoothing iterations

V-Cycle Algorithm

Key Ingredients :

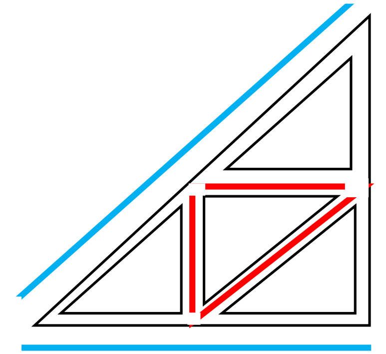
1. Smoother
2. Restriction Operator
3. Prolongation Operator



Algorithm 4.2 One cycle of a V-cycle algorithm

1: Choose an initial guess $\underline{x}_h^0$ ▷ $VG_h(\underline{x}_h^n, \underline{b}_h) \to \underline{x}_h^{n+1}$
2: Relax ν_1 iterations of $\underline{\underline{A}}_h\, \underline{x}_h^0 = \underline{b}_h \to \underline{x}_h^{1/3}$ ▷ Pre-smoothing iterations
3: Compute the residual as $\underline{r}_h = \underline{b}_h - \underline{\underline{A}}_h\, \underline{x}_h^{1/3}$
4: Restrict residual to coarse grid as $\underline{r}_{2h} = \underline{\underline{I}}_{2h}^{h}\, \underline{r}_h$
5: **if** h corresponds to coarsest grid **then**
6: $\quad \underline{x}_h^{2/3} = \underline{x}_h^{1/3}$
7: $\quad$ Skip to step 13
8: **else**
9: $\quad$ Compute coarse grid error as $VG_h(0, \underline{r}_{2h}) \to \underline{e}_{2h}$ ▷ Coarse grid correction
10: $\quad$ Prolongate error to fine grid as $\underline{e}_h = \underline{\underline{I}}_{h}^{2h}\, \underline{e}_{2h}$
11: $\quad$ Correct $\underline{x}_h^{2/3} = \underline{x}_h^{1/3} + \underline{e}_h$
12: **end if**
13: Relax ν_2 iterations of $\underline{\underline{A}}_h\, \underline{x}_h^{2/3} = \underline{b}_h \to \underline{x}_h^1$ ▷ Post-smoothing iterations =0

Multigrid for HDG



Difficulty with HDG:

- Numerical trace defined on the <u>edge space</u>
- Finer meshes have edges that are not refinements of the coarse mesh

Different approaches to resolve issue

1. Two-level multigrid scheme with a coarse space that consists of a piece-wise linear conforming (CG) FEM space [1,2]
2. Homogeneous HDG multigrid proposed in [3]

$$u_\ell = \mathcal{U}_\ell \lambda + \mathcal{U}_\ell f,$$
$$\boldsymbol{q}_\ell = \boldsymbol{\mathcal{Q}}_\ell \lambda + \boldsymbol{\mathcal{Q}}_\ell f.$$

$$\begin{tikzcd}
M_\ell & & V_\ell^{\mathrm{c}} \arrow[ll, "\gamma_\ell"'] \\
M_{\ell-1} \arrow[u, "I_\ell"'] \arrow[rr, "\mathcal{U}_{\ell-1}^{\mathrm{c}}"] & & V_{\ell-1}^{\mathrm{c}}. \arrow[u, "I_\ell^{\mathrm{c}}"]
\end{tikzcd}$$

$$[\mathcal{U}_\ell^{\mathrm{c}} \lambda](\boldsymbol{x}) = \begin{cases} \overline{\lambda}(\boldsymbol{x}) & \text{if } \boldsymbol{x} \text{ is on the boundary of a face,} \\ \lambda(\boldsymbol{x}) & \text{if } \boldsymbol{x} \text{ is in the interior of a face,} \\ [\mathcal{U}_\ell \lambda](\boldsymbol{x}) & \text{if } \boldsymbol{x} \text{ is in the interior of a cell.} \end{cases}$$

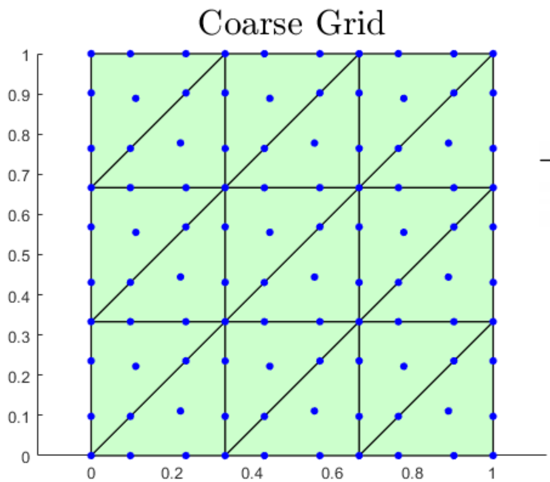
M_l - Edge space on refinement level l

V_l^c - Conforming space of the same porder on refinement level l

1. Cockburn et al 2013; 2. Tan & Gopalakrishnan 2009; 3. Lu et al 2021

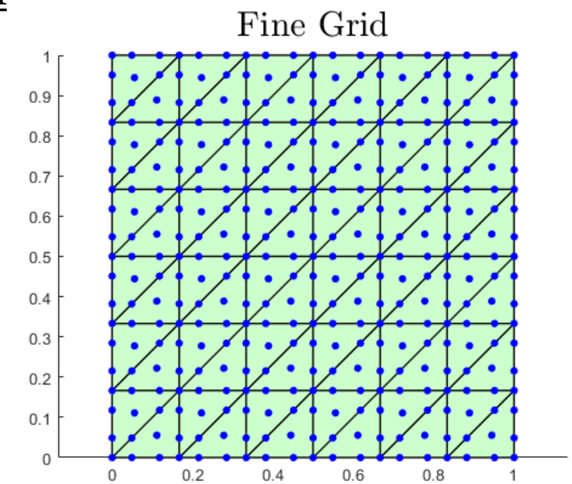
Injection and Restriction

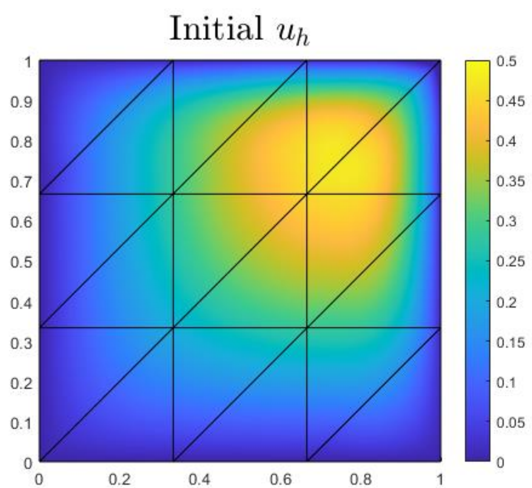
<u>Convection-Diffusion Equation</u>

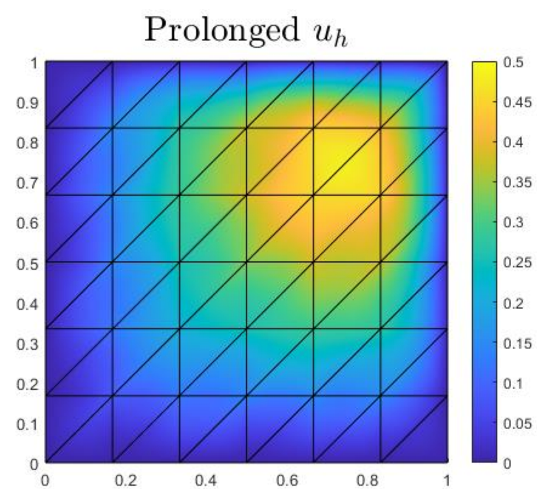


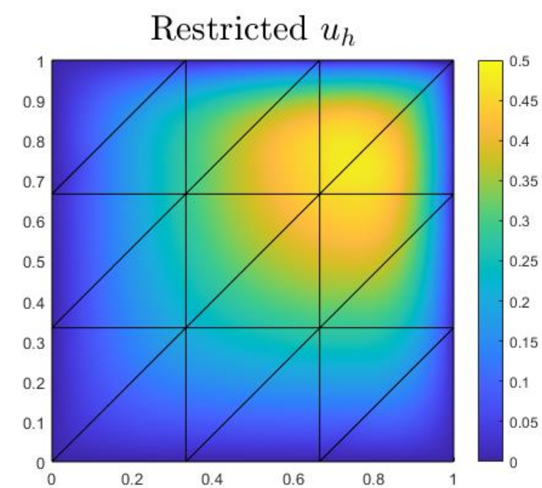
$$
\begin{aligned}
-\kappa\nabla^2 u + \nabla\cdot(\underline{c}u) &= f, \quad \text{in} \quad \Omega \\
u &= 0, \quad \text{on} \quad \Gamma
\end{aligned}
$$

$$\kappa = 1 \qquad \underline{c} = (10, 10) \qquad f = 1$$









As a sanity check, here we solve the equation, prolong the numerical trace to a finer mesh and reconstruct uh on this grid. Then, we restrict the numerical trace from the finer grid back to the coarser grid and reconstruct the uh.

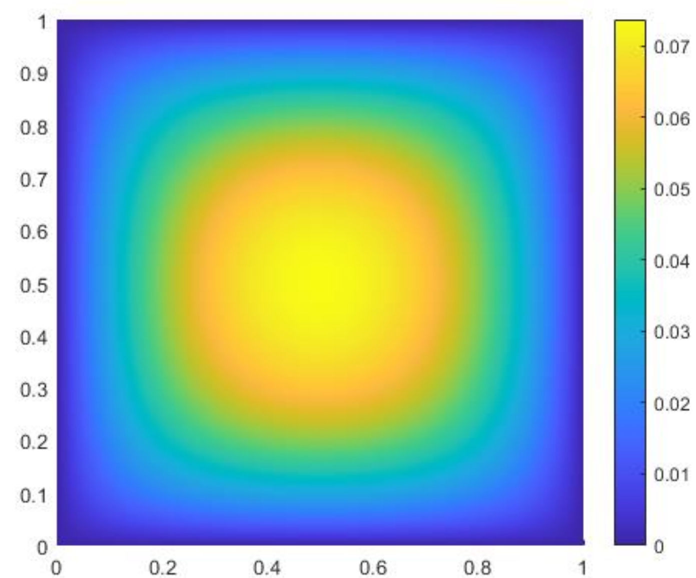
Numerical Results

Test Case : Poisson equation with $f = 1$ and homogeneous boundary conditions

Mesh level		$h = 0.354$	$h = 0.177$	$h = 0.088$	$h = 0.044$
$p = 1$	$\tau = 1$	82	308	1211	4816
$p = 2$	$\tau = 1$	113	435	1721	6860
$p = 3$	$\tau = 1$	147	581	2315	9249
$p = 4$	$\tau = 1$	186	722	2863	11422

Table 2: Number of iterations when using a Gauss Seidel iterative method

The number of iterations until convergence shoots up rapidly with resolution



Numerical Results

Test Case : Poisson equation with $f = 1$ and homogeneous boundary conditions

We see that the number of iterations until convergence is almost constant for our implementation

Current Work

Smoother / Mesh level		Two smoothing steps $h=0.354$	$h=0.177$	$h=0.088$	$h=0.044$	Four smoothing steps $h=0.354$	$h=0.177$	$h=0.088$	$h=0.044$
$p=1$	$\tau=\frac{1}{h}$	16	34	57	69	10	25	48	65
	$\tau=1$	16	33	56	69	9	24	47	64
$p=2$	$\tau=\frac{1}{h}$	29	39	51	56	12	27	42	48
	$\tau=1$	31	39	51	56	12	27	41	48
$p=3$	$\tau=\frac{1}{h}$	54	57	62	69	14	24	29	31
	$\tau=1$	62	75	96	153	14	23	29	31
$p=4$	$\tau=\frac{1}{h}$	90	82	77	72	18	35	48	54
	$\tau=1$	105	99	95	90	18	35	48	54

Table 1: Number of iterations with two and four smoothing steps as a function of polynomial order, grid size and stability parameter value

Lu et al 2021

TABLE 1 *Numbers of iterations with one and two smoothing steps for $f \equiv 1$. The polynomial degree of the HDG method is p*

Smoother		One step						Two steps					
Mesh level		1	2	3	4	5	6	1	2	3	4	5	6
$p=1$	$\tau=\frac{1}{h}$	33	39	38	36	35	35	17	20	19	19	18	18
	$\tau=1$	33	39	36	35	34	33	17	19	18	18	17	17
$p=2$	$\tau=\frac{1}{h}$	13	12	11	10	10	09	08	07	07	06	06	05
	$\tau=1$	13	12	11	10	10	09	08	07	07	06	06	05
$p=3$	$\tau=\frac{1}{h}$	24	25	25	25	25	25	15	15	15	15	15	15
	$\tau=1$	24	25	25	25	25	25	15	15	15	15	15	15