Multigrid method

Emile Parolin

February 6, 2025

1 Operators

1.1 Smoothers $\mathcal{S}$

The smoothing operator corresponds to one iteration of an iterative operator based on a matrix splitting, for instance one iteration of Jacobi; Gauss-Seidel; SOR...

1.2 Prolongation operator

The simplest prolongation operator $\mathcal{I}_{2h}^{h} : \mathcal{M}_{2h} \to \mathcal{M}_{h}$ is linear, based on the following scheme

$$\left] \begin{array}{ccc} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{array} \right[. \tag{1}$$

This notation should be read as follows: the center of the matrix is placed on each node of the coarse grid and its value contributes to each adajacent node of the fine grid with the weight indicated in the matrix.

1.3 Restriction operator

The simplest restriction operator $\mathcal{I}_{h}^{2h} : \mathcal{M}_{h} \to \mathcal{M}_{2h}$ is a sampling operator

$$U_{i,j}^{2h} = U_{2i,2j}^{h}. \tag{2}$$

It is preferable to use another operator with weights given by the following matrix

$$\frac{1}{4} \left[\begin{array}{ccc} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{array} \right] . \tag{3}$$

This time the center of the matrix is placed on each node of the fine grid and its value contributes to each adjacent node of the coarse grid with the weight indicated in the matrix.

2 Multigrid cycle

We consider a mesh defined by $h = \frac{1}{2^i}$, $i \geq 2$, a matrix A_h, a right-hand-side $\mathbf{F}^h$ and an initial guess $\mathbf{U}_0^h$.

Algorithm 2.1: Multigrid cycle $\mathbf{U}^h = \mathrm{MG}\big(A_h, \mathbf{U}_0^h, \mathbf{F}^h, \nu_1, \nu_2, \gamma\big)$

Input: $A_h, \mathbf{U}_0^h, \mathbf{F}^h, \nu_1, \nu_2, \gamma$.
Pre-smoothing: $\mathbf{U}^h = \mathcal{S}^{\nu_1}\big(A_h, \mathbf{U}_0^h, \mathbf{F}^h\big)$;
Coarse residual: $\mathbf{r}^H = \mathcal{I}_h^H \left(\mathbf{F}^h - A_h \mathbf{U}^h\right)$;
if $H = h_0 = \frac{1}{2}$ **then**
  Exact solve (scalar equation): $A_H \boldsymbol{\delta}^H = \mathbf{r}^H$;
else
  Recursive call: $\boldsymbol{\delta}^H = \mathrm{MG}^{\gamma}\big(A_H, \mathbf{0}, \mathbf{r}^H, \nu_1, \nu_2, \gamma\big)$;
end
Update: $\mathbf{U}^h = \mathbf{U}^h + \mathcal{I}_H^h \boldsymbol{\delta}^H$;
Post-smoothing: $\mathbf{U}^h = \mathcal{S}^{\nu_2}\big(A_h, \mathbf{U}^h, \mathbf{F}^h\big)$;
return $\mathbf{U}^h$.

The parameters $\nu_1 \geq 0$ and $\nu_2 \geq 0$ give the number of times the smoothing operators are applied, and $\gamma > 0$ the number of recursive call to perform (for $\gamma = 1$ this is a V-cycle and for $\gamma = 2$ this is a W-cycle).

3 Full multigrid cycle

This is the actual multigrid solver.

Algorithm 3.1: Full multigrid cycle

Input: $p, \nu_1, \nu_2, \gamma, \mu$.
Let $h = h_0 = \frac{1}{2}$ and solve exactly (scalar equation): $A_h \mathbf{U}^h = \mathbf{F}^h$;
for $l \leftarrow 1$ **to** p **do**
  $\mathbf{U}^{h/2} = \mathcal{I}_h^{h/2} \mathbf{U}^h$;
  $h = \frac{h}{2}$;
  $\mathbf{U}^h = \mathrm{MG}^{\mu}\big(A_h, \mathbf{U}_0^h, \mathbf{F}^h, \nu_1, \nu_2, \gamma\big)$;
end
return U^h.

The parameter p gives the highest level of refinement on which one wishes to obtain the solution and the parameter μ is the number of multrigrid cycles to perform at each refinement level.