Assignment 2: Advanced computational fluid dynamics (ME670)-2025

Department of Mechanical Engineering, Indian Institute of Technology Guwahati

- Provide: (i) the grid detail, (ii) the discretized equations detail, (iii) the boundary condition implementation detail, (iv) a well-documented code, (v) the required output (plots/any other such means).
- Items (i), (ii), and (iii) above should be written out/typed on a separate sheet and attached before items (iv) and (v).
- The input parameters such as the grid size (n), Jacobi weight (ω) and constants ν, C, k and σ should be read from an input file.

---

1. Develop a V-cycle program for the one-dimensional model problem $-u''(x) + \sigma u(x) = f$, with homogenous boundary conditions, solved using finite difference method. Write a function/subroutine for each individual component of the algorithm as follows.

(a) Given an approximation array $\underline{v}$, a right-side array $\underline{f}$, and a level number $1 \leq l \leq L$ (smallest level number corresponds to the finest grid), write separate subroutines that will carry out ν number of weighted Jacobi or Gauss-Seidel sweeps on level l. Keep them in a file named "relaxation_methods" for example: relaxation_methods.F90

(b) Given an array $\underline{f}$ and a level number $1 \leq l \leq L-1$, write a subroutine that will carry out full weighting between level l and level $l+1$. Keep it in a file named "restriction_methods".

(c) Given an array $\underline{v}$ and a level number $2 \leq l \leq L$, write a subroutine that will carry out linear interpolation between level l and level $l-1$. Keep it in a file named "prolongation_methods".

(d) Write a subroutine named 'V_cylce' that carries out a single V-cycle by calling the three preceding subroutines. The V-cycle should be able to start from a given level l. Keep it in a file named "MG_methods".

(e) Write a main program that initializes the data arrays and calls V-cycle subroutine. For testing, for fixed k, take $f(x) = C\sin(k\pi x)$ on the interval $0 \leq x \leq 1$, where C is a constant. Then the exact solution to model problem is

$$u(x) = \frac{C}{\pi^2 k^2 + \sigma}\sin(k\pi x)$$

(f) Write another subroutine that computes 2-norm of error and residual. Keep it in a file named "postprocessing_methods". You can also keep files writing subroutines in this file.

(g) Take $n = 512$, $\omega = 2/3$, $\nu_1 = \nu_2 = 2$, $\sigma = 1$ and $C = \pi^2 k^2 + \sigma$. For $k = 1$ and 10, apply weighted Jacobi, Gauss-Seidel and V-cycle iterations till the residual 2-norm is greater than 10^{-6}. Plot the residual norm against the number of iterations for all three methods on the same figure.

(h) Compare analytical and numerical solution by plotting them together.

2. Using the V-cycle subroutine, write a full multigrid (FMG) subroutine named 'FMG'. The FMG-cycle should start from a given level l. Keep it in the file named "MG_methods". Verify the solver using the problem statement given in Q1. Report the 2-norm of the residual after applying one FMG-cycle to the test problem in Q1. Afterwards, apply V-cycle to solution approximation till the residual 2-norm is greater than 10^{-6}. Report the number of V-cycle iterations. Compare analytical and numerical solution.

Coding guidelines/tips:

- Refer to assignment 1.
- Use **make** utility to compile and combine your codes kept in different files.

References:

- Mecklenburg, R. (2004). *Managing Projects with GNU Make: The Power of GNU Make for Building Anything*. " O'Reilly Media, Inc.".
- https://makefiletutorial.com/.