Applied Computational Fluid Dynamics

Course objective:

Applied Computational Fluid Dynamics (CFD) provides an introduction to the theoretical fundamentals as well as to the use of commercial CFD codes to analyze flow and heat transfer in problems of practical engineering interest. An overview of the theory and numerics of CFD is provided, but students are not expected to write programs. Students are trained to preprocess raw geometric data, mesh it and develop a CFD model. The students will understand the process of developing a geometrical model of the flow, applying appropriate boundary conditions, specifying solution parameters, and visualizing the results. They will also have an appreciation for the factors limiting the accuracy of CFD solutions.

Course content:

- Discretization techniques: Discretisation techniques using finite difference methods: Taylor-Series and control volume formulations; Finite element discretization techniques.
- Modelling of diffusion problems using finite volume method: One dimensional steady state diffusion problems; discretization technique; Solution methodology for linear and non-linear problems: Point-by-point iteration, TDMA; Two and three dimensional discretization; Discretization of unsteady diffusion problems: Explicit, Implicit and Crank-Nicolson’s algorithm; stability of solutions.
- Modelling of Convection-Diffusion Problems: One dimensional convection-diffusion problem: Central difference scheme; Discretization based on analytical approach (exponential scheme); Hybrid and power law discretization techniques; Higher order schemes (QUICK algorithm).
- Flow modeling: Discretization of incompressible flow equations; Pressure based algorithm: SIMPLE, SIMPLER etc; Unstructured grids; Introduction to FVM with unstructured grids; Introduction to turbulence modeling; Large Eddy Simulation (LES); Direct Numerical Simulation (DNS).
- Projects / Exercises/ Publications: Solving simplified problems: formulation, discretization with coarse grids, applying appropriate boundary and initial conditions and solving by hand calculations; Solving practical problems through software: writing user sub-routines; post-processing and interpretation of results.

References:


