

## Chapter 2

### CFD SIMULATION

---

---

#### 2.1. INTRODUCTION

Computational fluid dynamics or CFD is the analysis of systems involving fluid flow, heat transfer, chemical reactions and associated phenomena by means of simulation. It is a branch of continuum mechanics which deals with numerical simulation of fluid flow and heat transfer problems. This technique is very powerful and is widely used for industrial and non-industrial applications. Since 1960, the CFD techniques is used in aerospace industries for design and manufacturing of aircraft and jet engines. It is also used in various applications such as hydrodynamics of ships, combustion in IC engines and gas turbines, micro-fluidics of heat exchanger, cooling of electrical equipment including micro-circuits, chemical process engineering, and bio-medical engineering.

The problems in fluid mechanics can be solved using three approaches such as experimental, theoretical and computational (CFD) method. The experimental method is more realistic but involves high cost, measurement difficulties, equipment availability etc. The theoretical method is an analytical approach to solve the fluid flow problems. However, it is restricted to only simple geometry, simple flow physics and linear equations. On the contrary, the computational technique involves use of computer to solve the complex flow problems. Due to this, complex flow physics involving non-linear equations and the time evolution of the flow can be easily obtained. It is a suitable alternative approach to study the systems where experiments are difficult to be performed. Similarly, it is able to study the system under hazardous conditions beyond their performance limits. The CFD simulation is useful in micro and nano systems where the process phenomena are not clearly understood.

The three principles governing the physical aspects of any fluid flow are conservation of mass, momentum and energy. These are expressed in mathematical equations which are either integral or partial differential equations. Computational fluid dynamics is the technique in which the integrals or partial derivatives are replaced with discretized algebraic equations which are solved to obtain flow-field values at various points in time and space. The CFD solution involves conversion of the governing equations for a continuum medium into a set of discrete algebraic equations using a process called discretization. The numerical solution of these equations is found using computer at a desired level of accuracy. There are many discretization approaches and the most popular among them are finite difference method (FDM), finite element method (FEM) and finite volume method (FVM). The choice of the particular discretization method depends on the geometry, preference and pre-dominant trend in that particular application area. FEM is widely used for stress analysis applications, while FDM is used for simulation of turbulent flows and FVM is preferred in commercial CFD codes [77]. The FDM is the oldest method and it is also the easiest method to formulate the problems on simple geometries. In FDM, the domain is discretized using a structured grid.

The conservation equations in the partial differential form are approximated at each grid point by replacing the partial derivatives by finite difference approximations in terms of nodal values of the unknown variables. This results in an algebraic equation for each node and are collected for all the grid points, resulting in a system of discrete equations which are solved to yield the approximate solutions at the grid nodes. The major disadvantage of FDM is that it is restricted to simple geometries. In FEM, the domain is discretized using generally unstructured grid. The unknown variable is approximated using an interpolation procedure in terms of nodal values and a set of known functions (called shape functions). This approximation is substituted into the differential equation and the resulting error is minimized using a weighted residual procedure. This leads to a system of discrete equations in terms of unknown nodal values which is solved to obtain the solution. FEM is ideally suited for complex geometries and is popular in computational solid mechanics. The FVM is based on the integral form of conservation equations. The domain

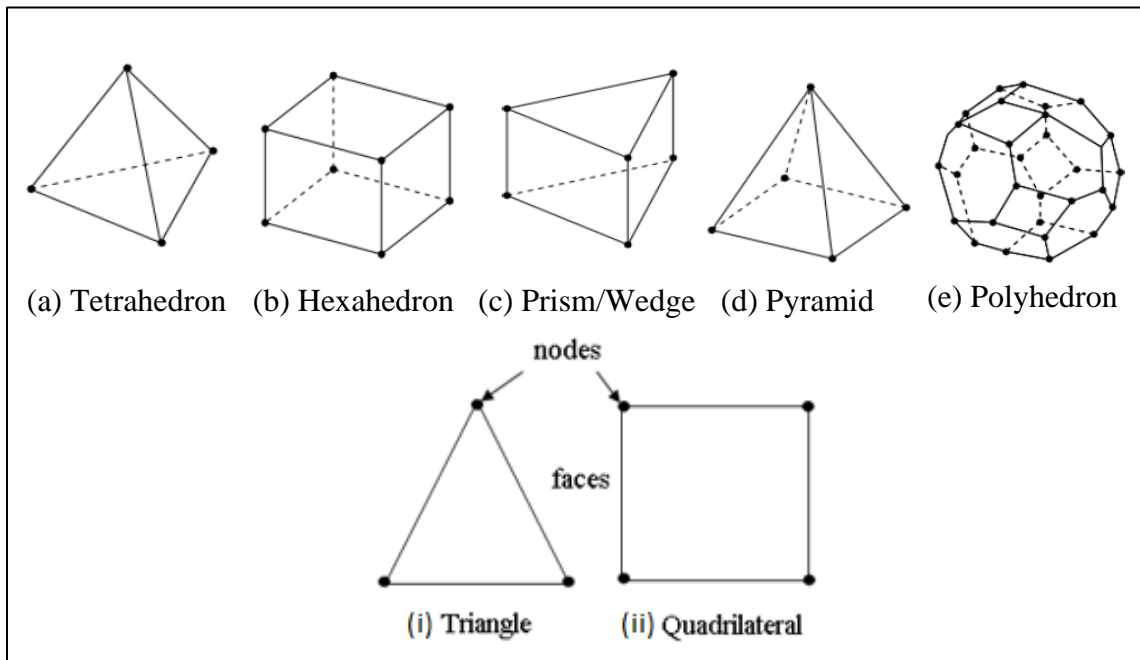
is discretized into a set of non-overlapping control volumes (finite volumes). The conservation equations are applied to each finite volume. The integrals occurring in the conservation equations are evaluated using function values at nodes (centroids of finite volumes). This involves use of approximate integral formulae and interpolation methods to obtain the values of variables at surfaces of the control volumes. The FVM can accommodate any type of grid and is suitable for complex geometries and hence it is popular in commercial CFD softwares.

The various CFD software packages are available out of which OpenFOAM is widely used open source software. It is based on C++ language to develop customized numerical solvers for continuum mechanics problems including CFD. Other softwares such as ANSYS Fluent, Star-CCM+, COMSOL are the popular and expensive licensed software. COMSOL Multiphysics is a common platform for finite element analysis, solver and multiphysics simulation software. Among these softwares, Fluent is most popular and has a maximum share in the electronic and industrial product market. In Fluent various flow such as steady, transient, laminar, turbulent, Newtonian, non-Newtonian, single and multiphase flows, chemical reactions including combustion, flow through porous media, heat transfer, and flow-induced vibrations can be easily modeled. The software packages used for specific applications such as electronics cooling, ventilating systems, and mixing are also available. For the present simulation study, Fluent software is used.

## **2.2. MESH GENERATION**

The partial differential equations that govern fluid flow and heat transfer can be solved for simple and small geometry. However, when the domain under study is complex and large, then the flow domain is split into smaller subdomains. These subdomains are made up of geometric primitives which are as shown in Figure 2.1. The governing equations are discretized and solved inside each of the subdomains and combined together to have a complete solution of the entire fluid domain. The subdomains are called elements or cells and their collection is called a mesh or grid. The 2D and 3D mesh elements available in Fluent are shown in Figure 2.1. The 2D mesh is comprised of triangular and

quadrilateral elements or combination of the both and the 3D elements consists of the tetrahedron, hexahedron, prism, pyramid, and polyhedron. In case of complex geometries, a triangular/tetrahedral mesh is used as compared to quadrilateral/hexahedral mesh. This is because it can create a smaller number of cells due to clustering of cells in the selected region of the flow domain. The type of mesh to be used for different geometry is summarized in Table 2.1.



**Figure 2.1** Schematic diagram showing 2D and 3D mesh elements

**Table 2.1** Type of mesh used for different geometry

Complexity of geometry	Type of mesh used
Simple geometry	Quadrilateral / hexahedral
Moderately complex geometry	Unstructured quadrilateral / hexahedral
Relatively complex geometry	Triangular/tetrahedral meshes with prism layers
Extremely complex geometry	Pure triangular/tetrahedral meshes

The quality of the mesh generated plays a significant role in the accuracy and stability of the numerical computation. The three important metrics deciding the quality of mesh are orthogonal quality, aspect ratio, and skewness. Orthogonal quality is computed for each cell and its value ranges from 0 to 1. The value of 0 indicates the worst cells with zero or negative volumes and the value of 1 indicates the best cells with perfect cube or square. Another important indicator of the mesh quality is the aspect ratio. The aspect ratio is a measure of the stretching of a cell. It is computed as the ratio of the maximum value to the minimum value of the normal distances between the cell centroid and face centroids or the distance between cell centroid and nodes. In addition to the above quality indicators, the skewness is also measured and it is the difference between the shape of the cell and the shape of an equilateral cell of equivalent volume. Highly skewed cells can affect the accuracy and destabilize the solution.

### 2.3. GOVERNING EQUATIONS

The CFD modeling is based on fundamental governing equations of fluid dynamics: the conservation of mass, momentum, and energy. The conservation equations for mass and momentum are solved for all types of flows but when the flow involves heat transfer or compressibility, an additional equation for energy conservation is solved. The turbulent flow is highly random in nature and hence the transport equations are solved which represent its characteristics.

The equation for conservation of mass, or continuity equation for laminar flow is given as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = S_m \quad (2.1)$$

this is the general form of the mass conservation equation and is valid for compressible and incompressible flows. The first term on the left-hand side is the rate of change of the density and the second term is the net flow of mass (convective term). The source  $S_m$  is the mass added to the continuous phase from the dispersed second phase (vaporization of liquid droplets) and any user-defined sources [78].

The conservation of momentum is given by

$$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot (\bar{\tau}) + \rho \vec{g} + \vec{F} \quad (2.2)$$

where  $\vec{v}$  is the velocity vector,  $p$  is the static pressure,  $\rho$  is the density,  $g$  is the gravity,  $\vec{F}$  is the external body forces that arise from interaction with the dispersed phase or user-defined sources. The left-hand side of the equation is the acceleration and it may contain convective terms and the right-side of the equation represents hydrostatic effects, divergence of deviatoric stress, gravitational force, and external body forces respectively.  $\bar{\tau}$  is the stress tensor and is given by

$$(\bar{\tau}) = \mu [ (\nabla \vec{v} + \nabla \vec{v}^T) - \frac{2}{3} \nabla \cdot \vec{v} I ] \quad (2.3)$$

where  $I$  is the unit tensor,  $\mu$  is the viscosity and the second term on the right-hand side is the effect of volume increase [78].

The conservation of energy is given by

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\vec{v} (\rho E + p)) = \nabla \cdot \left( k_{eff} \nabla T - \sum_j h_j \vec{J}_j + (\bar{\tau}_{eff} \cdot \vec{v}) \right) + S_h \quad (2.4)$$

where  $k_{eff}$  is the effective conductivity  $= (k + k_t)$ , where  $k$  is thermal conductivity and  $k_t$  is the turbulent thermal conductivity,  $T$  is the temperature,  $\vec{J}_j$  is the diffusion flux of species  $j$ . The first three terms on the right-hand side represents energy transfer due to conduction, species diffusion, and viscous dissipation, respectively.  $S_h$  includes the heat of chemical reaction, and any other volumetric heat sources defined by user,  $h$  is the sensible enthalpy and  $E$  is the energy which is given as [78]

$$E = h - \frac{p}{\rho} + \frac{v^2}{2} \quad (2.5)$$

In actual practice, various turbulence models available are selected for different classes of problems. The choice of turbulence model will depend on considerations such as the physics of the flow, the established practice for a specific class of problem, the level of accuracy required, the available computational resources, and the amount of time available for the simulation. The most appropriate choice of model for any application

depends on the capabilities and limitations of the several options available in the model. The turbulent flows are characterized by irregular nature of flow (chaotic), diffusivity which enhances mixing, rotationality which generates vortex and dissipative which transforms kinetic energy into internal energy by viscous shear stress. Due to different characteristics many turbulence models are developed to capture the flow behavior in real life applications. Out of the various available models, few popular models such as Spalart-Allmaras model,  $k - \omega$  models, Transition  $k - kl - \omega$  model,  $V^2 - f$  model and  $k - \varepsilon$  models are discussed. The Spalart-Allmaras model was designed for aerospace applications involving wall-bounded flows and shows good results for boundary layers subjected to adverse pressure gradients. It is also popularly used in turbomachinery applications. It is a one-equation model that solves a modeled transport equation for the kinematic eddy (turbulent) viscosity. However, it is not rectified for general industrial flows and produces large errors for some free shear flows such as plane and round jet flows. The  $k - \omega$  model is used to predict the turbulence parameters very close to boundary or wall region. The two types of  $k - \omega$  model is available that is standard and shear-stress transport (SST) model. The standard  $k - \omega$  model is based on model transport equations for the turbulence kinetic energy ( $k$ ) and the specific dissipation rate ( $\omega$ ) which is the ratio of  $\varepsilon$  (dissipation rate) and  $k$ . The  $k - kl - \omega$  transition model is used to find transition of the boundary layer from a laminar to a turbulent regime. The model is a three-equation eddy-viscosity type, which includes transport equations for turbulent kinetic energy ( $k_T$ ), laminar kinetic energy ( $k_L$ ), and the inverse turbulent time scale ( $\omega$ ). The  $V^2 - f$  model is used for flows dominated by separation. It is similar to the standard  $k - \varepsilon$  model, but incorporates near-wall turbulence anisotropy and non-local pressure-strain effects. Instead of  $k$ , here velocity scale  $\overline{v^2}$  is used to evaluate eddy viscosity. This model cannot be used to solve Eulerian multiphase problems.

The  $k - \varepsilon$  models [79] are the most popular and widely used for different engineering flow calculations and are presented in detail. The two equation turbulence models allow the determination of a turbulent length and time scale by solving two separate transport equations. The three  $k - \varepsilon$  models available are Standard, Renormalization-group

(RNG) and Realizable  $k - \varepsilon$  model. To improve the performance of the standard  $k - \varepsilon$  model, modifications have been introduced and two variants RNG  $k - \varepsilon$  model [80] and realizable  $k - \varepsilon$  [81] are developed. The modeled transport equations for  $k$  and  $\varepsilon$  in the standard  $k - \varepsilon$  model are

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + G_k + G_b - \rho \varepsilon - Y_M + S_k \quad (2.6)$$

and

$$\frac{\partial(\rho \varepsilon)}{\partial t} + \frac{\partial(\rho \varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_j} \right] + C_{1\varepsilon} \frac{\varepsilon}{k} (G_k + C_{3\varepsilon} G_b) - C_{2\varepsilon} \rho \frac{\varepsilon^2}{k} + S_\varepsilon \quad (2.7)$$

and the modeled transport equations for  $k$  and  $\varepsilon$  in the realizable  $k - \varepsilon$  model are [61]

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_j)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + G_k + G_b - \rho \varepsilon - Y_M + S_k \quad (2.8)$$

and

$$\frac{\partial(\rho \varepsilon)}{\partial t} + \frac{\partial(\rho \varepsilon u_j)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_j} \right] + \rho C_{1\varepsilon} S_\varepsilon - \rho C_{2\varepsilon} \frac{\varepsilon^2}{k + \sqrt{\nu \varepsilon}} + C_{1\varepsilon} \frac{\varepsilon}{k} C_{3\varepsilon} G_b + S_\varepsilon \quad (2.9)$$

where  $C_1 = \max \left[ 0.43, \frac{\eta}{\eta+5} \right]$ ,  $\eta = S \frac{k}{\varepsilon}$ , and  $S = \sqrt{2 S_{ij} S_{ij}}$

$G_k$  represents the generation of turbulence kinetic energy due to the mean velocity gradients which is given by

$$G_k = -\rho \overline{u_i u_j} \frac{\partial u_j}{\partial x_i} \quad (2.10)$$

$G_b$  is the generation of turbulence kinetic energy due to buoyancy and is given by

$$G_b = \beta g_i \frac{\mu_t}{Pr_t} \frac{\partial T}{\partial x_i} \quad (2.11)$$

where  $Pr_t$  is the turbulent Prandtl number for energy and  $g_i$  is the component of the gravitational vector in the  $i^{\text{th}}$  direction. For standard and realizable  $k - \varepsilon$  models, the default value of  $Pr_t$  is 0.85.  $Y_M$  represents the contribution of the fluctuating dilatation in



compressible turbulence to the overall dissipation rate and this is neglected in the modeling of incompressible flows,  $\nu$  is the molecular kinematic viscosity,  $\mu_t$  is the turbulent viscosity,  $\bar{u}_i$  is the mean velocity component,  $u_i$  is the fluctuating velocity component,  $C_2$ ,  $C_{1\varepsilon}$ ,  $C_{3\varepsilon}$  are constants,  $\sigma_k$  and  $\sigma_\varepsilon$  are the turbulent Prandtl numbers for  $k$  and  $\varepsilon$  respectively,  $S_k$  and  $S_\varepsilon$  are user-defined source terms,  $\beta$  is the coefficient of thermal expansion and is given by

$$\beta = \frac{1}{\rho} \left( \frac{\partial \rho}{\partial T} \right) p \quad (2.12)$$

All the three  $k - \varepsilon$  models have similar forms with transport equations for  $k$  and  $\varepsilon$ . However, the major differences are in the method of calculating turbulent viscosity, the turbulent Prandtl numbers governing the turbulent diffusion, and the generation/destruction terms in the  $\varepsilon$  equation. The realizable  $k - \varepsilon$  model differs from the standard  $k - \varepsilon$  model in two ways. The realizable  $k - \varepsilon$  model contains an alternative formulation for the turbulent viscosity. A modified transport equation for the dissipation rate  $\varepsilon$  has been derived from an exact equation for the transport of the mean-square vorticity fluctuation. The term ‘realizable’ means that the model satisfies certain mathematical constraints on the Reynolds stresses, consistent with the physics of turbulent flows. The equation of  $k$  is same in other models with the change in the model constants. However, the form of the  $\varepsilon$  equation is quite different in all models. The production term in the  $\varepsilon$  equation (the second term on the right-hand side) of the realizable model does not involve the production of  $k$ , that is it does not contain the same  $G_k$  term as the other two models. Likewise, the destruction term (the third term on the right-hand side) does not have any singularity that is its denominator never vanishes, even if  $k$  vanishes or becomes smaller than zero. This feature is contrasted with the standard and RNG models, which have a singularity due to  $k$  in the denominator.

Both the realizable and RNG  $k - \varepsilon$  models have shown substantial improvements over the standard  $k - \varepsilon$  model where the flow features include strong streamline curvature, vortices, and rotation [61] [81]. It is observed that the realizable model provides the best performance of all the  $k - \varepsilon$  model versions for several validations of separated flows and flows with complex secondary flow features. Hence in the present simulation, the

realizable  $k - \varepsilon$  model is found reasonably good to capture the rotation induced flow-field such as vortex flow.

## 2.4. DISCRETE PHASE MODELING (DPM)

Discrete phase modeling (DPM) in ANSYS fluent is used to study the behavior of the particles. The primary phase is dielectric fluid, and the secondary phase is debris particles. The primary phase is considered as a continuum by solving Navier Stokes equations, and the discrete phase is tracked by using Euler-Lagrange approach. Since the volume of debris particles is less than 10% of the continuous phase, the effect of particles on the dielectric fluid flow is neglected, and hence, one-way coupling is used. The trajectory of a discrete phase particle is calculated by integrating the force balance on the particle which is in the Lagrangian reference frame. This force balance equates the particle inertia with the forces acting on the particle. The force balance equation per unit mass of a particle is given as

$$\frac{du_p}{dt} = F_D(\vec{u} - \vec{u}_p) + \frac{\vec{g}(\rho_p - \rho)}{\rho_p} + \vec{F} \quad (2.13)$$

The left side of the equation ( $du_p/dt$ ) is the net particle inertia and the right side is the drag force, gravity force and additional forces in order [82]. Additional forces are the particle source terms. Drag force ( $F_D$ ) is given as:

$$F_D = \frac{18 \mu C_D R_E}{24 \rho_p d_p^2} \quad (2.14)$$

where  $\vec{u}$  is the fluid phase velocity,  $\vec{u}_p$  is the particle velocity,  $\mu$  is the dynamic viscosity of the fluid,  $\rho$  is the fluid density,  $\rho_p$  is the density of the particle, and  $d_p$  is the particle diameter.  $R_E$  is the relative Reynolds number which is defined as

$$R_E = \frac{\rho d_p |\vec{u}_p - \vec{u}|}{\mu} \quad (2.15)$$

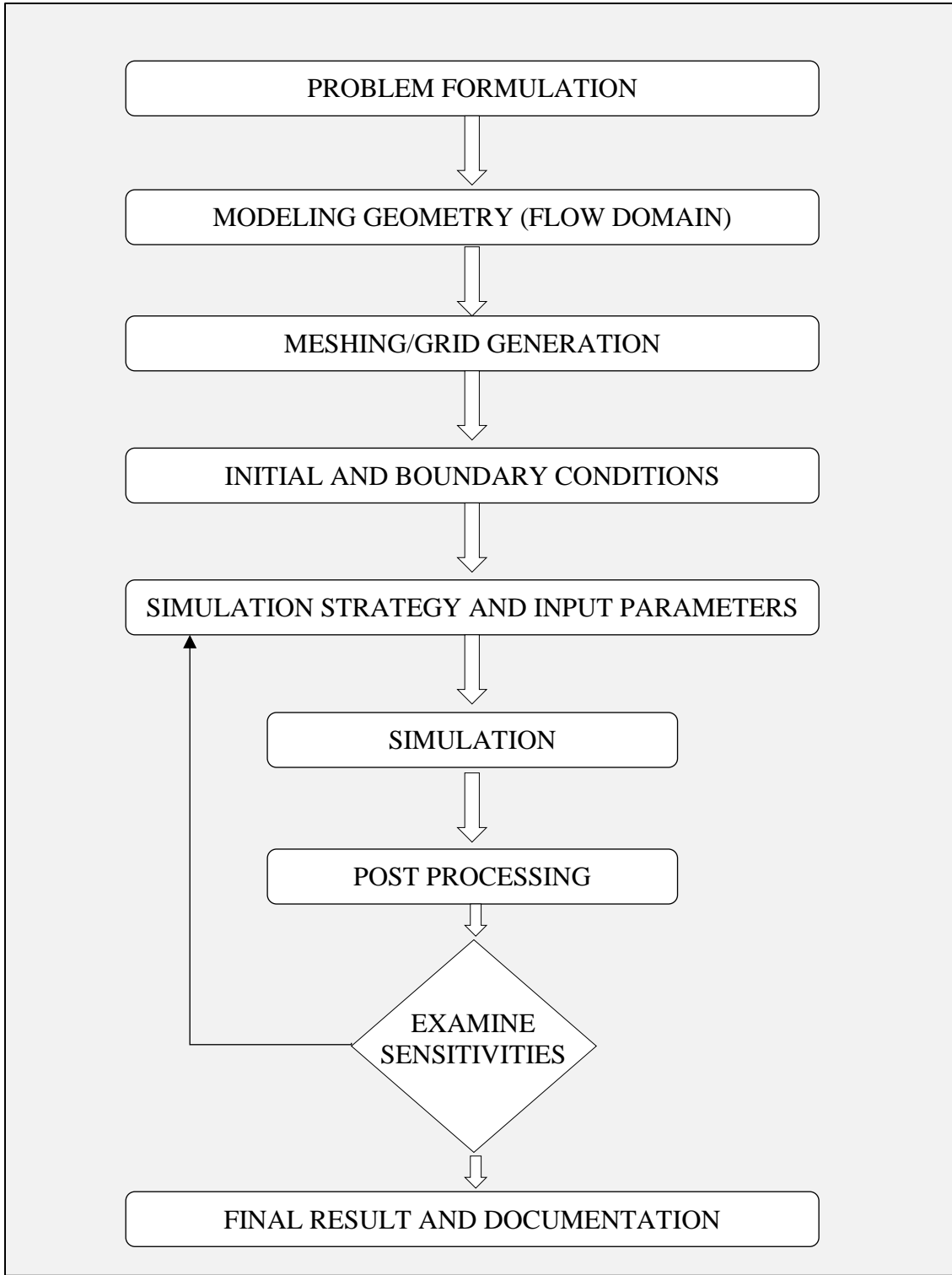
As the particles are spherical in shape, the drag coefficient for smooth spherical particles is given as

$$C_D = a_1 + \frac{a_2}{R_E} + \frac{a_3}{R_E^2} \quad (2.16)$$

where  $a_1$ ,  $a_2$  and  $a_3$  are constants that apply over several ranges of  $R_E$

## 2.5. CFD SIMULATION PROCESS

The CFD analysis require systematic and hierarchical approach to be followed. The major steps involved in CFD are categorized as preprocessing, computational solver and postprocessing. The flow chart showing the complete CFD analysis process is represented in Figure 2.2. The preprocessing involves problem formulation, modeling and meshing. The process starts with the selection of the real physical problem with the domain of interest and then the model is built to represent the same. The computational domain is selected and the mesh or grid is generated. The finite volume-based mesh is used to generate discrete meshing. Different software is used for meshing such as ICEM, Gambit, Gridgen, T-grid, and Hypermesh. The quality of the simulation results mainly depends on the quality of the mesh. After preprocessing, computation is executed by setting the initial conditions of the flow domain and selecting suitable boundary conditions (velocity, pressure, temperature etc.) on each edge and face depending upon the physics of the flow. The input parameters such as the type of fluid (water, kerosene, air etc.) along with its properties (density, viscosity, conductivity etc.) are specified. The numerical parameters and algorithms are selected and the convergence criteria is set. The final step is postprocessing which is visual representation of the simulated data. It involves gathering, recording and analyzing the data with different methods like contour plots, vector plots, streamlines, data curve etc. for appropriate graphical representation and report.

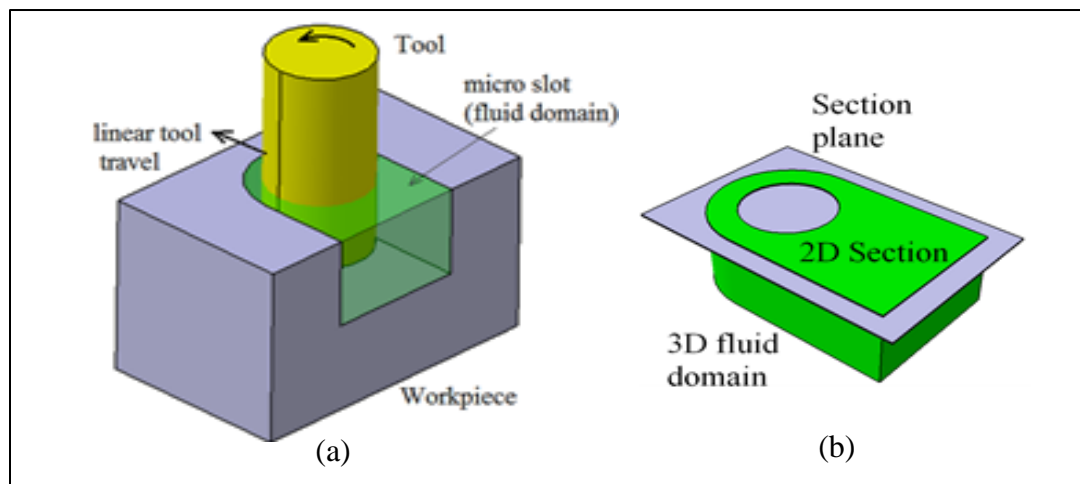


**Figure 2.2** Flow chart of CFD simulation process

## 2.6. PROBLEM DEFINITION

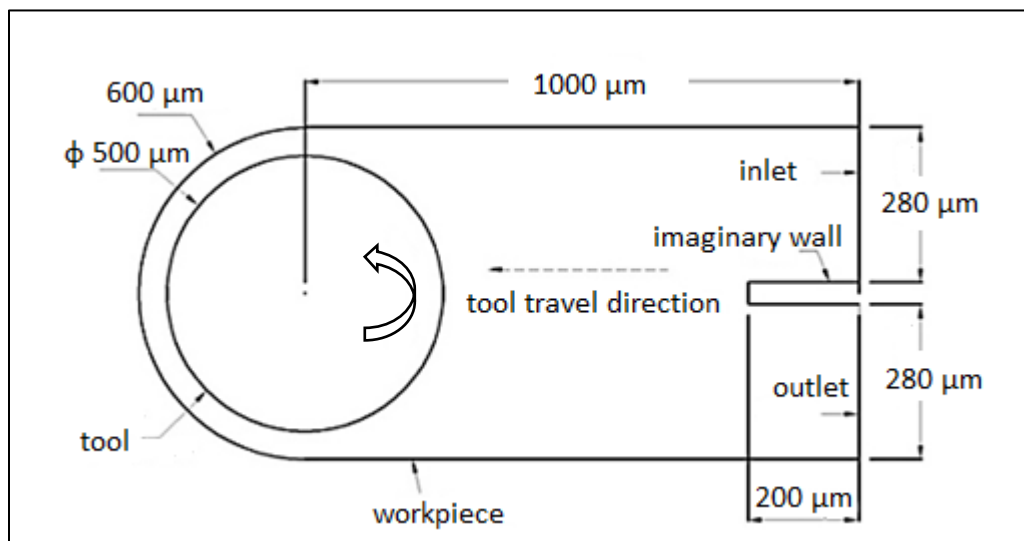
### 2.6.1 Model description

The simulation of the  $\mu$ ED-milling process is useful to understand the physical phenomenon of the process. The flow-field of the dielectric and debris movement in the gap can be visualized by CFD simulation. The schematic diagram representing the position of the tool in the  $\mu$ channel is shown in Figure 2.3 (a). The tool is rotating and is fed in a linear direction to generate spark and cut the  $\mu$ channel. During machining, certain portion of a  $\mu$ channel is made and tool positions itself to establish continuous spark along its circumferential surface with the workpiece. This position of tool inside the  $\mu$ channel during the machining is considered as the state of observation for CFD simulation. The gap between tool and workpiece is filled with liquid dielectric and is represented as fluid domain which is subjected to continuous motion by tool rotation and infeed of dielectric from the nozzle. During machining, debris are also formed at IEG which add up to the complexity of the problem. Since the behavior of  $\mu$ ED-milling process does not vary along the length of the channel only a planar cross-section is considered to represent the above phenomenon as shown in Figure 2.3 (b).



**Figure 2.3** Schematic diagram showing (a) position of tool in  $\mu$ channel and (b) planar section

The resemblance of the 2D section appearing at the position of the tool is represented schematically as shown in Figure 2.4 and the same is considered for simulation. The geometry of the model is considered with the tool diameter of  $500\ \mu\text{m}$ , width of the  $\mu\text{channel}$  cut is  $600\ \mu\text{m}$  with a constant IEG of  $50\ \mu\text{m}$  and the length of the channel cut is  $1000\ \mu\text{m}$ . This is simulated to the condition selected for experimentation. The IEG size depends on the input energy and for the energy level of  $2000\ \mu\text{J}$  it is nearly  $50\ \mu\text{m}$ . The tool represented in the model rotates in counter-clockwise direction at a high-speed ranging from 100 to 1000 rpm which uniformly agitates the fluid and make them move continuously along the IEG. The entire area between the tool and the workpiece is filled with dielectric fluid. To supply a continuous flow of dielectric, excess quantity is injected at the inlet cross-section. The used dielectric fluid after moving across the gap leaves the  $\mu\text{channel}$  through the outlet cross-section. To distinguish between the inlet and outlet of fluid flow, an imaginary wall is considered in the simulation as shown in Figure 2.4. Such a wall is not present in the actual  $\mu\text{ED}$ -milling process and its presence does not affect the flow pattern near the tool electrode which is discussed in the simulation of dielectric.



**Figure 2.4** Schematic diagram showing a 2D model of  $\mu\text{channel}$  used in CFD analysis

## 2.6.2 Boundary Conditions

The various types of the boundary conditions available are pressure inlet, velocity inlet, mass flow inlet, inlet vent, intake fan, pressure outlet, pressure far-field, outflow, outlet vent, and exhaust fan. The boundary condition for the inlet cross-section of the simulation model is velocity inlet and for outlet cross section is pressure outlet. The velocity inlet boundary condition is used to define the velocity and scalar properties of the flow at the inlet. The pressure outlet boundary condition is used to define the static pressure and other scalar variables in case of backflow. The pressure outlet boundary condition as compared to an outflow condition gives better convergence when backflow happens during iterations. The no-slip boundary condition is given to all other workpiece surfaces and the effect of gravity is neglected. The no-slip boundary condition indicates that the fluid has zero velocity at the wall surface. The moving reference frame is used to provide angular rotation to the tool. This enables to provide different speeds to the tool. Swirling and rotating flows requires a moving reference frame. The moving reference frame is used to model the problem involving a rotating boundary which moves with the fluid. Some examples include rotating blades, impellers, moving walls etc., the flow around them is of interest. When a moving reference frame is used, the equations of motions are modified to incorporate the additional acceleration terms which occur due to the transformation from the stationary to the moving reference frame. The property of each boundary is listed in Table 2.2.

**Table 2.2** Boundary settings of a computational model

<b>Boundary</b>	<b>Property</b>	<b>Type</b>
Inlet	Velocity-inlet	Stationary wall
Outlet	Pressure-outlet	Stationary wall
Workpiece	No slip condition	Stationary wall
Tool electrode	Moving reference frame	Rotating wall

### 2.6.3 Computational Solver

In Fluent, five pressure-velocity coupling algorithms are available. The main algorithm is Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) for pressure-velocity coupling. The SIMPLE algorithm obtains the pressure field from the relation between velocity and pressure corrections. Two numerical methods (solver) are available, pressure-based solver and density-based solver. The compressible flows are solved using density-based solver and the incompressible flows are solved using pressure-based solver. The pressure-based solver allows solving the flow problem in either a segregated or coupled manner. Both the solvers use similar discretization process (finite volume method), but the approach to linearize and solve the equations is different. In the present study, the pressure-based solver is used as the dielectric fluid is incompressible liquid. The second order discretization is used to achieve high order accuracy at cell faces. In this, quantities at cell faces are computed using a multidimensional linear reconstruction approach. In first order discretization, the face quantities are identical to the cell quantities and the accuracy is less. The realizable k-epsilon ( $k - \epsilon$ ) model is used for simulation. The common CFD parameters used for simulation of fluid flow and discrete phase modeling is summarized in Table 2.3.

**Table 2.3** CFD parameters

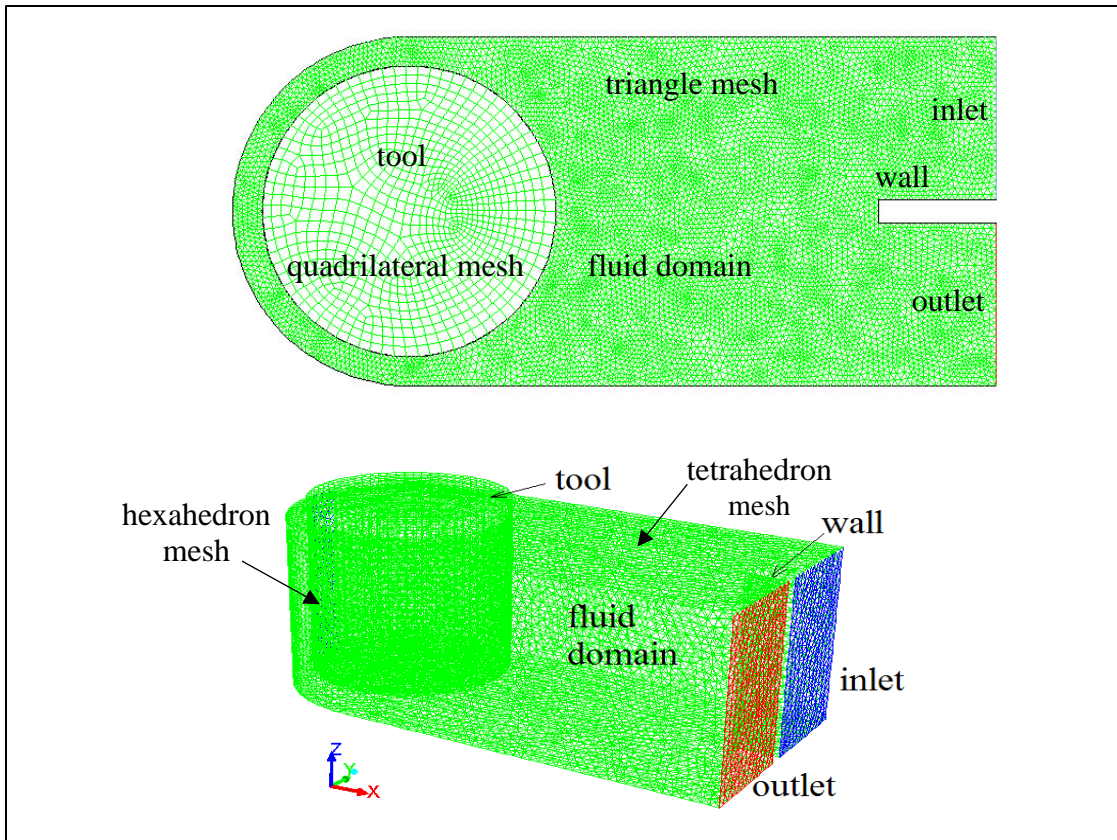
Solver type	Pressure based 2D double precision, no gravity effect
Velocity formulation	Absolute
2D space	Planar
Turbulence model	Realizable k-epsilon turbulence model
Pressure-velocity coupling	SIMPLE
Gradient option	Least Squares Cell-based
Pressure	Standard
Momentum	Second order upwinding
Turbulent kinetic energy	Second order upwinding
Turbulent dissipation rate	Second order upwinding
Energy discretization method	Second order upwinding
Residual absolute criteria	$10^{-3}$ for continuity, velocities, k & epsilon, $10^{-6}$ for energy



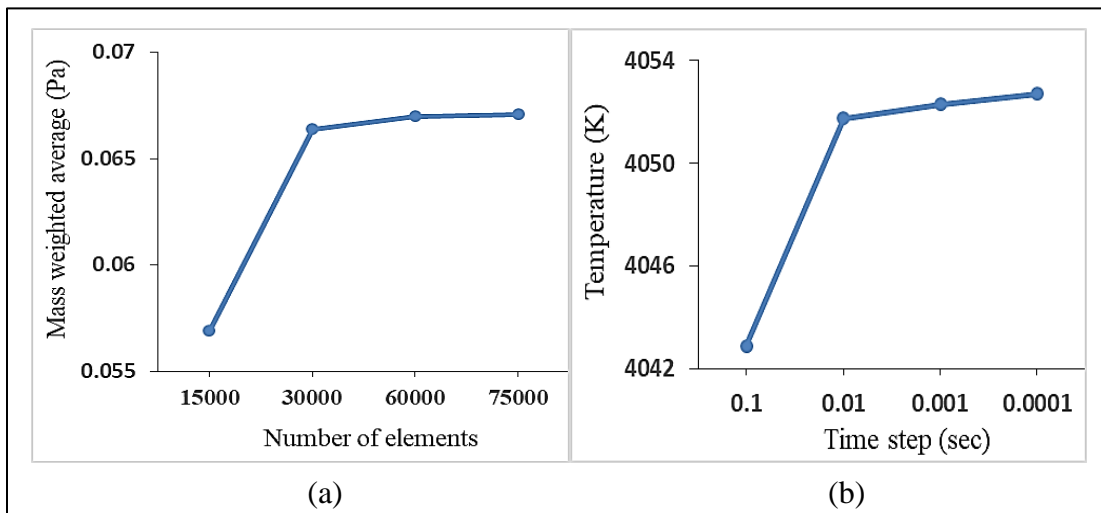
#### 2.6.4 Meshing

The meshed model considered for simulation is shown in Figure 2.5. Different meshing for two different domains: solid and fluid domain are used. A fluid domain is the group of cells for which all active equations are solved. Since the major flow behavior changes happen in the fluid domain, it has been meshed with structured triangular elements. A solid domain is the group of cells for which only a heat conduction equation is solved and no flow equations are solved. The rotating tool which is a solid domain, is meshed with quadrilateral elements. The meshing is generated automatically by the ANSYS mesh tool. As the dimensions are in micron meters, the entire domain is finely meshed to improve the accuracy of the results. The minimum orthogonal quality of the generated mesh is 0.53, the maximum aspect ratio is 6.56 and the maximum skewness is 0.59. The minimum orthogonal quality for all types of cells should be more than 0.01, with an average value that is significantly higher. The maximum aspect ratio must be less than 20 to avoid divergence of solutions. The maximum skewness for a triangular/tetrahedral mesh in most flows should be below 0.95, with an average value that is significantly lower. A maximum value above 0.95 may lead to convergence difficulties. The metrics of the generated mesh is reasonably good to be adopted for the CFD model.

A high-quality grid is essential to develop an accurate result. The solution converges quickly having small residuals with a good quality grid. It is necessary to check the simulation result to be grid independent or not. Figure 2.6 shows the mesh independency and time independency check for the generated mesh. The standard method to find mesh independency is to increase the resolution (element) of the domain and repeat the simulation. If there is a change in results, then a fine grid is used and the simulation is repeated. This procedure is continued until the results are not affected considerably by the grid size. In the present study, the accuracy of results improved with fine grid size and with increasing the count of elements above 60000, the results did not improve considerably. The time independency check shows that considerable change in the output result is not observed with the decrease in the time step. Hence, to reduce the total simulation time the time step of 0.01 sec is used. The mesh parameters used are listed in Table 2.4.



**Figure 2.5** Schematic diagram showing 2D and 3D mesh elements



**Figure 2.6** (a) Mesh independency check and (b) Time independency check

**Table 2.4** Mesh parameters

<b>Mesh boundary conditions</b>	<b>Type</b>	<b>Value</b>
Number of elements	element	61000
Element	triangle / quad	-
Element size	-	0.21 $\mu\text{m}$
Growth rate	-	1.2
Transition ratio	-	0.272
Maximum inflation layer	-	2
Inflation growth rate	factor	1.5

## **2.7. SUMMARY**

Large number of practical problems can be solved with defined level of accuracy with the CFD. Many algorithms are available in the software which are designed for the specific class of problems. In CFD analysis process, it is very essential to select an appropriate algorithm which will give desired results. The accuracy of the results is mainly depending on the quality of the mesh. So, more efforts or skill is required to generate the mesh on the selected geometrical model.