3. Finite Difference Methods

3.1 Introduction

As was mentioned in Chap. 1, all conservation equations have similar structure and may be regarded as special cases of a generic transport equation, Eq. (1.26), (1.27) or (1.28). For this reason, we shall treat only a single, generic conservation equation in this and the following chapters. It will be used to demonstrate discretization methods for the terms which are common to all conservation equations (convection, diffusion, and sources). The special features of the Navier-Stokes equations and techniques for solving coupled non-linear problems will be introduced later. Also, for the time being, the unsteady term will be dropped so we consider only time-independent problems.

For simplicity, we shall use only Cartesian grids at this point. The equation we shall deal with is:

$$\frac{\partial (\rho u_j \phi)}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \Gamma \frac{\partial \phi}{\partial x_j} \right) + q_\phi \cdot (3.1)$$

We shall assume that $\rho$, $u_j$, $\Gamma$ and $q_\phi$ are known. This may not be the case because the velocity may not have been computed yet and the properties of the fluid may depend on the temperature and, if turbulence models are used, on the velocity field as well. As we shall see, the iterative schemes used to solve these equations treat $\phi$ as the only unknown; all other variables are fixed at their values determined on the previous iteration so regarding these as known is a reasonable approach.

The special features of non-orthogonal and unstructured grids will be discussed in Chap. 8. Furthermore, of the many possible discretization techniques, only a selected few which illustrate the main ideas will be described; others may be found in the literature cited.

3.2 Basic Concept

The first step in obtaining a numerical solution is to discretize the geometric domain – i.e. a numerical grid must be defined. In finite difference (FD)
discretization methods the grid is usually locally structured, i.e. each grid node may be considered the origin of a local coordinate system, whose axes coincide with grid lines. This also implies that two grid lines belonging to the same family, say \( \xi_1 \), do not intersect, and that any pair of grid lines belonging to different families, say \( \xi_1 = \text{const.} \) and \( \xi_2 = \text{const.} \), intersect only once. In three dimensions, three grid lines intersect at each node; none of these lines intersect each other at any other point. Figure 3.1 shows examples of one-dimensional (1D) and two-dimensional (2D) Cartesian grids used in FD methods.

![Example of 1D and 2D Cartesian grids](image)

**Fig. 3.1.** An example of a 1D (above) and 2D (below) Cartesian grid for FD methods (full symbols denote boundary nodes and open symbols denote computational nodes)

Each node is uniquely identified by a set of indices, which are the indices of the grid lines that intersect at it, \((i, j)\) in 2D and \((i, j, k)\) in 3D. The neighbor nodes are defined by increasing or reducing one of the indices by unity.

The generic scalar conservation equation in differential form, (3.1), serves as the starting point for FD methods. As it is linear in \( \phi \), it will be approximated by a system of linear algebraic equations, in which the variable values at the grid nodes are the unknowns. The solution of this system approximates the solution to the partial differential equation (PDE).

Each node thus has one unknown variable value associated with it and must provide one algebraic equation. The latter is a relation between the variable value at that node and those at some of the neighboring nodes. It is obtained by replacing each term of the PDE at the particular node by a finite-difference approximation. Of course, the numbers of equations and unknowns must be equal. At boundary nodes where variable values are given (Dirichlet conditions), no equation is needed. When the boundary conditions