

# Chapter 1

## Basic Mathematics

In the field of computational fluid dynamics, it is essential to understand the equations and the mathematics. This will be helpful especially if we are going to implement, reorder or manipulate equations within a software or toolbox. There are a lot of ways to represent equations and thus a brief collection of the most essential mathematics are given in this chapter. The beauty of mathematics are also described in [Greenshields \[2015\]](#), [Dantzig and Rappaz \[2009\]](#), [Jasak \[1996\]](#) and [Moukalled et al. \[2015\]](#).

In the field of numerical simulations we are dealing with **tensors**  $\mathbf{T}^n$  of rank  $n$ . A **tensor** stands for any kind of field like a scalar, a vector or the classical known tensor that represents a matrix (normally a 3 by 3 matrix) and is of rank two. To keep things clear we use the following definition:

Zero rank **tensor**  $\mathbf{T}^0 :=$  scalar  $a$

First rank **tensor**  $\mathbf{T}^1 :=$  vector  $\mathbf{a}$

Second rank **tensor**  $\mathbf{T}^2 :=$  tensor  $\mathbf{T}$  (matrix of 3x3)

Third rank **tensor**  $\mathbf{T}^3 :=$  tensor  $T_{ijk}$

A tensor that is of higher order than rank zero is **always** given in bold symbol. Tensors higher than second order are only needed during the derivation of the Reynolds-Stress equation.

## 1.1 Basic Rules of Derivatives

The governing conservation equations in fluid dynamics are partial differential equations. That's why we briefly summarize the rules that are needed when we deal with this kind of equations. Considering the sum of two quantities  $\phi$  and  $\chi$  that are derived respectively to  $\tau$ , we can split the derivative:

$$\frac{\partial(\phi + \chi)}{\partial\tau} = \frac{\partial\phi}{\partial\tau} + \frac{\partial\chi}{\partial\tau} . \quad (1.1)$$

If we have to derive the product of the two quantities, we need to use the **product rule** to split the term. In other words, we have to keep one quantity constant whereas we derive the other one:

$$\frac{\partial\phi\chi}{\partial\tau} = \chi\frac{\partial\phi}{\partial\tau} + \phi\frac{\partial\chi}{\partial\tau} . \quad (1.2)$$

A constant quantity  $C$  can be taken inside or outside of a derivative:

$$\frac{\partial C\phi\chi}{\partial\tau} = C\frac{\partial\phi\chi}{\partial\tau} . \quad (1.3)$$

## 1.2 Einsteins Summation Convention

For vector and tensor equations there are several options of notations. The longest but clearest notation is the Cartesian one. This notation can be abbreviated using the Einsteins summation convention. In general, we are using this convention mostly for vector and tensor quantities. Assuming the sum of derivatives of the arbitrary variable  $\phi_i$  (like the momentum) in  $x$ ,  $y$  and  $z$  direction, the Cartesian form is written as:

$$\frac{\partial\phi_x}{\partial x} + \frac{\partial\phi_y}{\partial y} + \frac{\partial\phi_z}{\partial z} . \quad (1.4)$$

To simplify this equation, we can use the Einsteins summation convention. Commonly we neglect the summation sign  $\sum$  to keep things clear and short:

$$\sum_i \frac{\partial\phi_i}{\partial x_i} = \frac{\partial\phi_i}{\partial x_i} \quad i = x, y, z . \quad (1.5)$$

A more complex example that demonstrates the advantage of the Einsteins summation convention would be the convective term of the momentum equation (till now we do not need to know what this equation means and hence, we do not think

about the meaning). Due to the fact that the momentum is a vector quantity, we get three equations:

$$\frac{\partial u_x u_x}{\partial x} + \frac{\partial u_y u_x}{\partial y} + \frac{\partial u_z u_x}{\partial z} , \quad (1.6)$$

$$\frac{\partial u_x u_y}{\partial x} + \frac{\partial u_y u_y}{\partial y} + \frac{\partial u_z u_y}{\partial z} , \quad (1.7)$$

$$\frac{\partial u_x u_z}{\partial x} + \frac{\partial u_y u_z}{\partial y} + \frac{\partial u_z u_z}{\partial z} . \quad (1.8)$$

By using the Einsteins convention we can simplify the three equations into one:

$$\sum_i \frac{\partial u_i u_j}{\partial x_i} = \frac{\partial u_i u_j}{\partial x_i} \quad i = x, y, z; \ j = x, y, z . \quad (1.9)$$

The Einsteins summation convention is widely used in literatures. Hence, we should keep in mind what it stand for and how we have to apply it.

## 1.3 General Tensor Mathematics

A common and easy way to deal with equations is using the vector notation instead of the Einsteins summation convention. The vector notation require knowledge of special mathematics. We will familiarize different operations that act on scalars, vectors and tensors. For that purpose we introduce a scalar  $\phi$ , two vectors **a** and **b** and a tensor **T**:

$$\mathbf{a} = \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} , \quad \mathbf{b} = \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} ,$$

$$\mathbf{T} = \begin{bmatrix} T_{xx} & T_{xy} & T_{xz} \\ T_{yx} & T_{yy} & T_{yz} \\ T_{zx} & T_{zy} & T_{zz} \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} .$$

Depending on the operation we are investigating, we use either the numeric indices (1, 2, 3) or the space components ( $x, y, z$ ). Furthermore, we need the unit vectors

$e_i$  and the identity matrix  $\mathbf{I}$ :

$$e_1 = e_x = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad e_2 = e_y = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad e_3 = e_z = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad \mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

## Simple Operations

- The multiplication of a scalar  $\phi$  by a vector  $\mathbf{b}$  results in a vector and is commutative and associative. This is also valid for the multiplication of a scalar  $\phi$  and a tensor  $\mathbf{T}$ :

$$\phi\mathbf{b} = \begin{pmatrix} \phi b_x \\ \phi b_y \\ \phi b_z \end{pmatrix}, \quad \phi\mathbf{T} = \begin{bmatrix} \phi T_{xx} & \phi T_{xy} & \phi T_{xz} \\ \phi T_{yx} & \phi T_{yy} & \phi T_{yz} \\ \phi T_{zx} & \phi T_{zy} & \phi T_{zz} \end{bmatrix}. \quad (1.10)$$

## The Inner Product

- The inner product of two vectors  $\mathbf{a}$  and  $\mathbf{b}$  produces a scalar  $\phi$  and is commutative. This operation is indicated by the dot sign  $\bullet$ :

$$\phi = \mathbf{a} \bullet \mathbf{b} = \mathbf{a}^T \mathbf{b} = \sum_{i=1}^3 a_i b_i. \quad (1.11)$$

- The inner product of a vector  $\mathbf{a}$  and a tensor  $\mathbf{T}$  produces a vector  $\mathbf{b}$  and is non-commutative if the tensor is non-symmetric:

$$\mathbf{b} = \mathbf{T} \bullet \mathbf{a} = \sum_{i=1}^3 \sum_{j=1}^3 T_{ij} a_j e_i = \begin{pmatrix} T_{11}a_1 + T_{12}a_2 + T_{13}a_3 \\ T_{21}a_1 + T_{22}a_2 + T_{23}a_3 \\ T_{31}a_1 + T_{32}a_2 + T_{33}a_3 \end{pmatrix}. \quad (1.12)$$

$$\mathbf{b} = \mathbf{a} \bullet \mathbf{T} = \mathbf{T}^T \bullet \mathbf{a} = \sum_{i=1}^3 \sum_{j=1}^3 a_j T_{ji} e_i = \begin{pmatrix} a_1 T_{11} + a_2 T_{21} + a_3 T_{31} \\ a_1 T_{12} + a_2 T_{22} + a_3 T_{32} \\ a_1 T_{13} + a_2 T_{23} + a_3 T_{33} \end{pmatrix}, \quad (1.13)$$

A symmetric tensor is given, if  $\mathbf{T}_{ij} = \mathbf{T}_{ji}$  and hence,  $\mathbf{a} \bullet \mathbf{T} = \mathbf{T} \bullet \mathbf{a}$ .

## The Double Inner Product

- The double inner product of two tensors  $\mathbf{T}$  and  $\mathbf{S}$  results in a scalar  $\phi$  and is commutative. It will be indicated by the colon : sign:

$$\begin{aligned}\phi = \mathbf{T} : \mathbf{S} &= \sum_{i=1}^3 \sum_{j=1}^3 T_{ij} S_{ij} = T_{11}S_{11} + T_{12}S_{12} + T_{13}S_{13} + T_{21}S_{21} \\ &\quad + T_{22}S_{22} + T_{23}S_{23} + T_{31}S_{31} + T_{32}S_{32} + T_{33}S_{33} .\end{aligned}\quad (1.14)$$

### The Outer Product

- The outer product of two vectors  $\mathbf{a}$  and  $\mathbf{b}$ , also known as dyadic product, results in a tensor, is non-commutative and is expressed by the dyadic sign  $\otimes$ :

$$\mathbf{T} = \mathbf{a} \otimes \mathbf{b} = \mathbf{ab}^T = \begin{bmatrix} a_x b_x & a_x b_y & a_x b_z \\ a_y b_x & a_y b_y & a_y b_z \\ a_z b_x & a_z b_y & a_z b_z \end{bmatrix} . \quad (1.15)$$

In most of the literatures the dyadic sign  $\otimes$  is neglected for brevity as shown below:

$$\mathbf{ab} . \quad (1.16)$$

Keep in mind, that both variants are used in literature whereas the last one is more common but the first one is more clear. In this book we use the definition of equation (1.15), to be more consistent with the mathematics.

### Differential Operators

In vector notation, the spatial derivatives of a variable (scalar, vector or tensor) is made using the Nabla operator  $\nabla$ . It contains the three space derivatives of  $x, y$  and  $z$  in a Cartesian coordinate system:

$$\nabla = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{pmatrix} .$$

### Gradient Operator

- The gradient of a scalar  $\phi$  results in a vector  $\mathbf{a}$ :

$$\text{grad } \phi = \nabla \phi = \begin{pmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial y} \\ \frac{\partial \phi}{\partial z} \end{pmatrix} . \quad (1.17)$$

- The gradient of a vector  $\mathbf{b}$  results in a tensor  $\mathbf{T}$ :

$$\text{grad } \mathbf{b} = \nabla \otimes \mathbf{b} = \begin{bmatrix} \frac{\partial}{\partial x} b_x & \frac{\partial}{\partial x} b_y & \frac{\partial}{\partial x} b_z \\ \frac{\partial}{\partial y} b_x & \frac{\partial}{\partial y} b_y & \frac{\partial}{\partial y} b_z \\ \frac{\partial}{\partial z} b_x & \frac{\partial}{\partial z} b_y & \frac{\partial}{\partial z} b_z \end{bmatrix}. \quad (1.18)$$

We see that this operation is actually the outer product of the Nabla operator (special vector) and an arbitrary vector  $\mathbf{b}$ . Hence, it is commonly written as:

$$\nabla \mathbf{b} . \quad (1.19)$$

In this book we use the first notation (with the dyadic sign) to be more consistent within the mathematics.

**Note:** The gradient operation increase the rank of the **tensor** by one and hence, we can apply it to any **tensor** field.

## Divergence Operator

- The divergence of a vector  $\mathbf{b}$  results in a scalar  $\phi$  and is expressed by the combination of the Nabla operator and the dot sign,  $\nabla \bullet$ :

$$\text{div } \mathbf{b} = \nabla \bullet \mathbf{b} = \sum_{i=1}^3 \frac{\partial}{\partial x_i} b_i = \frac{\partial b_1}{\partial x_1} + \frac{\partial b_2}{\partial x_2} + \frac{\partial b_3}{\partial x_3} . \quad (1.20)$$

- The divergence of a tensor  $\mathbf{T}$  results in a vector  $\mathbf{b}$ :

$$\text{div } \mathbf{T} = \nabla \bullet \mathbf{T} = \sum_{i=1}^3 \sum_{j=1}^3 \frac{\partial}{\partial x_j} T_{ji} e_i = \begin{bmatrix} \frac{\partial T_{11}}{\partial x_1} + \frac{\partial T_{21}}{\partial x_2} + \frac{\partial T_{31}}{\partial x_3} \\ \frac{\partial T_{12}}{\partial x_1} + \frac{\partial T_{22}}{\partial x_2} + \frac{\partial T_{32}}{\partial x_3} \\ \frac{\partial T_{13}}{\partial x_1} + \frac{\partial T_{23}}{\partial x_2} + \frac{\partial T_{33}}{\partial x_3} \end{bmatrix} . \quad (1.21)$$

**Note:** The divergence operation decrease the rank of the **tensor** by one. Hence, it does not make sense to apply this operator on a **scalar**.

## The Product Rule within the Divergence Operator

If we have a product within a divergence term, we can split the term using the product rule. Based on the **tensor** ranks inside the divergence, we have to apply different rules, which are given now.

- The divergence of the product of a vector  $\mathbf{a}$  and a scalar  $\phi$  can be split as follow and results in a scalar:

$$\nabla \bullet (\mathbf{a}\phi) = \underbrace{\mathbf{a} \bullet \nabla\phi}_{\text{Eqn. (1.11)}} + \underbrace{\phi \nabla \bullet \mathbf{a}}_{\text{simple multiplication}} . \quad (1.22)$$

- The divergence of the outer product (dyadic product) of two vectors  $\mathbf{a}$  and  $\mathbf{b}$  can be split as follow and results in a vector:

$$\nabla \bullet (\mathbf{a} \otimes \mathbf{b}) = \underbrace{\mathbf{a} \bullet \nabla \otimes \mathbf{b}}_{\text{Eqn. (1.13)}} + \underbrace{\mathbf{b} \nabla \bullet \mathbf{a}}_{\text{Eqn. (1.10)}} . \quad (1.23)$$

- The divergence of the inner product of a tensor  $\mathbf{T}$  and a vector  $\mathbf{b}$  can be split as follow and results in a scalar:

$$\nabla \bullet (\mathbf{T} \bullet \mathbf{b}) = \underbrace{\mathbf{T} : \nabla \otimes \mathbf{b}}_{\text{Eqn. (1.14)}} + \underbrace{\mathbf{b} \bullet \nabla \bullet \mathbf{T}}_{\text{Eqn. (1.11)}} . \quad (1.24)$$

If you think that the product rule for the inner product of two vectors is missing, just think about the result of the inner product of two vectors and how this **tensor** will change (rank) if we apply the divergence operator. Hopefully you will figure out, that the divergence of a scalar does not make sense.

### 1.3.1 The Total Derivative

The definition of the total derivative of an arbitrary quantity  $\phi$  – in the field of fluid dynamics – is defined as:

$$\frac{D\phi}{Dt} = \frac{\partial\phi}{\partial t} + \underbrace{\mathbf{U} \bullet \nabla\phi}_{\text{inner product}} , \quad (1.25)$$

where  $\mathbf{U}$  represents the velocity vector. The last term in equation (1.25) denotes the inner product. Depending on the quantity  $\phi$  (scalar, vector, tensor, and so on), we have to use the correct mathematical expressions for the second term on the right hand side. Example given:

- If  $\phi$  is a scalar, we have to use equation (1.11),
- If  $\phi$  is a vector, we have to use equation (1.13).

## Short Outline for the Total Derivative

The total derivative is used to represent non-conserved equations. In other words, each conserved equation can be changed into a non-conserved equation using the continuity equation. In literature people start to derive equations using the total derivative and using the continuity equation to extend the non-conservative equation to the conserved one. The better way would be to derive *first* the conserved equation and *then* using the continuity equation to get the non-conserved form. *Why?* It is easier to understand. The difference between both equations is the frame of reference. In the conserved representation, we have the Euler expression, for non-conserved equations it is the Lagrange expression.

If you have literature that start with the non-conservation equations, this would help to understand the following extension (at the moment we do not need to understand this equations, it is just an example):

- Incompressible:

$$\frac{D\phi}{Dt} = \underbrace{\frac{\partial\phi}{\partial t} + \mathbf{U} \bullet \nabla\phi}_{\text{non-conserved}} + \underbrace{\phi (\nabla \bullet (\mathbf{U}))}_{\text{continuity} = 0}. \quad (1.26)$$

- Compressible:

$$\rho \frac{D\phi}{Dt} = \rho \underbrace{\left[ \frac{\partial\phi}{\partial t} + \mathbf{U} \bullet \nabla\phi \right]}_{\text{non-conserved}} + \underbrace{\phi \left( \frac{\partial\rho}{\partial t} + \nabla \bullet (\rho\mathbf{U}) \right)}_{\text{continuity} = 0}. \quad (1.27)$$

The reason we multiply the continuity equation (second term on the right hand side) by the quantity  $\phi$  comes from the product rule, that is applied to the convective term. After the momentum equation is derived and the conservative form is transformed into the non-conserved one, this will get clear.

### 1.3.2 Matrix Algebra, Deviatoric and Hydrostatic Part

In the field of numerical simulations we are dealing with quantities that are represented by matrices like the stress tensor. That's why we need to introduce some basics here. Each matrix  $\mathbf{A}$  can be split into a deviatoric  $\mathbf{A}^{\text{dev}}$  and hydrostatic  $\mathbf{A}^{\text{hyd}}$  part:

$$\mathbf{A} = \mathbf{A}^{\text{hyd}} + \mathbf{A}^{\text{dev}}. \quad (1.28)$$

The hydrostatic part can be expressed as scalar or matrix and is defined by using the trace operator. If we want to get the scalar, we use the following definition:

$$A^{\text{hyd}} = \frac{1}{3} \text{tr}(\mathbf{A}) = \frac{1}{3} \sum_{i=1}^n (a_{ii}) . \quad (1.29)$$

The operator  $\text{tr}$  denotes the trace operator and is applied on the matrix. This operator is simply the sum of the diagonal elements. The matrix notation of the hydrostatic part given by:

$$\mathbf{A}^{\text{hyd}} = A^{\text{hyd}} \mathbf{I} = \frac{1}{3} \text{tr}(\mathbf{A}) \mathbf{I} = \frac{1}{3} \sum_{i=1}^n (a_{ii}) \mathbf{I} . \quad (1.30)$$

The deviatoric part  $\mathbf{A}^{\text{dev}}$  is given as:

$$\mathbf{A}^{\text{dev}} = \mathbf{A} - \mathbf{A}^{\text{hyd}} = \mathbf{A} - \frac{1}{3} \text{tr}(\mathbf{A}) \mathbf{I} . \quad (1.31)$$

**Note:** The deviatoric part of a matrix is *traceless*. Hence,  $\text{tr}(\mathbf{A}^{\text{dev}}) = 0$ ; The trace operator is zero not the diagonal elements.

### 1.3.3 The Gauss Theorem

To transform any equation from the differential form to the integral one (or vice versa), it is necessary to know the Gauss theorem. This theorem allows us to establish a relation between the *fluxes through the surface* of an arbitrary volume element and the *divergence operator on the volume element*:

$$\oint \mathbf{a} \cdot \mathbf{n} dS = \int (\nabla \bullet \mathbf{a}) dV . \quad (1.32)$$

In equation (1.32),  $\mathbf{n}$  represents the surface normal vector pointing outwards,  $dS$  the integration with respect to the surface and  $dV$  the integration with respect to the volume.

**Note:** The small dot  $\bullet$  denotes the inner product of two vectors (1.11). In the following book, we use the small dot in all integrals to sign that we calculate the inner product of a vector  $\mathbf{a}$  and the *surface normal vector*  $\mathbf{n}$ . Keep in mind that the small dot expresses exact the same as the bullet.



# Chapter two Conservation laws of fluid motion and boundary conditions

In this chapter we develop the mathematical basis for a comprehensive general-purpose model of fluid flow and heat transfer from the basic principles of conservation of mass, momentum and energy. This leads to the governing equations of fluid flow and a discussion of the necessary auxiliary conditions – initial and boundary conditions. The main issues covered in this context are:

- Derivation of the system of partial differential equations (PDEs) that govern flows in Cartesian ( $x, y, z$ ) co-ordinates
- Thermodynamic equations of state
- Newtonian model of viscous stresses leading to the Navier–Stokes equations
- Commonalities between the governing PDEs and the definition of the transport equation
- Integrated forms of the transport equation over a finite time interval and a finite control volume
- Classification of physical behaviours into three categories: elliptic, parabolic and hyperbolic
- Appropriate boundary conditions for each category
- Classification of fluid flows
- Auxiliary conditions for viscous fluid flows
- Problems with boundary condition specification in high Reynolds number and high Mach number flows

## 2.1

### Governing equations of fluid flow and heat transfer

The governing equations of fluid flow represent mathematical statements of the **conservation laws of physics**:

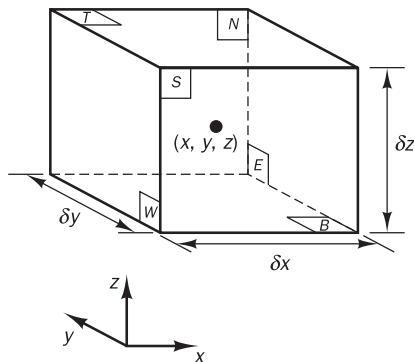
- The mass of a fluid is conserved
- The rate of change of momentum equals the sum of the forces on a fluid particle (Newton's second law)
- The rate of change of energy is equal to the sum of the rate of heat addition to and the rate of work done on a fluid particle (first law of thermodynamics)

The fluid will be regarded as a continuum. For the analysis of fluid flows at macroscopic length scales (say  $1 \mu\text{m}$  and larger) the molecular structure of matter and molecular motions may be ignored. We describe the behaviour of the fluid in terms of macroscopic properties, such as velocity, pressure, density and temperature, and their space and time derivatives. These may

be thought of as averages over suitably large numbers of molecules. A fluid particle or point in a fluid is then the smallest possible element of fluid whose macroscopic properties are not influenced by individual molecules.

We consider such a small element of fluid with sides  $\delta x$ ,  $\delta y$  and  $\delta z$  (Figure 2.1).

**Figure 2.1** Fluid element for conservation laws



The six faces are labelled  $N$ ,  $S$ ,  $E$ ,  $W$ ,  $T$  and  $B$ , which stands for North, South, East, West, Top and Bottom. The positive directions along the co-ordinate axes are also given. The centre of the element is located at position  $(x, y, z)$ . A systematic account of changes in the mass, momentum and energy of the fluid element due to fluid flow across its boundaries and, where appropriate, due to the action of sources inside the element, leads to the fluid flow equations.

All fluid properties are functions of space and time so we would strictly need to write  $\rho(x, y, z, t)$ ,  $p(x, y, z, t)$ ,  $T(x, y, z, t)$  and  $\mathbf{u}(x, y, z, t)$  for the density, pressure, temperature and the velocity vector respectively. To avoid unduly cumbersome notation we will not explicitly state the dependence on space co-ordinates and time. For instance, the density at the centre  $(x, y, z)$  of a fluid element at time  $t$  is denoted by  $\rho$  and the  $x$ -derivative of, say, pressure  $p$  at  $(x, y, z)$  and time  $t$  by  $\partial p / \partial x$ . This practice will also be followed for all other fluid properties.

The element under consideration is so small that fluid properties at the faces can be expressed accurately enough by means of the first two terms of a Taylor series expansion. So, for example, the pressure at the  $W$  and  $E$  faces, which are both at a distance of  $\frac{1}{2}\delta x$  from the element centre, can be expressed as

$$p - \frac{\partial p}{\partial x} \frac{1}{2} \delta x \quad \text{and} \quad p + \frac{\partial p}{\partial x} \frac{1}{2} \delta x$$

### 2.1.1 Mass conservation in three dimensions

The first step in the derivation of the mass conservation equation is to write down a mass balance for the fluid element:

Rate of increase of mass in fluid element	= Net rate of flow of mass into fluid element
---	---

The rate of increase of mass in the fluid element is

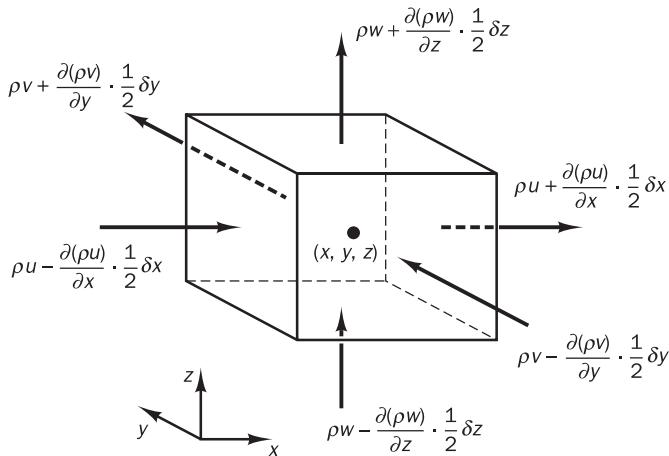
$$\frac{\partial}{\partial t}(\rho \delta x \delta y \delta z) = \frac{\partial \rho}{\partial t} \delta x \delta y \delta z \quad (2.1)$$

Next we need to account for the mass flow rate across a face of the element, which is given by the product of density, area and the velocity component normal to the face. From Figure 2.2 it can be seen that the net rate of flow of mass into the element across its boundaries is given by

$$\begin{aligned} & \left( \rho u - \frac{\partial(\rho u)}{\partial x} \frac{1}{2} \delta x \right) \delta y \delta z - \left( \rho u + \frac{\partial(\rho u)}{\partial x} \frac{1}{2} \delta x \right) \delta y \delta z \\ & + \left( \rho v - \frac{\partial(\rho v)}{\partial y} \frac{1}{2} \delta y \right) \delta x \delta z - \left( \rho v + \frac{\partial(\rho v)}{\partial y} \frac{1}{2} \delta y \right) \delta x \delta z \\ & + \left( \rho w - \frac{\partial(\rho w)}{\partial z} \frac{1}{2} \delta z \right) \delta x \delta y - \left( \rho w + \frac{\partial(\rho w)}{\partial z} \frac{1}{2} \delta z \right) \delta x \delta y \end{aligned} \quad (2.2)$$

Flows which are directed into the element produce an increase of mass in the element and get a positive sign and those flows that are leaving the element are given a negative sign.

**Figure 2.2** Mass flows in and out of fluid element



The rate of increase of mass inside the element (2.1) is now equated to the net rate of flow of mass into the element across its faces (2.2). All terms of the resulting mass balance are arranged on the left hand side of the equals sign and the expression is divided by the element volume  $\delta x \delta y \delta z$ . This yields

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad (2.3)$$

or in more compact vector notation

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) = 0$$

(2.4)

Equation (2.4) is the **unsteady, three-dimensional mass conservation or continuity equation** at a point in a **compressible fluid**. The first term

on the left hand side is the rate of change in time of the density (mass per unit volume). The second term describes the net flow of mass out of the element across its boundaries and is called the convective term.

For an **incompressible fluid** (i.e. a liquid) the density  $\rho$  is constant and equation (2.4) becomes

$$\operatorname{div} \mathbf{u} = 0 \quad (2.5)$$

or in longhand notation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (2.6)$$

### 2.1.2 Rates of change following a fluid particle and for a fluid element

The momentum and energy conservation laws make statements regarding changes of properties of a fluid particle. This is termed the Lagrangian approach. Each property of such a particle is a function of the position  $(x, y, z)$  of the particle and time  $t$ . Let the value of a property per unit mass be denoted by  $\phi$ . The total or substantive derivative of  $\phi$  with respect to time following a fluid particle, written as  $D\phi/Dt$ , is

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial x} \frac{dx}{dt} + \frac{\partial \phi}{\partial y} \frac{dy}{dt} + \frac{\partial \phi}{\partial z} \frac{dz}{dt}$$

A fluid particle follows the flow, so  $dx/dt = u$ ,  $dy/dt = v$  and  $dz/dt = w$ . Hence the substantive derivative of  $\phi$  is given by

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + u \frac{\partial \phi}{\partial x} + v \frac{\partial \phi}{\partial y} + w \frac{\partial \phi}{\partial z} = \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \operatorname{grad} \phi \quad (2.7)$$

$D\phi/Dt$  defines rate of change of property  $\phi$  per unit mass. It is possible to develop numerical methods for fluid flow calculations based on the Lagrangian approach, i.e. by tracking the motion and computing the rates of change of conserved properties  $\phi$  for collections of fluid particles. However, it is far more common to develop equations for collections of fluid elements making up a region fixed in space, for example a region defined by a duct, a pump, a furnace or similar piece of engineering equipment. This is termed the Eulerian approach.

As in the case of the mass conservation equation, we are interested in developing equations for rates of change per unit volume. The rate of change of property  $\phi$  per unit volume for a fluid particle is given by the product of  $D\phi/Dt$  and density  $\rho$ , hence

$$\rho \frac{D\phi}{Dt} = \rho \left( \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \operatorname{grad} \phi \right) \quad (2.8)$$

The most useful forms of the conservation laws for fluid flow computation are concerned with changes of a flow property for a fluid element that is stationary in space. The relationship between the substantive derivative of  $\phi$ , which follows a fluid particle, and rate of change of  $\phi$  for a fluid element is now developed.

The mass conservation equation contains the mass per unit volume (i.e. the density  $\rho$ ) as the conserved quantity. The sum of the rate of change of density in time and the convective term in the mass conservation equation (2.4) for a fluid element is

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u})$$

The generalisation of these terms for an arbitrary conserved property is

$$\frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) \quad (2.9)$$

Formula (2.9) expresses the rate of change in time of  $\phi$  per unit volume plus the net flow of  $\phi$  out of the fluid element per unit volume. It is now rewritten to illustrate its relationship with the substantive derivative of  $\phi$ :

$$\begin{aligned} \frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) &= \rho \left[ \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \operatorname{grad} \phi \right] + \phi \left[ \frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) \right] \\ &= \rho \frac{D\phi}{Dt} \end{aligned} \quad (2.10)$$

The term  $\phi[(\partial\rho/\partial t) + \operatorname{div}(\rho\mathbf{u})]$  is equal to zero by virtue of mass conservation (2.4). In words, relationship (2.10) states

Rate of increase of $\phi$ of fluid element	$+ \quad$	Net rate of flow of $\phi$ out of fluid element	$= \quad$	Rate of increase of $\phi$ for a fluid particle
---	-----------	---	-----------	---

To construct the three components of the momentum equation and the energy equation the relevant entries for  $\phi$  and their rates of change per unit volume as defined in (2.8) and (2.10) are given below:

$x$ -momentum	$u$	$\rho \frac{Du}{Dt}$	$\frac{\partial(\rho u)}{\partial t} + \operatorname{div}(\rho u \mathbf{u})$
$y$ -momentum	$v$	$\rho \frac{Dv}{Dt}$	$\frac{\partial(\rho v)}{\partial t} + \operatorname{div}(\rho v \mathbf{u})$
$z$ -momentum	$w$	$\rho \frac{Dw}{Dt}$	$\frac{\partial(\rho w)}{\partial t} + \operatorname{div}(\rho w \mathbf{u})$
energy	$E$	$\rho \frac{DE}{Dt}$	$\frac{\partial(\rho E)}{\partial t} + \operatorname{div}(\rho E \mathbf{u})$

Both the conservative (or divergence) form and non-conservative form of the rate of change can be used as alternatives to express the conservation of a physical quantity. The non-conservative forms are used in the derivations of momentum and energy equations for a fluid flow in sections 2.4 and 2.5 for brevity of notation and to emphasise that the conservation laws are fundamentally conceived as statements that apply to a particle of fluid. In the final

section 2.8 we will return to the conservative form that is used in finite volume CFD calculations.

### 2.1.3 Momentum equation in three dimensions

**Newton's second law** states that the rate of change of momentum of a fluid particle equals the sum of the forces on the particle:

Rate of increase of momentum of fluid particle	=	Sum of forces on fluid particle
--	---	---------------------------------------

The **rates of increase of  $x$ -,  $y$ - and  $z$ -momentum** per unit volume of a fluid particle are given by

$$\rho \frac{Du}{Dt} \quad \rho \frac{Dv}{Dt} \quad \rho \frac{Dw}{Dt} \quad (2.11)$$

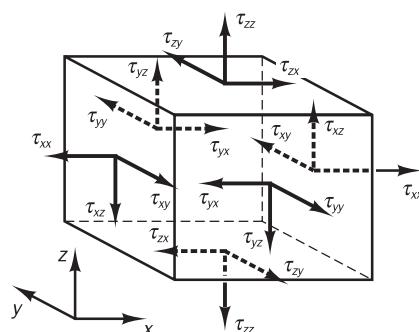
We distinguish two types of **forces** on fluid particles:

- **surface forces**
  - pressure forces
  - viscous forces
  - gravity force
- **body forces**
  - centrifugal force
  - Coriolis force
  - electromagnetic force

It is common practice to highlight the contributions due to the surface forces as separate terms in the momentum equation and to include the effects of body forces as source terms.

The state of stress of a fluid element is defined in terms of the pressure and the nine viscous stress components shown in Figure 2.3. The pressure, a normal stress, is denoted by  $p$ . Viscous stresses are denoted by  $\tau$ . The usual suffix notation  $\tau_{ij}$  is applied to indicate the direction of the viscous stresses. The suffices  $i$  and  $j$  in  $\tau_{ij}$  indicate that the stress component acts in the  $j$ -direction on a surface normal to the  $i$ -direction.

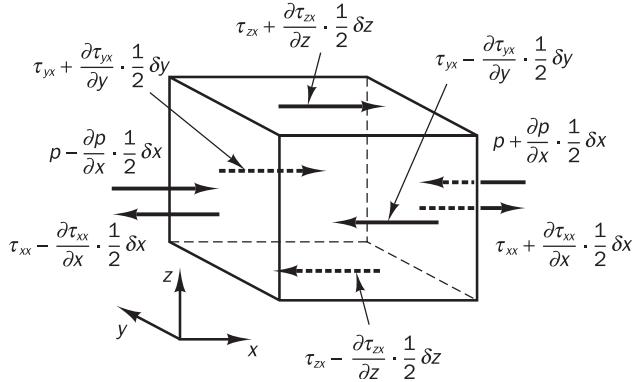
**Figure 2.3** Stress components on three faces of fluid element



First we consider the  $x$ -components of the forces due to pressure  $p$  and stress components  $\tau_{xx}$ ,  $\tau_{yx}$  and  $\tau_{zx}$  shown in Figure 2.4. The magnitude of a

force resulting from a surface stress is the product of stress and area. Forces aligned with the direction of a co-ordinate axis get a positive sign and those in the opposite direction a negative sign. The net force in the  $x$ -direction is the sum of the force components acting in that direction on the fluid element.

**Figure 2.4** Stress components in the  $x$ -direction



On the pair of faces ( $E, W$ ) we have

$$\left[ \left( p - \frac{\partial p}{\partial x} \frac{1}{2} \delta x \right) - \left( \tau_{xx} - \frac{\partial \tau_{xx}}{\partial x} \frac{1}{2} \delta x \right) \right] \delta y \delta z + \left[ - \left( p + \frac{\partial p}{\partial x} \frac{1}{2} \delta x \right) + \left( \tau_{xx} + \frac{\partial \tau_{xx}}{\partial x} \frac{1}{2} \delta x \right) \right] \delta y \delta z = \left( - \frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} \right) \delta x \delta y \delta z \quad (2.12a)$$

The net force in the  $x$ -direction on the pair of faces ( $N, S$ ) is

$$- \left( \tau_{yx} - \frac{\partial \tau_{yx}}{\partial y} \frac{1}{2} \delta y \right) \delta x \delta z + \left( \tau_{yx} + \frac{\partial \tau_{yx}}{\partial y} \frac{1}{2} \delta y \right) \delta x \delta z = \frac{\partial \tau_{yx}}{\partial y} \delta x \delta y \delta z \quad (2.12b)$$

Finally the net force in the  $x$ -direction on faces  $T$  and  $B$  is given by

$$- \left( \tau_{zx} - \frac{\partial \tau_{zx}}{\partial z} \frac{1}{2} \delta z \right) \delta x \delta y + \left( \tau_{zx} + \frac{\partial \tau_{zx}}{\partial z} \frac{1}{2} \delta z \right) \delta x \delta y = \frac{\partial \tau_{zx}}{\partial z} \delta x \delta y \delta z \quad (2.12c)$$

The total force per unit volume on the fluid due to these surface stresses is equal to the sum of (2.12a), (2.12b) and (2.12c) divided by the volume  $\delta x \delta y \delta z$ :

$$\frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \quad (2.13)$$

Without considering the body forces in further detail their overall effect can be included by defining a source  $S_{Mx}$  of  $x$ -momentum per unit volume per unit time.

The  **$x$ -component of the momentum equation** is found by setting the rate of change of  $x$ -momentum of the fluid particle (2.11) equal to the

total force in the  $x$ -direction on the element due to surface stresses (2.13) plus the rate of increase of  $x$ -momentum due to sources:

$$\rho \frac{Du}{Dt} = \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + S_{Mx} \quad (2.14a)$$

It is not too difficult to verify that the  **$y$ -component of the momentum equation** is given by

$$\rho \frac{Dv}{Dt} = \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + S_{My} \quad (2.14b)$$

and the  **$z$ -component of the momentum equation** by

$$\rho \frac{Dw}{Dt} = \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial(-p + \tau_{zz})}{\partial z} + S_{Mz} \quad (2.14c)$$

The sign associated with the pressure is opposite to that associated with the normal viscous stress, because the usual sign convention takes a tensile stress to be the positive normal stress so that the pressure, which is by definition a compressive normal stress, has a minus sign.

The effects of surface stresses are accounted for explicitly; the source terms  $S_{Mx}$ ,  $S_{My}$  and  $S_{Mz}$  in (2.14a–c) include contributions due to body forces only. For example, the body force due to gravity would be modelled by  $S_{Mx} = 0$ ,  $S_{My} = 0$  and  $S_{Mz} = -\rho g$ .

#### 2.1.4 Energy equation in three dimensions

The energy equation is derived from the **first law of thermodynamics**, which states that the rate of change of energy of a fluid particle is equal to the rate of heat addition to the fluid particle plus the rate of work done on the particle:

Rate of increase of energy of fluid particle	=	Net rate of heat added to fluid particle	+	Net rate of work done on fluid particle
--	---	--	---	---

As before, we will be deriving an equation for the **rate of increase of energy** of a fluid particle per unit volume, which is given by

$$\rho \frac{DE}{Dt} \quad (2.15)$$

##### Work done by surface forces

The **rate of work done** on the fluid particle in the element by a **surface force** is equal to the product of the force and velocity component in the direction of the force. For example, the forces given by (2.12a–c) all act in the  $x$ -direction. The work done by these forces is given by

$$\begin{aligned}
& \left[ \left( pu - \frac{\partial(pu)}{\partial x} \frac{1}{2} \delta x \right) - \left( \tau_{xx} u - \frac{\partial(\tau_{xx} u)}{\partial x} \frac{1}{2} \delta x \right) \right. \\
& \quad \left. - \left( pu + \frac{\partial(pu)}{\partial x} \frac{1}{2} \delta x \right) + \left( \tau_{xx} u + \frac{\partial(\tau_{xx} u)}{\partial x} \frac{1}{2} \delta x \right) \right] \delta y \delta z \\
& \quad + \left[ - \left( \tau_{yx} u - \frac{\partial(\tau_{yx} u)}{\partial y} \frac{1}{2} \delta y \right) + \left( \tau_{yx} u + \frac{\partial(\tau_{yx} u)}{\partial y} \frac{1}{2} \delta y \right) \right] \delta x \delta z \\
& \quad + \left[ - \left( \tau_{zx} u - \frac{\partial(\tau_{zx} u)}{\partial z} \frac{1}{2} \delta z \right) + \left( \tau_{zx} u + \frac{\partial(\tau_{zx} u)}{\partial z} \frac{1}{2} \delta z \right) \right] \delta x \delta y
\end{aligned}$$

The net rate of work done by these surface forces acting in the  $x$ -direction is given by

$$\left[ \frac{\partial(u(-p + \tau_{xx}))}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} \right] \delta x \delta y \delta z \quad (2.16a)$$

Surface stress components in the  $y$ - and  $z$ -direction also do work on the fluid particle. A repetition of the above process gives the additional rates of work done on the fluid particle due to the work done by these surface forces:

$$\left[ \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v(-p + \tau_{yy}))}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} \right] \delta x \delta y \delta z \quad (2.16b)$$

and

$$\left[ \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w(-p + \tau_{zz}))}{\partial z} \right] \delta x \delta y \delta z \quad (2.16c)$$

The total rate of work done per unit volume on the fluid particle by all the surface forces is given by the sum of (2.16a–c) divided by the volume  $\delta x \delta y \delta z$ . The terms containing pressure can be collected together and written more compactly in vector form

$$-\frac{\partial(up)}{\partial x} - \frac{\partial(vp)}{\partial y} - \frac{\partial(wp)}{\partial z} = -\operatorname{div}(p\mathbf{u})$$

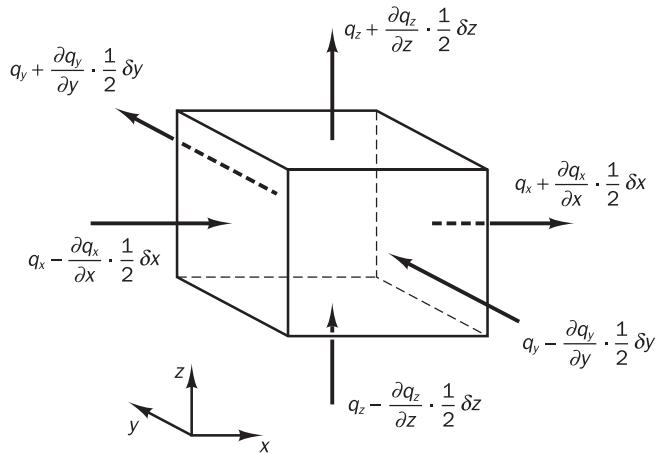
This yields the following **total rate of work done on the fluid particle by surface stresses**:

$$\begin{aligned}
& [-\operatorname{div}(p\mathbf{u})] + \left[ \frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} \right. \\
& \quad \left. + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z} \right]
\end{aligned} \quad (2.17)$$

### **Energy flux due to heat conduction**

The heat flux vector  $\mathbf{q}$  has three components:  $q_x$ ,  $q_y$  and  $q_z$  (Figure 2.5).

**Figure 2.5** Components of the heat flux vector



The **net rate of heat transfer to the fluid particle** due to heat flow in the  $x$ -direction is given by the difference between the rate of heat input across face  $W$  and the rate of heat loss across face  $E$ :

$$\left[ \left( q_x - \frac{\partial q_x}{\partial x} \frac{1}{2} \delta x \right) - \left( q_x + \frac{\partial q_x}{\partial x} \frac{1}{2} \delta x \right) \right] \delta y \delta z = -\frac{\partial q_x}{\partial x} \delta x \delta y \delta z \quad (2.18a)$$

Similarly, the net rates of heat transfer to the fluid due to heat flows in the  $y$ - and  $z$ -direction are

$$-\frac{\partial q_y}{\partial y} \delta x \delta y \delta z \quad \text{and} \quad -\frac{\partial q_z}{\partial z} \delta x \delta y \delta z \quad (2.18b-c)$$

The total rate of heat added to the fluid particle per unit volume due to heat flow across its boundaries is the sum of (2.18a–c) divided by the volume  $\delta x \delta y \delta z$ :

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} - \frac{\partial q_z}{\partial z} = -\operatorname{div} \mathbf{q} \quad (2.19)$$

Fourier's law of heat conduction relates the heat flux to the local temperature gradient. So

$$q_x = -k \frac{\partial T}{\partial x} \quad q_y = -k \frac{\partial T}{\partial y} \quad q_z = -k \frac{\partial T}{\partial z}$$

This can be written in vector form as follows:

$$\mathbf{q} = -k \operatorname{grad} T \quad (2.20)$$

Combining (2.19) and (2.20) yields the final form of the **rate of heat addition to the fluid particle due to heat conduction** across element boundaries:

$$-\operatorname{div} \mathbf{q} = \operatorname{div}(k \operatorname{grad} T) \quad (2.21)$$

### Energy equation

Thus far we have not defined the specific energy  $E$  of a fluid. Often the energy of a fluid is defined as the sum of internal (thermal) energy  $i$ , kinetic energy  $\frac{1}{2}(u^2 + v^2 + w^2)$  and gravitational potential energy. This definition

takes the view that the fluid element is storing gravitational potential energy. It is also possible to regard the gravitational force as a body force, which does work on the fluid element as it moves through the gravity field.

Here we will take the latter view and include the effects of potential energy changes as a source term. As before, we define a source of energy  $S_E$  per unit volume per unit time. Conservation of energy of the fluid particle is ensured by equating the rate of change of energy of the fluid particle (2.15) to the sum of the net rate of work done on the fluid particle (2.17), the net rate of heat addition to the fluid (2.21) and the rate of increase of energy due to sources. The **energy equation** is

$$\rho \frac{DE}{Dt} = -\text{div}(p\mathbf{u}) + \left[ \frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} \right. \\ \left. + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z} \right] \\ + \text{div}(k \text{ grad } T) + S_E \quad (2.22)$$

In equation (2.22) we have  $E = i + \frac{1}{2}(u^2 + v^2 + w^2)$ .

Although (2.22) is a perfectly adequate energy equation it is common practice to extract the changes of the (mechanical) kinetic energy to obtain an equation for internal energy  $i$  or temperature  $T$ . The part of the energy equation attributable to the kinetic energy can be found by multiplying the  $x$ -momentum equation (2.14a) by velocity component  $u$ , the  $y$ -momentum equation (2.14b) by  $v$  and the  $z$ -momentum equation (2.14c) by  $w$  and adding the results together. It can be shown that this yields the following conservation equation for the kinetic energy:

$$\rho \frac{D[\frac{1}{2}(u^2 + v^2 + w^2)]}{Dt} = -\mathbf{u} \cdot \text{grad } p + u \left( \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \right) \\ + v \left( \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \right) \\ + w \left( \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \right) + \mathbf{u} \cdot \mathbf{S}_M \quad (2.23)$$

Subtracting (2.23) from (2.22) and defining a new source term as  $S_i = S_E - \mathbf{u} \cdot \mathbf{S}_M$  yields the internal energy equation

$$\rho \frac{Di}{Dt} = -p \text{ div } \mathbf{u} + \text{div}(k \text{ grad } T) + \tau_{xx} \frac{\partial u}{\partial x} + \tau_{yx} \frac{\partial u}{\partial y} + \tau_{zx} \frac{\partial u}{\partial z} \\ + \tau_{xy} \frac{\partial v}{\partial x} + \tau_{yy} \frac{\partial v}{\partial y} + \tau_{zy} \frac{\partial v}{\partial z} \\ + \tau_{xz} \frac{\partial w}{\partial x} + \tau_{yz} \frac{\partial w}{\partial y} + \tau_{zz} \frac{\partial w}{\partial z} + S_i \quad (2.24)$$

For the special case of an incompressible fluid we have  $i = cT$ , where  $c$  is the specific heat and  $\operatorname{div} \mathbf{u} = 0$ . This allows us to recast (2.24) into a temperature equation

$$\begin{aligned} \rho c \frac{DT}{Dt} &= \operatorname{div}(k \operatorname{grad} T) + \tau_{xx} \frac{\partial u}{\partial x} + \tau_{yx} \frac{\partial u}{\partial y} + \tau_{zx} \frac{\partial u}{\partial z} + \tau_{xy} \frac{\partial v}{\partial x} \\ &\quad + \tau_{yy} \frac{\partial v}{\partial y} + \tau_{zy} \frac{\partial v}{\partial z} + \tau_{xz} \frac{\partial w}{\partial x} + \tau_{yz} \frac{\partial w}{\partial y} + \tau_{zz} \frac{\partial w}{\partial z} + S_i \end{aligned} \quad (2.25)$$

For compressible flows equation (2.22) is often rearranged to give an equation for the **enthalpy**. The specific enthalpy  $h$  and the specific total enthalpy  $h_0$  of a fluid are defined as

$$h = i + p/\rho \quad \text{and} \quad h_0 = h + \frac{1}{2}(u^2 + v^2 + w^2)$$

Combining these two definitions with the one for specific energy  $E$  we get

$$h_0 = i + p/\rho + \frac{1}{2}(u^2 + v^2 + w^2) = E + p/\rho \quad (2.26)$$

Substitution of (2.26) into (2.22) and some rearrangement yields the (**total enthalpy equation**)

$$\begin{aligned} \frac{\partial(\rho h_0)}{\partial t} + \operatorname{div}(\rho h_0 \mathbf{u}) &= \operatorname{div}(k \operatorname{grad} T) + \frac{\partial p}{\partial t} \\ &\quad + \left[ \frac{\partial(u \tau_{xx})}{\partial x} + \frac{\partial(u \tau_{yx})}{\partial y} + \frac{\partial(u \tau_{zx})}{\partial z} \right. \\ &\quad + \frac{\partial(v \tau_{xy})}{\partial x} + \frac{\partial(v \tau_{yy})}{\partial y} + \frac{\partial(v \tau_{zy})}{\partial z} \\ &\quad \left. + \frac{\partial(w \tau_{xz})}{\partial x} + \frac{\partial(w \tau_{yz})}{\partial y} + \frac{\partial(w \tau_{zz})}{\partial z} \right] + S_h \end{aligned} \quad (2.27)$$

It should be stressed that equations (2.24), (2.25) and (2.27) are *not* new (extra) conservation laws but merely alternative forms of the energy equation (2.22).

## 2.2

### Equations of state

The motion of a fluid in three dimensions is described by a system of five partial differential equations: mass conservation (2.4),  $x$ -,  $y$ - and  $z$ -momentum equations (2.14a–c) and energy equation (2.22). Among the unknowns are four thermodynamic variables:  $\rho$ ,  $p$ ,  $i$  and  $T$ . In this brief discussion we point out the linkage between these four variables. Relationships between the thermodynamic variables can be obtained through the assumption of **thermodynamic equilibrium**. The fluid velocities may be large, but they are usually small enough that, even though properties of a fluid particle change rapidly from place to place, the fluid can thermodynamically adjust itself to new conditions so quickly that the changes are effectively instantaneous. Thus the fluid always remains in thermodynamic equilibrium. The only exceptions are certain flows with strong shockwaves, but even some of those are often well enough approximated by equilibrium assumptions.

We can describe the state of a substance in thermodynamic equilibrium by means of just two state variables. **Equations of state** relate the other variables to the two state variables. If we use  $\rho$  and  $T$  as state variables we have state equations for pressure  $p$  and specific internal energy  $i$ :

$$p = p(\rho, T) \quad \text{and} \quad i = i(\rho, T) \quad (2.28)$$

For a **perfect gas** the following, well-known, equations of state are useful:

$$p = \rho RT \quad \text{and} \quad i = C_v T \quad (2.29)$$

The assumption of thermodynamic equilibrium eliminates all but the two thermodynamic state variables. In the flow of **compressible fluids** the equations of state provide the linkage between the energy equation on the one hand and mass conservation and momentum equations on the other. This linkage arises through the possibility of density variations as a result of pressure and temperature variations in the flow field.

Liquids and gases flowing at low speeds behave as **incompressible fluids**. Without density variations there is no linkage between the energy equation and the mass conservation and momentum equations. The flow field can often be solved by considering mass conservation and momentum equations only. The energy equation only needs to be solved alongside the others if the problem involves heat transfer.

## 2.3

### Navier–Stokes equations for a Newtonian fluid

The governing equations contain as further unknowns the viscous stress components  $\tau_{ij}$ . The most useful forms of the conservation equations for fluid flows are obtained by introducing a suitable model for the viscous stresses  $\tau_{ij}$ . In many fluid flows the viscous stresses can be expressed as functions of the local deformation rate or strain rate. In three-dimensional flows the local rate of deformation is composed of the linear deformation rate and the volumetric deformation rate.

All gases and many liquids are isotropic. Liquids that contain significant quantities of polymer molecules may exhibit anisotropic or directional viscous stress properties as a result of the alignment of the chain-like polymer molecules with the flow. Such fluids are beyond the scope of this introductory course and we shall continue the development by assuming that the fluids are isotropic.

The rate of linear deformation of a fluid element has nine components in three dimensions, six of which are independent in isotropic fluids (Schlichting, 1979). They are denoted by the symbol  $s_{ij}$ . The suffix system is identical to that for stress components (see section 2.1.3). There are three linear elongating deformation components:

$$s_{xx} = \frac{\partial u}{\partial x} \quad s_{yy} = \frac{\partial v}{\partial y} \quad s_{zz} = \frac{\partial w}{\partial z} \quad (2.30a)$$

There are also six shearing linear deformation components:

$$s_{xy} = s_{yx} = \frac{1}{2} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \quad \text{and} \quad s_{xz} = s_{zx} = \frac{1}{2} \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)$$

$$s_{yz} = s_{zy} = \frac{1}{2} \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \quad (2.30b)$$

The volumetric deformation is given by

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = \operatorname{div} \mathbf{u} \quad (2.30c)$$

In a **Newtonian fluid** the viscous stresses are proportional to the rates of deformation. The three-dimensional form of Newton's law of viscosity for compressible flows involves two constants of proportionality: the first (dynamic) viscosity,  $\mu$ , to relate stresses to linear deformations, and the second viscosity,  $\lambda$ , to relate stresses to the volumetric deformation. The nine viscous stress components, of which six are independent, are

$$\begin{aligned} \tau_{xx} &= 2\mu \frac{\partial u}{\partial x} + \lambda \operatorname{div} \mathbf{u} & \tau_{yy} &= 2\mu \frac{\partial v}{\partial y} + \lambda \operatorname{div} \mathbf{u} & \tau_{zz} &= 2\mu \frac{\partial w}{\partial z} + \lambda \operatorname{div} \mathbf{u} \\ \tau_{xy} &= \tau_{yx} = \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \tau_{xz} &= \tau_{zx} = \mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \\ \tau_{yz} &= \tau_{zy} = \mu \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \end{aligned} \quad (2.31)$$

Not much is known about the second viscosity  $\lambda$ , because its effect is small in practice. For gases a good working approximation can be obtained by taking the value  $\lambda = -\frac{2}{3}\mu$  (Schlichting, 1979). Liquids are incompressible so the mass conservation equation is  $\operatorname{div} \mathbf{u} = 0$  and the viscous stresses are just twice the local rate of linear deformation times the dynamic viscosity.

Substitution of the above shear stresses (2.31) into (2.14a–c) yields the so-called Navier–Stokes equations, named after the two nineteenth-century scientists who derived them independently:

$$\begin{aligned} \rho \frac{Du}{Dt} &= -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left[ 2\mu \frac{\partial u}{\partial x} + \lambda \operatorname{div} \mathbf{u} \right] + \frac{\partial}{\partial y} \left[ \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] \\ &\quad + \frac{\partial}{\partial z} \left[ \mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right] + S_{M_x} \end{aligned} \quad (2.32a)$$

$$\begin{aligned} \rho \frac{Dv}{Dt} &= -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left[ \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial y} \left[ 2\mu \frac{\partial v}{\partial y} + \lambda \operatorname{div} \mathbf{u} \right] \\ &\quad + \frac{\partial}{\partial z} \left[ \mu \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right] + S_{M_y} \end{aligned} \quad (2.32b)$$

$$\begin{aligned} \rho \frac{Dw}{Dt} &= -\frac{\partial p}{\partial z} + \frac{\partial}{\partial x} \left[ \mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right] + \frac{\partial}{\partial y} \left[ \mu \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right] \\ &\quad + \frac{\partial}{\partial z} \left[ 2\mu \frac{\partial w}{\partial z} + \lambda \operatorname{div} \mathbf{u} \right] + S_{M_z} \end{aligned} \quad (2.32c)$$

Often it is useful to rearrange the viscous stress terms as follows:

$$\begin{aligned}
& \frac{\partial}{\partial x} \left[ 2\mu \frac{\partial u}{\partial x} + \lambda \operatorname{div} \mathbf{u} \right] + \frac{\partial}{\partial y} \left[ \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial z} \left[ \mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right] \\
&= \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left( \mu \frac{\partial u}{\partial z} \right) \\
&\quad + \left[ \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial z} \left( \mu \frac{\partial w}{\partial x} \right) + \frac{\partial}{\partial x} (\lambda \operatorname{div} \mathbf{u}) \right] \\
&= \operatorname{div}(\mu \operatorname{grad} u) + [s_{Mx}]
\end{aligned}$$

The viscous stresses in the  $y$ - and  $z$ -component equations can be recast in a similar manner. We clearly intend to simplify the momentum equations by ‘hiding’ the bracketed smaller contributions to the viscous stress terms in the momentum source. Defining a new source by

$$S_M = S_m + [s_M] \quad (2.33)$$

the **Navier–Stokes equations** can be written in the most useful form for the development of the finite volume method:

$$\boxed{\rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \operatorname{div}(\mu \operatorname{grad} u) + S_{Mx}} \quad (2.34a)$$

$$\boxed{\rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \operatorname{div}(\mu \operatorname{grad} v) + S_{My}} \quad (2.34b)$$

$$\boxed{\rho \frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \operatorname{div}(\mu \operatorname{grad} w) + S_{Mz}} \quad (2.34c)$$

If we use the Newtonian model for viscous stresses in the internal energy equation (2.24) we obtain after some rearrangement

$$\boxed{\rho \frac{Di}{Dt} = -p \operatorname{div} \mathbf{u} + \operatorname{div}(k \operatorname{grad} T) + \Phi + S_i} \quad (2.35)$$

All the effects due to viscous stresses in this internal energy equation are described by the dissipation function  $\Phi$ , which, after considerable algebra, can be shown to be equal to

$$\begin{aligned}
\Phi &= \mu \left\{ 2 \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \left( \frac{\partial w}{\partial z} \right)^2 \right] \right. \\
&\quad \left. + \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)^2 \right\} \\
&\quad + \lambda (\operatorname{div} \mathbf{u})^2
\end{aligned} \quad (2.36)$$

The dissipation function is non-negative since it only contains squared terms and represents a source of internal energy due to deformation work on the fluid particle. This work is extracted from the mechanical agency which causes the motion and converted into internal energy or heat.

## 2.4 Conservative form of the governing equations of fluid flow

To summarise the findings thus far, we quote in Table 2.1 the conservative or divergence form of the system of equations which governs the time-dependent three-dimensional fluid flow and heat transfer of a compressible Newtonian fluid.

**Table 2.1** Governing equations of the flow of a compressible Newtonian fluid

$$\text{Continuity} \quad \frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) = 0 \quad (2.4)$$

$$x\text{-momentum} \quad \frac{\partial(\rho u)}{\partial t} + \operatorname{div}(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + \operatorname{div}(\mu \operatorname{grad} u) + S_{M_x} \quad (2.37a)$$

$$y\text{-momentum} \quad \frac{\partial(\rho v)}{\partial t} + \operatorname{div}(\rho v \mathbf{u}) = -\frac{\partial p}{\partial y} + \operatorname{div}(\mu \operatorname{grad} v) + S_{M_y} \quad (2.37b)$$

$$z\text{-momentum} \quad \frac{\partial(\rho w)}{\partial t} + \operatorname{div}(\rho w \mathbf{u}) = -\frac{\partial p}{\partial z} + \operatorname{div}(\mu \operatorname{grad} w) + S_{M_z} \quad (2.37c)$$

$$\text{Energy} \quad \frac{\partial(\rho i)}{\partial t} + \operatorname{div}(\rho i \mathbf{u}) = -p \operatorname{div} \mathbf{u} + \operatorname{div}(k \operatorname{grad} T) + \Phi + S_i \quad (2.38)$$

$$\begin{aligned} \text{Equations} \\ \text{of state} \end{aligned} \quad p = p(\rho, T) \text{ and } i = i(\rho, T) \quad (2.28)$$

$$\text{e.g. perfect gas } p = \rho RT \text{ and } i = C_v T \quad (2.29)$$

Momentum source  $S_M$  and dissipation function  $\Phi$  are defined by (2.33) and (2.36) respectively.

It is interesting to note that the thermodynamic equilibrium assumption of section 2.2 has supplemented the five flow equations (PDEs) with two further algebraic equations. The further introduction of the Newtonian model, which expresses the viscous stresses in terms of gradients of velocity components, has resulted in a system of seven equations with seven unknowns. With an equal number of equations and unknown functions this system is mathematically closed, i.e. it can be solved provided that suitable auxiliary conditions, namely initial and boundary conditions, are supplied.

## 2.5 Differential and integral forms of the general transport equations

It is clear from Table 2.1 that there are significant commonalities between the various equations. If we introduce a general variable  $\phi$  the conservative form of all fluid flow equations, including equations for scalar quantities such as temperature and pollutant concentration etc., can usefully be written in the following form:

$$\frac{\partial(\rho \phi)}{\partial t} + \operatorname{div}(\rho \phi \mathbf{u}) = \operatorname{div}(\Gamma \operatorname{grad} \phi) + S_\phi \quad (2.39)$$

In words,

Rate of increase of $\phi$ of fluid element	Net rate of flow + of $\phi$ out of fluid element	Rate of increase = of $\phi$ due to diffusion	Rate of increase + of $\phi$ due to sources
---	---	---	---

Equation (2.39) is the so-called **transport equation** for property  $\phi$ . It clearly highlights the various transport processes: the **rate of change** term and the **convective** term on the left hand side and the **diffusive** term ( $\Gamma$  = diffusion coefficient) and the **source** term respectively on the right hand side. In order to bring out the common features we have, of course, had to hide the terms that are not shared between the equations in the source terms. Note that equation (2.39) can be made to work for the internal energy equation by changing  $i$  into  $T$  or vice versa by means of an equation of state.

Equation (2.39) is used as the starting point for computational procedures in the finite volume method. By setting  $\phi$  equal to  $1, u, v, w$  and  $i$  (or  $T$  or  $h_0$ ) and selecting appropriate values for diffusion coefficient  $\Gamma$  and source terms, we obtain special forms of Table 2.1 for each of the five PDEs for mass, momentum and energy conservation. The key step of the finite volume method, which is to be developed from Chapter 4 onwards, is the integration of (2.39) over a three-dimensional control volume (CV):

$$\int_{\text{CV}} \frac{\partial(\rho\phi)}{\partial t} dV + \int_{\text{CV}} \text{div}(\rho\phi\mathbf{u})dV = \int_{\text{CV}} \text{div}(\Gamma \text{ grad } \phi)dV + \int_{\text{CV}} S_\phi dV \quad (2.40)$$

The volume integrals in the second term on the left hand side, the convective term, and in the first term on the right hand side, the diffusive term, are rewritten as integrals over the entire bounding surface of the control volume by using Gauss's divergence theorem. For a vector  $\mathbf{a}$  this theorem states

$$\int_{\text{CV}} \text{div}(\mathbf{a})dV = \int_A \mathbf{n} \cdot \mathbf{a} dA \quad (2.41)$$

The physical interpretation of  $\mathbf{n} \cdot \mathbf{a}$  is the component of vector  $\mathbf{a}$  in the direction of the vector  $\mathbf{n}$  normal to surface element  $dA$ . Thus the integral of the divergence of a vector  $\mathbf{a}$  over a volume is equal to the component of  $\mathbf{a}$  in the direction normal to the surface which bounds the volume summed (integrated) over the entire bounding surface  $A$ . Applying Gauss's divergence theorem, equation (2.40) can be written as follows:

$$\frac{\partial}{\partial t} \left( \int_{\text{CV}} \rho\phi dV \right) + \int_A \mathbf{n} \cdot (\rho\phi\mathbf{u})dA = \int_A \mathbf{n} \cdot (\Gamma \text{ grad } \phi)dA + \int_{\text{CV}} S_\phi dV \quad (2.42)$$

The order of integration and differentiation has been changed in the first term on the left hand side of (2.42) to illustrate its physical meaning. This term signifies the **rate of change of the total amount of fluid property  $\phi$  in the control volume**. The product  $\mathbf{n} \cdot \rho\phi\mathbf{u}$  expresses the flux component of property  $\phi$  due to fluid flow along the outward normal vector  $\mathbf{n}$ , so the second term on the left hand side of (2.42), the convective term, therefore is the **net rate of decrease of fluid property  $\phi$  of the fluid element due to convection**.

A diffusive flux is positive in the direction of a negative gradient of the fluid property  $\phi$ , i.e. along direction  $-\text{grad } \phi$ . For instance, heat is conducted in the direction of negative temperature gradients. Thus, the product  $\mathbf{n} \cdot (-\Gamma \text{ grad } \phi)$  is the component of diffusion flux along the outward normal vector, so out of the fluid element. Similarly, the product  $\mathbf{n} \cdot (\Gamma \text{ grad } \phi)$ , which is also equal to  $\Gamma(-\mathbf{n} \cdot (-\text{grad } \phi))$ , can be interpreted as a positive diffusion flux in the direction of the inward normal vector  $-\mathbf{n}$ , i.e. into the fluid element. The first term on the right hand side of (2.42), the diffusive term, is thus associated with a flux into the element and represents the **net rate of increase of fluid property  $\phi$  of the fluid element due to diffusion**. The final term on the right hand side of this equation gives the **rate of increase of property  $\phi$  as a result of sources** inside the fluid element.

In words, relationship (2.42) can be expressed as follows:

Rate of increase of $\phi$ inside the control volume	Net rate of decrease of $\phi$ due to convection across the control boundaries	= Net rate of increase of $\phi$ due to diffusion across the control boundaries	+ Net rate of creation of $\phi$ inside the control volume
--	--	---	--

This discussion clarifies that integration of the PDE generates a statement of the conservation of a fluid property for a finite size (macroscopic) control volume.

In steady state problems the rate of change term of (2.42) is equal to zero. This leads to the integrated form of the steady transport equation:

$$\int_A \mathbf{n} \cdot (\rho \phi \mathbf{u}) dA = \int_A \mathbf{n} \cdot (\Gamma \text{ grad } \phi) dA + \int_{CV} S_\phi dV \quad (2.43)$$

In time-dependent problems it is also necessary to integrate with respect to time  $t$  over a small interval  $\Delta t$  from, say,  $t$  until  $t + \Delta t$ . This yields the most general integrated form of the transport equation:

$$\begin{aligned} & \int_{\Delta t} \frac{\partial}{\partial t} \left( \int_{CV} \rho \phi dV \right) dt + \int_{\Delta t} \int_A \mathbf{n} \cdot (\rho \phi \mathbf{u}) dA dt \\ &= \int_{\Delta t} \int_A \mathbf{n} \cdot (\Gamma \text{ grad } \phi) dA dt + \int_{\Delta t} \int_{CV} S_\phi dV dt \end{aligned} \quad (2.44)$$

## 2.6

### Classification of physical behaviours

Now that we have derived the conservation equations of fluid flows the time has come to turn our attention to the issue of the initial and boundary conditions that are needed in conjunction with the equations to construct a well-posed mathematical model of a fluid flow. First we distinguish two principal categories of physical behaviour:

- Equilibrium problems
- Marching problems

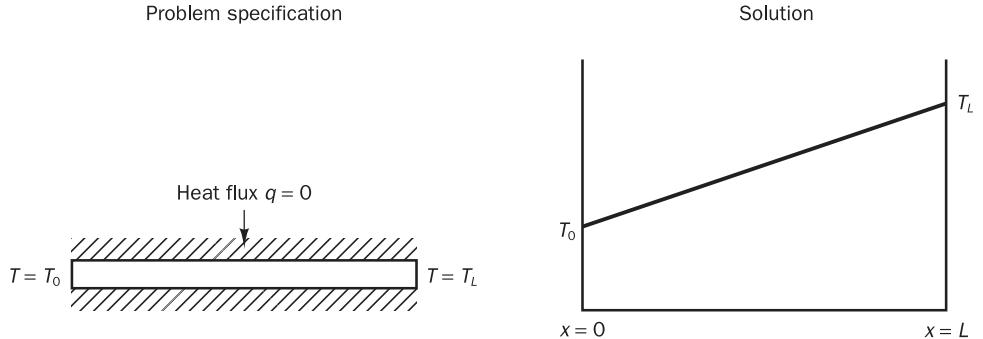
### Equilibrium problems

The problems in the first category are steady state situations, e.g. the steady state distribution of temperature in a rod of solid material or the equilibrium stress distribution of a solid object under a given applied load, as well as many steady fluid flows. These and many other steady state problems are governed by **elliptic equations**. The prototype elliptic equation is Laplace's equation, which describes irrotational flow of an incompressible fluid and steady state conductive heat transfer. In two dimensions we have

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (2.45)$$

A very simple example of an equilibrium problem is the steady state heat conduction (where  $\phi = T$  in equation (2.45)) in an insulated rod of metal whose ends at  $x = 0$  and  $x = L$  are kept at constant, but different, temperatures  $T_0$  and  $T_L$  (Figure 2.6).

**Figure 2.6** Steady state temperature distribution of an insulated rod



This problem is one-dimensional and governed by the equation  $k d^2T/dx^2 = 0$ . Under the given boundary conditions the temperature distribution in the  $x$ -direction will, of course, be a straight line. A unique solution to this and all elliptic problems can be obtained by specifying conditions on the dependent variable (here the temperature or its normal derivative the heat flux) on all the boundaries of the solution domain. Problems requiring data over the entire boundary are called **boundary-value problems**.

An important feature of elliptic problems is that a disturbance in the interior of the solution, e.g. a change in temperature due to the sudden appearance of a small local heat source, changes the solution everywhere else. Disturbance signals travel in all directions through the interior solution. Consequently, the solutions to physical problems described by elliptic equations are always smooth even if the boundary conditions are discontinuous, which is a considerable advantage to the designer of numerical methods. To ensure that information propagates in all directions, the numerical techniques for elliptic problems must allow events at each point to be influenced by all its neighbours.

### Marching problems

Transient heat transfer, all unsteady flows and wave phenomena are examples of problems in the second category, the marching or propagation problems. These problems are governed by **parabolic or hyperbolic equations**. However, not all marching problems are unsteady. We will see further on

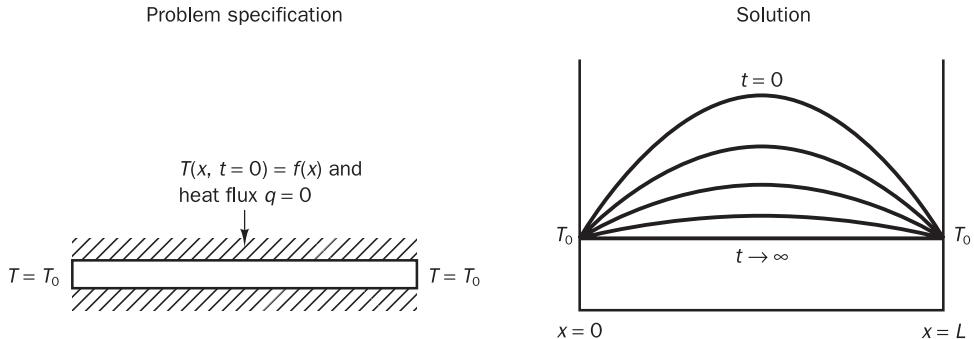
that certain steady flows are described by parabolic or hyperbolic equations. In these cases the flow direction acts as a time-like co-ordinate along which marching is possible.

**Parabolic equations** describe time-dependent problems, which involve significant amounts of diffusion. Examples are unsteady viscous flows and unsteady heat conduction. The prototype parabolic equation is the diffusion equation

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2} \quad (2.46)$$

The transient distribution of temperature (again  $\phi = T$ ) in an insulated rod of metal whose ends at  $x = 0$  and  $x = L$  are kept at constant and equal temperature  $T_0$  is governed by the diffusion equation. This problem arises when the rod cools down after an initially uniform source is switched off at time  $t = 0$ . The temperature distribution at the start is a parabola with a maximum at  $x = L/2$  (Figure 2.7).

**Figure 2.7** Transient distribution of temperature in an insulated rod



The steady state consists of a uniform distribution of temperature  $T = T_0$  throughout the rod. The solution of the diffusion equation (2.46) yields the exponential decay of the initial quadratic temperature distribution. Initial conditions are needed in the entire rod and conditions on all its boundaries are required for all times  $t > 0$ . This type of problem is termed an **initial-boundary-value problem**.

A disturbance at a point in the interior of the solution region (i.e.  $0 < x < L$  and time  $t_1 > 0$ ) can only influence events at later times  $t > t_1$  (unless we allow time travel!). The solutions move forward in time and diffuse in space. The occurrence of diffusive effects ensures that the solutions are always smooth in the interior at times  $t > 0$  even if the initial conditions contain discontinuities. The steady state is reached as time  $t \rightarrow \infty$  and is elliptic. This change of character can be easily seen by setting  $\partial\phi/\partial t = 0$  in equation (2.46). The governing equation is now equal to the one governing the steady temperature distribution in the rod.

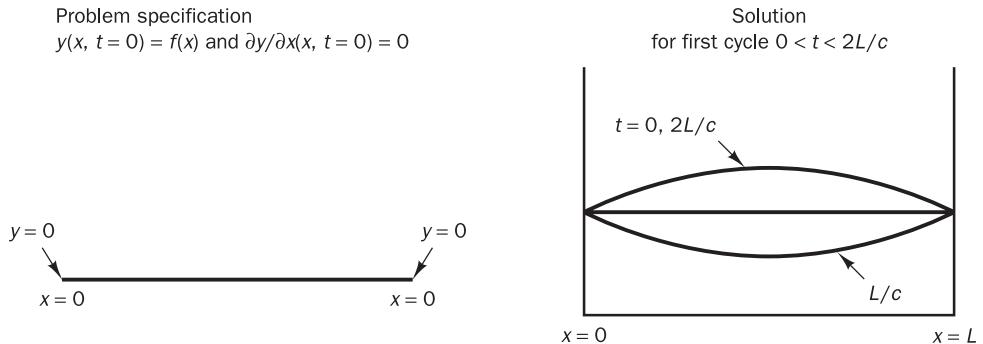
**Hyperbolic equations** dominate the analysis of vibration problems. In general they appear in time-dependent processes with negligible amounts of energy dissipation. The prototype hyperbolic equation is the wave equation

$$\frac{\partial^2 \phi}{\partial t^2} = c^2 \frac{\partial^2 \phi}{\partial x^2} \quad (2.47)$$

The above form of the equation governs the transverse displacement ( $\phi = y$ ) of a string under tension during small-amplitude vibrations and also acoustic oscillations (Figure 2.8). The constant  $c$  is the wave speed. It is relatively

straightforward to compute the fundamental mode of vibration of a string of length  $L$  using (2.47).

**Figure 2.8** Vibrations of a string under tension



Solutions to wave equation (2.47) and other hyperbolic equations can be obtained by specifying two initial conditions on the displacement  $y$  of the string and one condition on all boundaries for times  $t > 0$ . Thus hyperbolic problems are also initial-boundary-value problems.

If the initial amplitude is given by  $a$ , the solution of this problem is

$$y(x, t) = a \cos\left(\frac{\pi c t}{L}\right) \sin\left(\frac{\pi x}{L}\right)$$

The solution shows that the vibration amplitude remains constant, which demonstrates the lack of damping in the system. This absence of damping has a further important consequence. Consider, for example, initial conditions corresponding to a near-triangular initial shape whose apex is a section of a circle with very small radius of curvature. This initial shape has a sharp discontinuity at the apex, but it can be represented by means of a Fourier series as a combination of sine waves. The governing equation is linear so each of the individual Fourier components (and also their sum) would persist in time without change of amplitude. The final result is that the discontinuity remains undiminished due to the absence of a dissipation mechanism to remove the kink in the slope.

Compressible fluid flows at speeds close to and above the speed of sound exhibit shockwaves and it turns out that the inviscid flow equations are hyperbolic at these speeds. The shockwave discontinuities are manifestations of the hyperbolic nature of such flows. Computational algorithms for hyperbolic problems are shaped by the need to allow for the possible existence of discontinuities in the interior of the solution.

It will be shown that disturbances at a point can only influence a limited region in space. The speed of disturbance propagation through an hyperbolic problem is finite and equal to the wave speed  $c$ . In contrast, parabolic and elliptic models assume infinite propagation speeds.

## 2.7

### The role of characteristics in hyperbolic equations

Hyperbolic equations have a special behaviour, which is associated with the finite speed, namely the wave speed, at which information travels through the problem. This distinguishes hyperbolic equations from the two other types. To develop the ideas about the role of characteristic lines in hyperbolic problems we consider again a simple hyperbolic problem described by

wave equation (2.47). It can be shown (The Open University, 1984) that a change of variables to  $\zeta = x - ct$  and  $\eta = x + ct$  transforms the wave equation into the following standard form:

$$\frac{\partial^2 \phi}{\partial \zeta \partial \eta} = 0 \quad (2.48)$$

The transformation requires repeated application of the chain rule for differentiation to express the derivatives of equation (2.47) in terms of derivatives of the transform variables. Equation (2.48) can be solved very easily. The solution is, of course,  $\phi(\zeta, \eta) = F_1(\zeta) + F_2(\eta)$ , where  $F_1$  and  $F_2$  can be any function.

A return to the original variables yields the general solution of equation (2.47):

$$\phi(x, t) = F_1(x - ct) + F_2(x + ct) \quad (2.49)$$

The first component of the solution, function  $F_1$ , is constant if  $x - ct$  is constant and hence along lines of slope  $dt/dx = 1/c$  in the  $x-t$  plane. The second component  $F_2$  is constant if  $x + ct$  is constant, so along lines of slope  $dt/dx = -1/c$ . The lines  $x - ct = \text{constant}$  and  $x + ct = \text{constant}$  are called the characteristics. Functions  $F_1$  and  $F_2$  represent the so-called **simple wave solutions** of the problem, which are travelling waves with velocities  $+c$  and  $-c$  without change of shape or amplitude.

The particular forms of functions  $F_1$  and  $F_2$  can be obtained from the initial and boundary conditions of the problem. Let us consider a very long string ( $-\infty < x < \infty$ ) and let the following initial conditions hold:

$$\phi(x, 0) = f(x) \quad \text{and} \quad \partial\phi/\partial t(x, 0) = g(x) \quad (2.50)$$

Combining (2.49) and (2.50) we obtain

$$F_1(x) + F_2(x) = f(x) \quad \text{and} \quad -cF'_1(x) + cF'_2(x) = g(x) \quad (2.51)$$

It can be shown (Bland, 1988) that the particular solution of wave equation (2.47) with initial conditions (2.50) is given by

$$\phi(x, t) = \frac{1}{2}[f(x - ct) + f(x + ct)] + \frac{1}{2c} \int_{x-ct}^{x+ct} g(s)ds \quad (2.52)$$

Careful inspection of (2.52) shows that  $\phi$  at point  $(x, t)$  in the solution domain depends only on the initial conditions in the interval  $(x - ct, x + ct)$ . It is particularly important to note that this implies that **the solution at  $(x, t)$  does not depend on initial conditions outside this interval**.

Figure 2.9 seeks to illustrate this point. The characteristics  $x - ct = \text{constant}$  and  $x + ct = \text{constant}$  through the point  $(x', t')$  intersect the  $x$ -axis at the points  $(x' - ct', 0)$  and  $(x' + ct', 0)$  respectively. The region in the  $x-t$  plane enclosed by the  $x$ -axis and the two characteristics is termed the **domain of dependence**.

In accordance with (2.52) the solution at  $(x', t')$  is influenced only by events inside the domain of dependence and not those outside. Physically this is caused by the limited propagation speed (equal to wave speed  $c$ ) of mutual influences through the solution domain. Changes at the point  $(x', t')$  influence events at later times within the **zone of influence** shown in Figure 2.9, which is again bounded by the characteristics.

**Figure 2.9** Domain of dependence and zone of influence for an hyperbolic problem

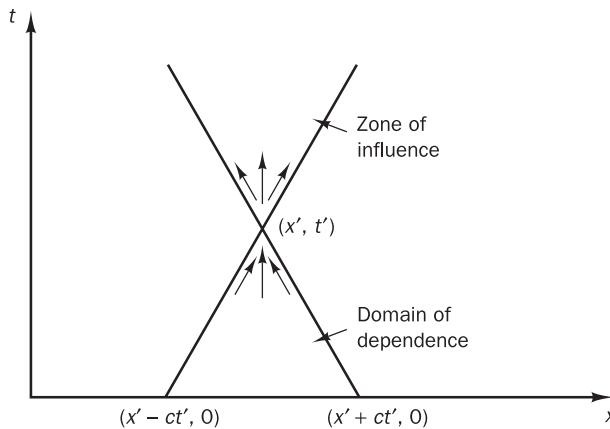
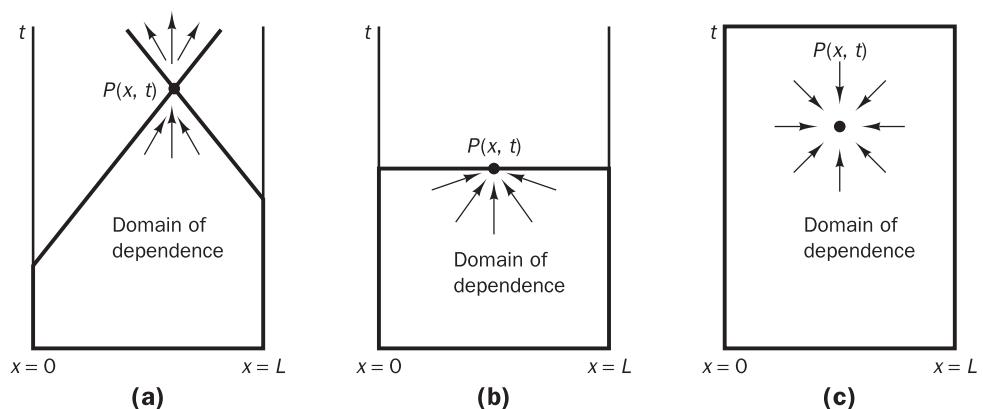


Figure 2.10a shows the situation for the vibrations of a string fixed at  $x = 0$  and  $x = L$ . For points very close to the  $x$ -axis the domain of dependence is enclosed by two characteristics, which originate at points on the  $x$ -axis. The characteristics through points such as  $P$  intersect the problem boundaries. The domain of dependence of  $P$  is bounded by these two characteristics and the lines  $t = 0$ ,  $x = 0$  and  $x = L$ .

**Figure 2.10** Domains of dependence for the (a) hyperbolic, (b) parabolic and (c) elliptic problem



The shape of the domains of dependence (see Figures 2.10b and c) in parabolic and elliptic problems is different because the speed of information travel is assumed to be infinite. The bold lines which demarcate the boundaries of each domain of dependence give the regions for which initial and/or boundary conditions are needed in order to be able to generate a solution at the point  $P(x, t)$  in each case.

The way in which changes at one point affect events at other points depends on whether a physical problem represents a steady state or a transient phenomenon and whether the propagation speed of disturbances is finite or infinite. This has resulted in a classification of physical behaviours, and hence attendant PDEs, into elliptic, parabolic and hyperbolic problems. The distinguishing features of each of the categories were illustrated by considering three simple prototype second-order equations. In the following sections we will discuss methods of classifying more complex PDEs and briefly state the limitations of the computational methods that will be developed later in this text in terms of the classification of the flow problems to be solved. A summary of the main features that have been identified so far is given in Table 2.2.

**Table 2.2** Classification of physical behaviours

<i>Problem type</i>	<i>Equation type</i>	<i>Prototype equation</i>	<i>Conditions</i>	<i>Solution domain</i>	<i>Solution smoothness</i>
Equilibrium problems	Elliptic	$\operatorname{div} \operatorname{grad} \phi = 0$	Boundary conditions	Closed domain	Always smooth
Marching problems with dissipation	Parabolic	$\frac{\partial \phi}{\partial t} = \alpha \operatorname{div} \operatorname{grad} \phi$	Initial and boundary conditions	Open domain	Always smooth
Marching problems without dissipation	Hyperbolic	$\frac{\partial^2 \phi}{\partial t^2} = c^2 \operatorname{div} \operatorname{grad} \phi$	Initial and boundary conditions	Open domain	May be discontinuous

**2.8****Classification method for simple PDEs**

A practical method of classifying PDEs is developed for a general second-order PDE in two co-ordinates  $x$  and  $y$ . Consider

$$a \frac{\partial^2 \phi}{\partial x^2} + b \frac{\partial^2 \phi}{\partial x \partial y} + c \frac{\partial^2 \phi}{\partial y^2} + d \frac{\partial \phi}{\partial x} + e \frac{\partial \phi}{\partial y} + f\phi + g = 0 \quad (2.53)$$

At first we shall assume that the equation is linear and  $a, b, c, d, e, f$  and  $g$  are constants.

The classification of a PDE is governed by the behaviour of its highest-order derivatives, so we need only consider the second-order derivatives. The class of a second-order PDE can be identified by searching for possible simple wave solutions. If they exist this indicates a hyperbolic equation. If not the equation is parabolic or elliptic.

Simple wave solutions occur if the characteristic equation (2.54) below has two real roots:

$$a \left( \frac{dy}{dx} \right)^2 - b \left( \frac{dy}{dx} \right) + c = 0 \quad (2.54)$$

The existence or otherwise of roots of the characteristic equation depends on the value of discriminant ( $b^2 - 4ac$ ). Table 2.3 outlines the three cases.

**Table 2.3** Classification of linear second-order PDEs

$b^2 - 4ac$	<i>Equation type</i>	<i>Characteristics</i>
$> 0$	Hyperbolic	Two real characteristics
$= 0$	Parabolic	One real characteristic
$< 0$	Elliptic	No characteristics

It is left as an exercise for the reader to verify the nature of the three prototype PDEs in section 2.6 by evaluating the discriminant.

The classification method by searching for the roots of the characteristic equation also applies if the coefficients  $a, b$  and  $c$  are functions of  $x$  and  $y$  or if the equation is non-linear. In the latter case  $a, b$  and  $c$  may be functions of dependent variable  $\phi$  or its first derivatives. It is now possible that the

equation type differs in various regions of the solution domain. As an example we consider the following equation:

$$y \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (2.55)$$

We look at the behaviour within the region  $-1 < y < 1$ . Hence  $a = a(x, y) = y$ ,  $b = 0$  and  $c = 1$ . The value of discriminant ( $b^2 - 4ac$ ) is equal to  $-4y$ . We need to distinguish three cases:

- If  $y < 0$ :  $b^2 - 4ac > 0$  so the equation is hyperbolic
- If  $y = 0$ :  $b^2 - 4ac = 0$  so the equation is parabolic
- If  $y > 0$ :  $b^2 - 4ac < 0$  and hence the equation is elliptic

Equation (2.55) is of mixed type. The equation is locally hyperbolic, parabolic or elliptic depending on the value of  $y$ . For the non-linear case similar remarks apply. The classification of the PDE depends on the local values of  $a$ ,  $b$  and  $c$ .

Second-order PDEs in  $N$  independent variables ( $x_1, x_2, \dots, x_N$ ) can be classified by rewriting them first in the following form with  $A_{jk} = A_{kj}$ :

$$\sum_{j=1}^N \sum_{k=1}^N A_{jk} \frac{\partial^2 \phi}{\partial x_j \partial x_k} + H = 0 \quad (2.56)$$

Fletcher (1991) explains that the equation can be classified on the basis of the eigenvalues of a matrix with entries  $A_{jk}$ . Hence we need to find values for  $\lambda$  for which

$$\det[A_{jk} - \lambda I] = 0 \quad (2.57)$$

The classification rules are:

- if any eigenvalue  $\lambda = 0$ : the equation is parabolic
- if all eigenvalues  $\lambda \neq 0$  and they are all of the same sign: the equation is elliptic
- if all eigenvalues  $\lambda \neq 0$  and all but one are of the same sign: the equation is hyperbolic

In the cases of Laplace's equation, the diffusion equation and the wave equation it is simple to verify that this method yields the same results as the solution of characteristic equation (2.54).

## 2.9

### Classification of fluid flow equations

Systems of first-order PDEs with more than two independent variables are similarly cast in matrix form. Their classification involves finding eigenvalues of the resulting matrix. Systems of second-order PDEs or mixtures of first- and second-order PDEs can also be classified with this method. The first stage of the method involves the introduction of auxiliary variables, which express each second-order equation as first-order equations. Care must be taken to select the auxiliary variables in such a way that the resulting matrix is non-singular.

The Navier–Stokes equation and its reduced forms can be classified using such a matrix approach. The details are beyond the scope of this introduction to the subject. We quote the main results in Table 2.4 and refer the interested reader to Fletcher (1991) for a full discussion.

**Table 2.4** Classification of the main categories of fluid flow

	<i>Steady flow</i>	<i>Unsteady flow</i>
Viscous flow	Elliptic	Parabolic
Inviscid flow	$M < 1$ , elliptic $M > 1$ , hyperbolic	Hyperbolic
Thin shear layers	Parabolic	Parabolic

The classifications in Table 2.4 are the ‘formal’ classifications of the flow equations. In practice many fluid flows behave in a complex way. The steady Navier–Stokes equations and the energy (or enthalpy) equations are formally elliptic and the unsteady equations are parabolic.

The mathematical classification of inviscid flow equations is different from the Navier–Stokes and energy equations due to the complete absence of the (viscous) higher-order terms. The classification of the resulting equation set depends on the extent to which fluid compressibility plays a role and hence on the magnitude of the Mach number  $M$ . The elliptic nature of inviscid flows at Mach numbers below 1 originates from the action of pressure. If  $M < 1$  the pressure can propagate disturbances at the speed of sound, which is greater than the flow speed. But if  $M > 1$  the fluid velocity is greater than the propagation speed of disturbances and the pressure is unable to influence events in the upstream direction. Limitations on the zone of influence are a key feature of hyperbolic phenomena, so the supersonic inviscid flow equations are hyperbolic. Below, we will see a simple example that demonstrates this behaviour.

In thin shear layer flows all velocity derivatives in the flow ( $x$ - and  $z$ -) direction are much smaller than those in the cross-stream ( $y$ -) direction. Boundary layers, jets, mixing layers and wakes as well as fully developed duct flows fall within this category. In these conditions the governing equations contain only one (second-order) diffusion term and are therefore classified as parabolic.

As an illustration of the complexities which may arise in inviscid flows we analyse the potential equation which governs steady, isentropic, inviscid, compressible flow past a slender body (Shapiro, 1953) with a free stream Mach number  $M_\infty$ :

$$(1 - M_\infty^2) \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (2.58)$$

Taking  $x_1 = x$  and  $x_2 = y$  in equation (2.56) we have matrix elements  $A_{11} = 1 - M_\infty^2$ ,  $A_{12} = A_{21} = 0$  and  $A_{22} = 1$ . To classify the equation we need to solve

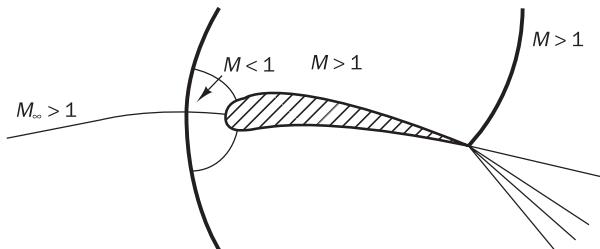
$$\det \begin{vmatrix} (1 - M_\infty^2) - \lambda & 0 \\ 0 & 1 - \lambda \end{vmatrix} = 0$$

The two solutions are  $\lambda_1 = 1$  and  $\lambda_2 = 1 - M_\infty^2$ . If the free stream Mach number is smaller than 1 (subsonic flow) both eigenvalues are greater than zero and the flow is elliptic. If the Mach number is greater than 1 (supersonic flow) the second eigenvalue is negative and the flow is hyperbolic. The reader is left to demonstrate that these results are identical to those obtained by considering the discriminant of characteristic equation (2.54).

It is interesting to note that we have discovered an instance of hyperbolic behaviour in a steady flow where both independent variables are space coordinates. The flow direction behaves in a time-like manner in hyperbolic inviscid flows and also in the parabolic thin shear layers. These problems are of the marching type and flows can be computed by marching in the time-like direction of increasing  $x$ .

The above example shows the dependence of the classification of compressible flows on the parameter  $M_\infty$ . The general equations of inviscid, compressible flow (the Euler equations) exhibit similar behaviour, but the classification parameter is now the local Mach number  $M$ . This complicates matters greatly when flows around and above  $M = 1$  are to be computed. Such flows may contain shockwave discontinuities and regions of subsonic (elliptic) flow and supersonic (hyperbolic) flow, whose exact locations are not known a priori. Figure 2.11 is a sketch of the flow around an aerofoil at a Mach number somewhat greater than 1.

**Figure 2.11** Sketch of flow around an aerofoil at supersonic free stream speed



## 2.10

### Auxiliary conditions for viscous fluid flow equations

The complicated mixture of elliptic, parabolic and hyperbolic behaviours has implications for the way in which boundary conditions enter into a flow problem, in particular at locations where flows are bounded by fluid boundaries. Unfortunately few theoretical results regarding the range of permissible boundary conditions are available for compressible flows. CFD practice is guided here by physical arguments and the success of its simulations. The **boundary conditions for a compressible viscous flow** are given in Table 2.5.

In the table subscripts  $n$  and  $t$  indicate directions normal (outward) and tangential to the boundary respectively and  $F$  are the given surface stresses.

**Table 2.5** Boundary conditions for compressible viscous flow

*Initial conditions for unsteady flows:*

- Everywhere in the solution region  $\rho$ ,  $\mathbf{u}$  and  $T$  must be given at time  $t = 0$ .

*Boundary conditions for unsteady and steady flows:*

- On solid walls  $\mathbf{u} = \mathbf{u}_w$  (no-slip condition)  
 $T = T_w$  (fixed temperature) or  $k\partial T/\partial n = -q_w$  (fixed heat flux)
- On fluid boundaries inlet:  $\rho$ ,  $\mathbf{u}$  and  $T$  must be known as a function of position  
outlet:  $-p + \mu\partial u_n/\partial n = F_n$  and  $\mu\partial u_t/\partial n = F_t$  (stress continuity)

---

CHAPTER  
**THREE**

---

## DISCRETIZATION METHODS

So far we have seen that there are significant benefits in obtaining a theoretical prediction of physical phenomena. The phenomena of interest here are governed by differential equations, which we have represented by a general equation for the variable  $\phi$ . Now our main task is to develop the means of solving this equation.

For ease of understanding, we shall assume in this chapter that the variable  $\phi$  is a function of only one independent variable  $x$ . However, the ideas developed here continue to be applicable when more than one independent variable is active.

### 3.1 THE NATURE OF NUMERICAL METHODS

#### 3.1-1 The Task

A numerical solution of a differential equation consists of a set of numbers from which the distribution of the dependent variable  $\phi$  can be constructed. In this sense, a numerical method is akin to a laboratory experiment, in which a set of instrument readings enables us to establish the distribution of the measured quantity in the domain under investigation. The numerical analyst and the laboratory experimenter both must remain content with only a *finite* number of numerical values as the outcome, although this number can, at least in principle, be made large enough for practical purposes.

Let us suppose that we decide to represent the variation of  $\phi$  by a polynomial in  $x$ ,

$$\phi = a_0 + a_1x + a_2x^2 + \cdots + a_mx^m, \quad (3.1)$$

and employ a numerical method to find the finite number of coefficients  $a_0, a_1, a_2, \dots, a_m$ . This will enable us to evaluate  $\phi$  at any location  $x$  by substituting the value of  $x$  and the values of the  $a$ 's into Eq. (3.1). This procedure is, however, somewhat inconvenient if our ultimate interest is to obtain the *values* of  $\phi$  at various locations. The values of the  $a$ 's are, by themselves, not particularly meaningful, and the substitution operation must be carried out to arrive at the required values of  $\phi$ . This leads us to the following thought: Why not construct a method that employs the *values* of  $\phi$  at a number of given points as the primary unknowns? Indeed, most numerical methods for solving differential equations do belong in this category, and therefore we shall limit our attention to such methods.

Thus, a numerical method treats as its basic unknowns the values of the dependent variable at a finite number of locations (called the *grid points*) in the calculation domain. The method includes the tasks of providing a set of algebraic equations for these unknowns and of prescribing an algorithm for solving the equations.

### 3.1-2 The Discretization Concept

In focusing attention on the values at the grid points, we have replaced the continuous information contained in the exact solution of the differential equation with discrete values. We have thus discretized the distribution of  $\phi$ , and it is appropriate to refer to this class of numerical methods as *discretization methods*.

The algebraic equations involving the unknown values of  $\phi$  at chosen grid points, which we shall now name the *discretization equations*, are derived from the differential equation governing  $\phi$ . In this derivation, we must employ some assumption about how  $\phi$  varies *between* the grid points. Although this "profile" of  $\phi$  could be chosen such that a single algebraic expression suffices for the whole calculation domain, it is often more practical to use *piecewise* profiles such that a given segment describes the variation of  $\phi$  over only a small region in terms of the  $\phi$  values at the grid points within and around that region. Thus, it is common to subdivide the calculation domain into a number of subdomains or elements such that a separate profile assumption can be associated with each subdomain.

In this manner, we encounter the discretization concept in another context. The continuum calculation domain has been discretized. It is this systematic discretization of space and of the dependent variables that makes it

possible to replace the governing differential equations with simple algebraic equations, which can be solved with relative ease.

### 3.1-3 The Structure of the Discretization Equation

A discretization equation is an algebraic relation connecting the values of  $\phi$  for a group of grid points. Such an equation is derived from the differential equation governing  $\phi$  and thus expresses the same physical information as the differential equation. That only a few grid points participate in a given discretization equation is a consequence of the piecewise nature of the profiles chosen. The value of  $\phi$  at a grid point thereby influences the distribution of  $\phi$  only in its immediate neighborhood. As the number of grid points becomes very large, the solution of the discretization equations is expected to approach the exact solution of the corresponding differential equation. This follows from the consideration that, as the grid points get closer together, the change in  $\phi$  between neighboring grid points becomes small, and then the actual details of the profile assumption become unimportant.

For a given differential equation, the possible discretization equations are by no means unique, although all types of discretization equations are, in the limit of a very large number of grid points, expected to give the same solution. The different types arise from the differences in the profile assumptions and in the methods of derivation.

Until now we have deliberately refrained from making reference to finite-difference and finite-element methods. Now it may be stated that these can be thought of as two alternative versions of the discretization method, which we have described in general terms. The distinction between the finite-difference method and the finite-element method results from the ways of choosing the profiles and deriving the discretization equations. The method that is to be the main focus of attention in this book has the *appearance* of a finite-difference method, but it employs many ideas that are typical of the finite-element methodology. To call the present method a finite-difference method might convey an adherence to the conventional finite-difference practice. For this reason, we shall refer to it simply as a discretization method. Also, we shall note in Chapter 8 how a method that has the appearance of a finite-element method can be constructed from the general principles presented in this book.

## 3.2 METHODS OF DERIVING THE DISCRETIZATION EQUATIONS

For a given differential equation, the required discretization equations can be derived in many ways. Here, we shall outline a few common methods and then indicate a preference.

### 3.2-1 Taylor-Series Formulation

The usual procedure for deriving finite-difference equations consists of approximating the derivatives in the differential equation via a truncated Taylor series. Let us consider the grid points shown in Fig. 3.1. For grid point 2, located midway between grid points 1 and 3 such that  $\Delta x = x_2 - x_1 = x_3 - x_2$ , the Taylor-series expansion around 2 gives

$$\phi_1 = \phi_2 - \Delta x \left( \frac{d\phi}{dx} \right)_2 + \frac{1}{2} (\Delta x)^2 \left( \frac{d^2\phi}{dx^2} \right)_2 - \dots \quad (3.2)$$

and 
$$\phi_3 = \phi_2 + \Delta x \left( \frac{d\phi}{dx} \right)_2 + \frac{1}{2} (\Delta x)^2 \left( \frac{d^2\phi}{dx^2} \right)_2 + \dots \quad (3.3)$$

Truncating the series just after the third term, and adding and subtracting the two equations, we obtain

$$\left( \frac{d\phi}{dx} \right)_2 = \frac{\phi_3 - \phi_1}{2 \Delta x} \quad (3.4)$$

and 
$$\left( \frac{d^2\phi}{dx^2} \right)_2 = \frac{\phi_1 + \phi_3 - 2\phi_2}{(\Delta x)^2} \quad (3.5)$$

The substitution of such expressions into the differential equation leads to the finite-difference equation.

The method includes the assumption that the variation of  $\phi$  is somewhat like a polynomial in  $x$ , so that the higher derivatives are unimportant. This assumption, however, leads to an undesirable formulation when, for example, exponential variations are encountered. (We shall refer to this matter again in Chapter 5.) The Taylor-series formulation is relatively straightforward but allows less flexibility and provides little insight into the physical meanings of the terms.\*

\*This is admittedly an entirely subjective view. Someone with proper mathematical training may find the Taylor-series method highly illuminating and meaningful.

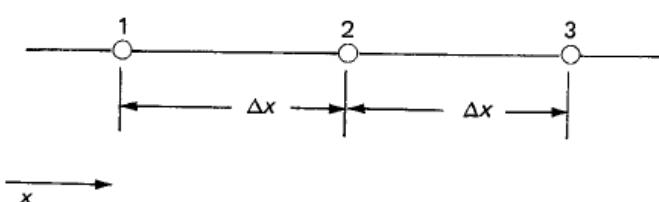


Figure 3.1 Three successive grid points used for the Taylor-series expansion.

### 3.2-2 Variational Formulation

Another method of obtaining the discretization equations is based on the calculus of variations. To understand the method fully, the reader should have sufficient knowledge of this branch of calculus. However, a general appreciation of the main ingredients of the formulation is all that is needed for the present purposes.

The calculus of variations shows that solving certain differential equations is equivalent to minimizing a related quantity called the *functional*. This equivalence is known as a variational principle. If the functional is minimized with respect to the grid-point values of the dependent variable, the resulting conditions give the required discretization equations. The variational formulation is very commonly employed in finite-element methods for stress analysis, where it can be linked to the virtual-work principle. In addition to its algebraic and conceptual complexity, the main drawback of this formulation is its limited applicability, since a variational principle does not exist for all differential equations of interest.

### 3.2-3 Method of Weighted Residuals

A powerful method for solving differential equations is the method of weighted residuals, which is described in detail by Finlayson (1972). The basic concept is simple and interesting. Let the differential equation be represented by

$$L(\phi) = 0. \quad (3.6)$$

Further, let us assume an approximate solution  $\bar{\phi}$  that contains a number of undetermined parameters, for example,

$$\bar{\phi} = a_0 + a_1x + a_2x^2 + \cdots + a_mx^m, \quad (3.7)$$

the  $a$ 's being the parameters. The substitution of  $\bar{\phi}$  into the differential equation leaves a residual  $R$ , defined as

$$R = L(\bar{\phi}). \quad (3.8)$$

We wish to make this residual small in some sense. Let us propose that

$$\int WR dx = 0, \quad (3.9)$$

where  $W$  is a weighting function and the integration is performed over the domain of interest. By choosing a succession of weighting functions, we can

generate as many equations as are required for evaluating the parameters. These algebraic equations containing the parameters as the unknowns are solved to obtain the approximate solution to the differential equation. Different versions of the method (known by specific names) result from the choice of different classes of weighting functions.

The method was very popular in boundary-layer analysis before the finite-difference method nearly replaced it. However, a connection with the finite-difference method, or rather with the discretization method, can be established if the approximate solution  $\phi$ , instead of being a single algebraic expression over the whole domain, is constructed via piecewise profiles with the grid-point values of  $\phi$  as the unknown parameters. Indeed, much of the recent development of the finite-element technique is also based on piecewise profiles used in conjunction with a particular weighted-residual practice known as the Galerkin method.

The simplest weighting function is  $W = 1$ . From this, a number of weighted-residual equations can be generated by dividing the calculation domain into subdomains or control volumes, and setting the weighting function to be unity over one subdomain at a time and zero everywhere else. This variant of the method of weighted residuals is called the *subdomain* method or the *control-volume* formulation. It implies that the integral of the residual over each control volume must become zero. Since we shall adopt the control-volume approach in this book, a more detailed discussion is desirable, which now follows.

### 3.2-4 Control-Volume Formulation

Often elementary textbooks on heat transfer derive the finite-difference equation via the Taylor-series method and then demonstrate that the resulting equation is consistent with a heat balance over a small region surrounding a grid point. We have also seen that the control-volume formulation can be regarded as a special version of the method of weighted residuals. The basic idea of the control-volume formulation is easy to understand and lends itself to direct physical interpretation. The calculation domain is divided into a number of nonoverlapping control volumes such that there is one control volume surrounding each grid point. The differential equation is integrated over each control volume. Piecewise profiles expressing the variation of  $\phi$  between the grid points are used to evaluate the required integrals. The result is the discretization equation containing the values of  $\phi$  for a group of grid points.

The discretization equation obtained in this manner expresses the conservation principle for  $\phi$  for the finite control volume, just as the differential equation expresses it for an infinitesimal control volume.\*

\*Indeed, deriving the control-volume discretization equation by integrating the differential equation over a finite control volume is a rather roundabout process, much

The most attractive feature of the control-volume formulation is that the resulting solution would imply that the *integral* conservation of quantities such as mass, momentum, and energy is exactly satisfied over any group of control volumes and, of course, over the whole calculation domain. This characteristic exists for *any* number of grid points—not just in a limiting sense when the number of grid points becomes large. Thus, even the coarse-grid solution exhibits *exact* integral balances.

When the discretization equations are solved to obtain the grid-point values of the dependent variable, the result can be viewed in two different ways. In the finite-element method and in most weighted-residual methods, the assumed variation of  $\phi$  consisting of the grid-point values and the interpolation functions (or profiles) between the grid points is taken as the approximate solution. In the finite-difference method, however, only the grid-point values of  $\phi$  are considered to constitute the solution, without any explicit reference as to how  $\phi$  varies between the grid points. This is akin to a laboratory experiment where the distribution of a quantity is obtained in terms of the measured values at some discrete locations without any statement about the variation *between* these locations. In our control-volume approach, we shall also adopt this view. We shall seek the solution in the form of the grid-point values only. The interpolation formulas or the profiles will be regarded as auxiliary relations needed to evaluate the required integrals in the formulation. Once the discretization equations are derived, the profile assumptions can be forgotten. This viewpoint permits complete freedom of choice in employing, if we wish, different profile assumptions for integrating different terms in the differential equation.

To make the foregoing discussion more concrete, we shall now derive the control-volume discretization equation for a simple situation.

### 3.3 AN ILLUSTRATIVE EXAMPLE

Let us consider steady one-dimensional heat conduction governed by

$$\frac{d}{dx} \left( k \frac{dT}{dx} \right) + S = 0, \quad (3.10)$$

where  $k$  is the thermal conductivity,  $T$  is the temperature, and  $S$  is the rate of heat generation per unit volume.

---

like preparing mashed potatoes from dehydrated potato powder. After all, textbook derivations of differential equations always start from the conservation principle applied to a small control volume. It is useful to imagine ourselves to be in the pre-calculus days; then the control-volume equation would have been our only way of stating the conservation principle.

## 1 Introduction to Transport Equations

A solid body will remain stationary or in motion at constant velocity unless acted on by external forces. When acted on by external forces, the momentum of a solid body will change according to Newton's second law. For a solid body with constant mass, Newton's second law can be written concisely in differential form as:

$$\mathbf{F} = m\mathbf{a} \quad \rightarrow \quad \mathbf{F} = m \frac{d\mathbf{v}}{dt} \quad (1)$$

where  $\mathbf{F}$  is the sum of the forces acting on the body,  $\mathbf{v}$  is the velocity of the body,  $m$  is the mass of the body and bold symbols represent vector quantities. If the mass of the solid body changes with time, then equation 1 becomes:

$$\mathbf{F} = \frac{d(m\mathbf{v})}{dt} \quad (2)$$

Hence, Newton's second law physically states that the rate of change of momentum (mass multiplied by velocity), is equal to the sum of the external forces acting on the body. The solid body velocity ( $\mathbf{v}$ ) is a vector quantity. In a Cartesian coordinate system, the solid body velocity can be resolved into components in the  $x$ ,  $y$  and  $z$  directions ( $\mathbf{v} = (v_x, v_y, v_z)$ ). Hence, when written in vector form, Newton's second law is a compact way of expressing three individual equations for the change of momentum of the body in the  $x$ ,  $y$  and  $z$  directions.

$$F_x = \frac{d(mv_x)}{dt} \quad F_y = \frac{d(mv_y)}{dt} \quad F_z = \frac{d(mv_z)}{dt} \quad (3)$$

If the mass of the object and the forces acting on it are known, then Newton's second law can be solved to calculate the velocity ( $\mathbf{v}$ ) of the object at a given time. The equations are solved by integration. In the same way that the velocity of a solid body can be calculated by solving Newton's second law, the velocity of a fluid (liquid or gas) can be calculated by solving the *Navier-Stokes* equations. The Navier-Stokes equations are analogous to Newton's second law and state that the rate of change of momentum of a fluid is equal to the sum of the external forces acting on the fluid. However, the Navier-Stokes equations are applied to a parcel/finite volume of fluid rather than a solid body.

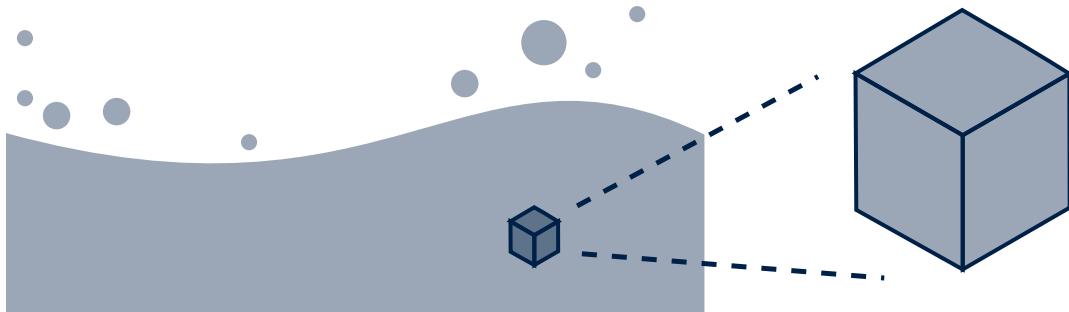
Figure 1 shows an example of a fluid parcel/volume that forms a part of the fluid continuum. The parcel has a volume  $V$  and can be any size. In concise vector form, the Navier-Stokes equations can be written as:

$$\frac{D(m\mathbf{U})}{Dt} = \mathbf{F} \quad (4)$$

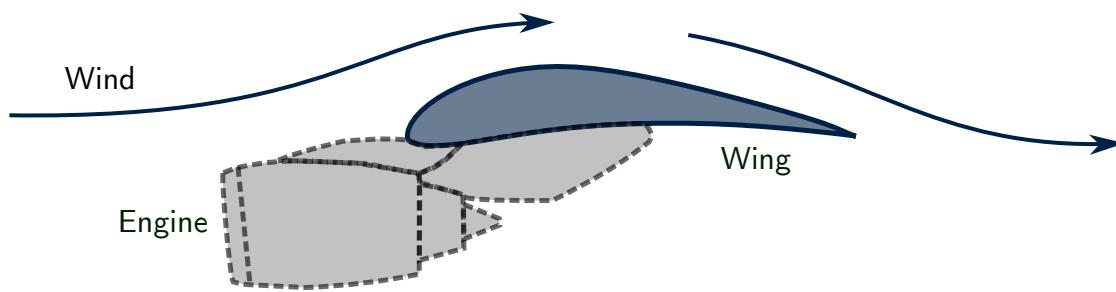
where  $m$  is the mass of the fluid parcel,  $\mathbf{U}$  is the velocity of the fluid parcel and  $\mathbf{F}$  is the sum of the external forces acting on a fluid parcel. Note the similarities between this form of the Navier-Stokes equations and Newton's second law for a solid body (equation 2). It is standard practice to divide the Navier-Stokes equations by the volume of the fluid parcel, as this is constant. This simplification leads to:

$$\frac{D(\rho\mathbf{U})}{Dt} = \mathbf{f} \quad (5)$$

where  $\rho$  is the fluid density and  $\mathbf{f}$  is the sum of the external forces per unit volume, acting on the fluid parcel. In the same manner that Newton's second law can be solved by integration to



**Figure 1:** A finite parcel/volume of fluid that forms a part of the fluid continuum.



**Figure 2:** Calculating the flow of air over a wing allows the lift and drag forces acting on the wing to be calculated.

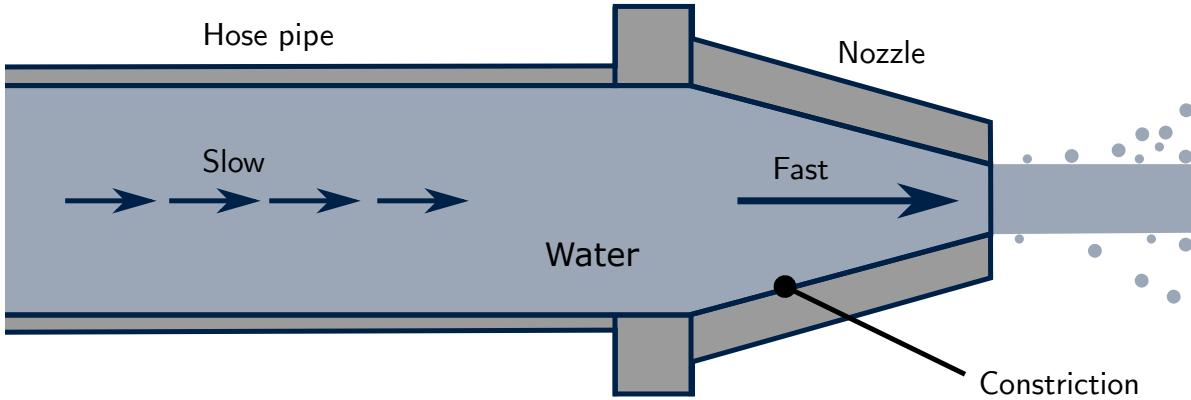
calculate the velocity of a solid body, the Navier-Stokes equations can be solved to calculate the velocity (motion) of the fluid. Once the velocity of the fluid has been determined, the forces acting on the solid surfaces that are in contact with the fluid can be computed. For example, solving the Navier-Stokes equations for the flow of air around a wing allows pressure and skin friction forces acting on the wing to be calculated (see Figure 2). These forces generate lift and drag and allow the plane to fly. Hence, solving the Navier-Stokes equations numerically (for real geometries) is of considerable interest to scientists and engineers. Solving the Navier-Stokes equations numerically will be the focus of this fundamentals course.

## Fluid Acceleration

In the Navier-Stokes equation (equation 5), the change in momentum of the fluid parcel has been written as:

$$\frac{D(\rho \mathbf{U})}{Dt} \quad (6)$$

where  $D/Dt$  is the *total derivative*. The total derivative has been used instead of the temporal derivative ( $d/dt$ ) in the Navier-Stokes equation. The reason for this change is the fluid volume may change its momentum in *time* and also change its momentum as it moves through *space*. For example, consider the flow of water through a garden hose (Figure 3), which is held at a constant flow rate. The overall flow rate of water will be constant with time if the tap is kept open at the same setting. However, the water accelerates (in space) as it moves into the nozzle. As time advances while the water moves through space, the water experiences an acceleration in time as it moves through the nozzle. The total derivative can be expanded to show the change in time and the change in space. Adopting a Cartesian coordinate system for the spatial dimensions ( $x$ ,  $y$  and  $z$ ):



**Figure 3:** The flow of water through a garden hose. The water accelerates as it moves through the nozzle due to the constriction. Even if the global flow does not change in time, a fluid parcel accelerates in time as it moves through the nozzle due to the constriction.

$$\frac{D}{Dt} = \underbrace{\frac{\partial}{\partial t}}_{\text{Time}} + \underbrace{U_x \frac{\partial}{\partial x} + U_y \frac{\partial}{\partial y} + U_z \frac{\partial}{\partial z}}_{\text{Space}} \quad (7)$$

The first term represents the change in momentum in time and the second, third and fourth terms represent the change in momentum in the  $x$ ,  $y$  and  $z$  spatial directions respectively, as the fluid parcel is convected through the flow field. In vector form, the total derivative can be written compactly as:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{U} \cdot \nabla \quad (8)$$

By using the expanded form of the total derivative, the Navier-Stokes equations can be written:

$$\frac{\partial(\rho\mathbf{U})}{\partial t} + \mathbf{U} \cdot \nabla(\rho\mathbf{U}) = \mathbf{f} \quad (9)$$

Equation 9 can be simplified slightly by applying conservation of mass and the product rule. For conciseness, the details are not included here.

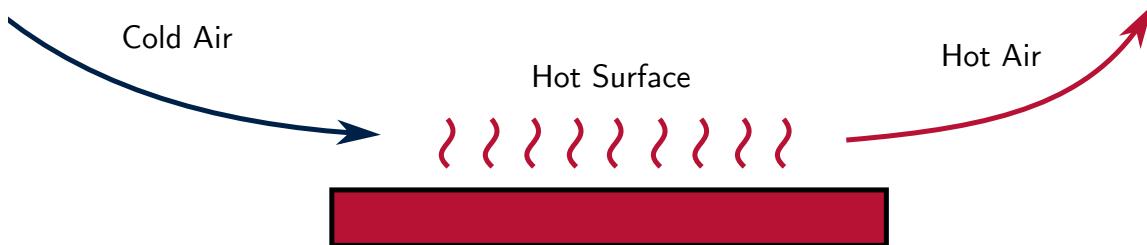
$$\frac{\partial(\rho\mathbf{U})}{\partial t} + \nabla \cdot (\rho\mathbf{U}\mathbf{U}) = \mathbf{f} \quad (10)$$

While the Navier-Stokes equations have been expanded and rewritten, their physical interpretation remains the same. The change in momentum of a fluid parcel in time is equal to the sum of the forces acting on the fluid parcel.

## External Forces

Depending on the flow condition, a variety of external forces may act to change the momentum of a fluid. Three of the most common forces that act to change the momentum of fluids are pressure, viscosity and gravity. These terms are included in the Navier-Stokes equations as forces (per unit volume) on the right hand side of the equation:

$$\frac{\partial(\rho\mathbf{U})}{\partial t} + \nabla \cdot (\rho\mathbf{U}\mathbf{U}) = \underbrace{-\nabla p}_{\text{Pressure}} + \underbrace{\nabla \cdot \boldsymbol{\tau}}_{\text{Shear Stress}} + \underbrace{\rho\mathbf{g}}_{\text{Gravity}} \quad (11)$$



**Figure 4:** The flow of cold air over a hot plate, cooling the plate.

where  $p$  is the static pressure (normal stress),  $\tau$  is the shear stress (which includes the action of viscosity) and  $\mathbf{g}$  is the acceleration due to gravity. All of the terms on the right hand-side represent forces acting on the fluid parcel, while the terms on the left hand-side represent the acceleration of the fluid parcel in response to the forces. Physically, the equations state that pressure, gravity and viscosity all act to change the momentum of the fluid ( $\rho\mathbf{U}$ ). By solving the equations numerically, the velocity of the fluid can be computed in response to these forces. Once the equations are solved, the forces acting on the solid surfaces that contact the fluid can be computed.

In this course, the *Finite Volume Method* will be examined, which is the most popular method for solving the Navier-Stokes equations numerically. By following this course, you will develop an understanding of the fundamentals of how the finite volume method can be used to solve the Navier-Stokes and other transport equations.

## Transport Equations

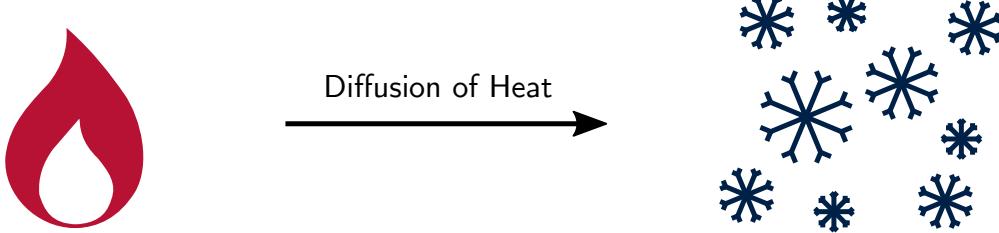
In addition to solving the Navier-Stokes equations to determine the fluid velocity ( $\mathbf{U}$ ), additional equations may need to be solved, depending on the application. For example, the fluid flow may be used to cool a hot surface, as shown in Figure 4. In this instance, the fluid transports heat away from the surface, cooling the surface. In order to determine rate of cooling of the hot surface by the fluid, the temperature (and velocity) of the fluid need to be computed. For air or water cooling at low velocity (incompressible flow), the following equation can be solved to compute the temperature  $T$  of the fluid:

$$\frac{\partial(\rho c_p T)}{\partial t} + \underbrace{\nabla \cdot (\rho c_p \mathbf{U} T)}_{\text{Convection}} = \underbrace{\nabla \cdot (k \nabla T)}_{\text{Diffusion}} + S \quad (12)$$

where  $c_p$  is the specific heat capacity of the fluid,  $k$  is the thermal conductivity of the fluid and  $S$  is a heat source (per unit fluid volume). This type of equation is called a *transport equation*, as the temperature (representing the thermal energy of the fluid) is transported by the motion of the fluid ( $\mathbf{U}$ ).

Thermal energy is transported through the fluid by two main mechanisms: convection and diffusion. Thermal energy is also transported by radiation, but this will not be considered here. The mathematical form of the convective and diffusive transport mechanisms are highlighted in equation 12. Diffusion represents the physical process where thermal energy moves from areas of high temperature to areas of low temperature due to the temperature gradient (see Figure 5). The diffusion of heat takes the following mathematical form:

$$\nabla \cdot (k \nabla T) \equiv \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) \quad (13)$$



**Figure 5:** Heat diffuses from regions of high temperature to regions of low temperature. This diffusion is represented mathematically by  $\nabla \cdot (k \nabla T)$ .

The thermal conductivity  $k$  gives the strength of diffusion. A highly conductive material (such as copper) will transfer significant quantities of heat with a small temperature gradient. On the other hand, thermal insulators (like oven gloves) have low thermal conductivity  $k$  and will transmit relatively little heat, even with a large temperature gradient.

Diffusion occurs in moving and stationary fluids, hence it does not depend on the velocity of the fluid  $\mathbf{U}$ . Diffusion is often referred to as conduction, when applied to solids. In a stationary fluid where the velocity  $\mathbf{U} = 0$ , the convection term is zero and the temperature equation reduces to:

$$\frac{\partial (\rho c_p T)}{\partial t} = \underbrace{\nabla \cdot (k \nabla T)}_{\text{Diffusion}} + S \quad (14)$$

Convection of heat is the transport of thermal energy by the motion (velocity) of the fluid. It has the following mathematical form:

$$\nabla \cdot (\rho c_p \mathbf{U} T) \equiv \frac{\partial}{\partial x} (\rho c_p T U_x) + \frac{\partial}{\partial y} (\rho c_p T U_y) + \frac{\partial}{\partial z} (\rho c_p T U_z) \quad (15)$$

The thermal energy is physically transported by the motion of the moving fluid ( $\mathbf{U}$ ). This is similar to the transport of leaves and branches that are dropped into a moving river. The leaves and branches are physically transported by the motion of the fluid and are carried along with the river. Convection increases the rate of heat transfer and is the reason why blowing over the surface of a hot drink reduces its temperature, so that we can drink it!

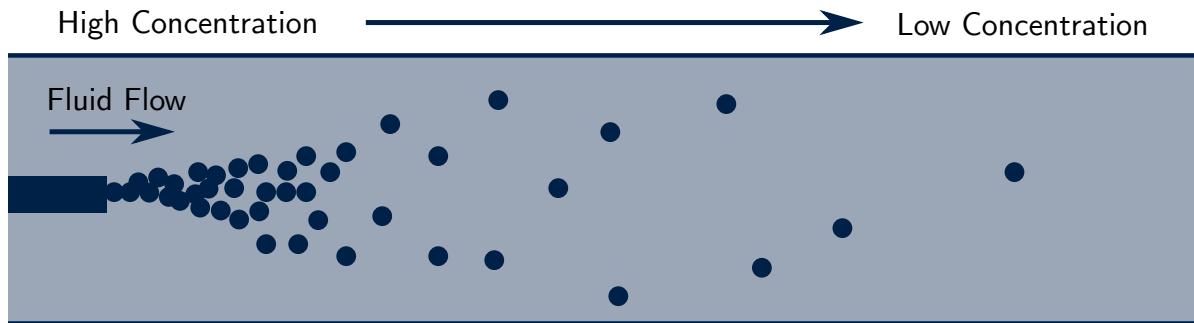
## Other Transport Equations

A variety of quantities that are transported by fluids follow a similar transport equation to the temperature/thermal energy equation. One example is the injection of dye or fine solid particles into a flow stream, as shown in Figure 6. The particles will be convected by the fluid and will also diffuse from areas of high concentration to low concentration. Hence, the concentration  $C$  of dye/solid particles follows a similar transport equation to temperature:

$$\frac{\partial (\rho C)}{\partial t} + \underbrace{\nabla \cdot (\rho \mathbf{U} C)}_{\text{Convection}} = \underbrace{\nabla \cdot (D \nabla C)}_{\text{Diffusion}} + S_c \quad (16)$$

where  $D$  is the diffusivity of the dye/solid particles. The transport equations that govern the convection and diffusion of quantities in a fluid flow (velocity, temperature, concentration etc.) all share the same common form:

$$\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \mathbf{U} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \quad (17)$$



**Figure 6:** The concentration of dye/solid particles diffuses from regions of high concentration to regions of low concentration. This diffusion is represented mathematically by  $\nabla \cdot (D \nabla C)$

where  $\phi$  is a transported quantity (velocity, temperature, concentration etc.),  $\rho$  is the fluid density,  $\Gamma$  is the diffusivity of the quantity and  $S_\phi$  is the additional source per unit volume of the quantity  $\phi$ . In this course, the finite volume method will be used to solve a general transport equation that includes convection, diffusion and a source term. As the governing equations of fluid flow all share the same general form, the same method can then be applied to any transport equation (velocity, temperature, concentration etc.) that is required.

In the three remaining chapters in this course, the finite volume method will be applied to a transport equation for temperature. Temperature has been chosen specifically in this course, as it is conceptually the most straightforward quantity to understand while applying the method. The same techniques applied in this course can then be applied to any transported quantity of interest, by following the same analysis steps. The diffusion and source terms will be considered first, to develop a general understanding of the method. The convection term will then be added in the third chapter of this course, allowing the effects of diffusion and convection to be studied simultaneously. In the fourth chapter of this course, a special technique called *upwind differencing* will be introduced. This technique is essential to solve the majority of convection-diffusion equations and is adopted by all modern CFD codes.

## 2 The 1D Diffusion Equation

In the previous chapter, the convection-diffusion equation for the transport of temperature (thermal energy) was introduced.

$$\frac{\partial(\rho c_p T)}{\partial t} + \nabla \cdot (\rho c_p \mathbf{U} T) = \nabla \cdot (\kappa \nabla T) + S \quad (18)$$

In this chapter, the transport equation for temperature will be solved for the limited case of one-dimensional (1D) steady-state diffusion. This limited case will be used to introduce the finite volume method and demonstrate how it works. The same approach can also be applied to other transport equations (momentum, species concentration, turbulence etc.). Temperature has been specifically chosen for this chapter, as it is conceptually the most straightforward to follow and understand. In the next chapter, the one-dimensional steady-state diffusion example will be extended to also include convection. Starting with the three-dimensional (3D) transport equation for thermal energy (temperature), the temporal derivative and the convection term will be neglected in this chapter.

$$\cancel{\frac{\partial(\rho c_p T)}{\partial t}}^0 + \cancel{\nabla \cdot (\rho c_p \mathbf{U} T)}^0 = \nabla \cdot (\kappa \nabla T) + S \quad (19)$$

$$0 = \nabla \cdot (\kappa \nabla T) + S \quad (20)$$

Expand the gradient ( $\nabla$ ) and dot product ( $\nabla \cdot$ ) operators in Cartesian coordinates:

$$0 = \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( \kappa \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( \kappa \frac{\partial T}{\partial z} \right) + S \quad (21)$$

For one-dimensional diffusion, the  $y$  and  $z$  derivatives are zero.

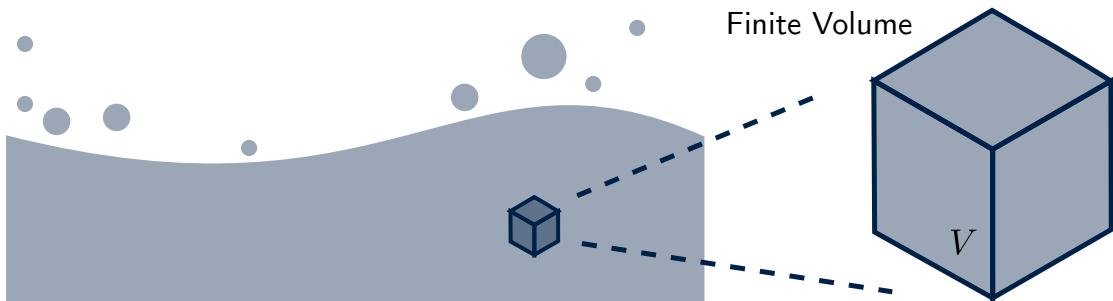
$$0 = \frac{\partial}{\partial x} \left( \kappa \frac{\partial T}{\partial x} \right) + \cancel{\frac{\partial}{\partial y} \left( \kappa \frac{\partial T}{\partial y} \right)}^0 + \cancel{\frac{\partial}{\partial z} \left( \kappa \frac{\partial T}{\partial z} \right)}^0 + S \quad (22)$$

$$\boxed{0 = \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S} \quad (23)$$

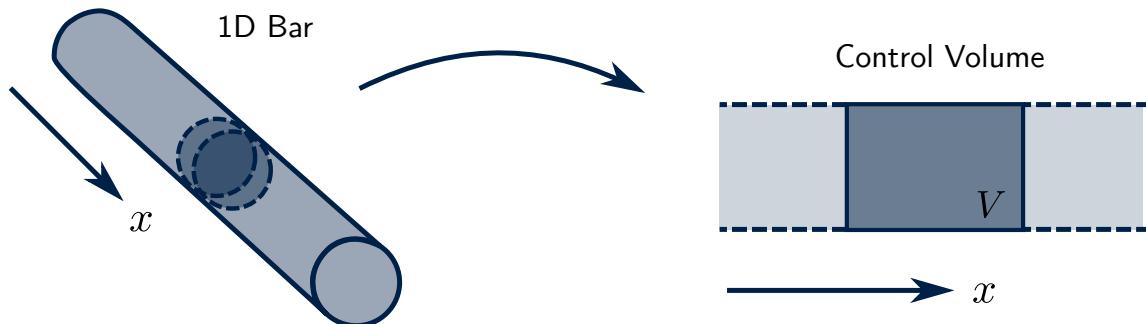
Equation 23 is the 1D steady-state heat diffusion equation. This equation will be solved using the *finite-volume method*, which is the most common approach used by modern CFD codes. The finite volume method can also be applied to more detailed equations and is not limited to one dimensional analysis. However, one dimensional flow has been specifically selected here to illustrate the principals of the method clearly.

Equation 23 is a differential equation (not an algebraic equation). Hence, the solution of this equation requires integration and the application of boundary conditions. Rather than integrate the equation over the entire domain of interest, the first stage in the finite volume method is to integrate the equation over a small piece of the domain. This piece is called a finite volume or parcel of fluid. Figure 7 shows an example of a finite volume of fluid, which forms a part of the continuum of fluid. Remember that the differential equation is valid for every finite volume of fluid in the domain, regardless of the size of the volume and its location. Mathematically, the integration process is described as:

$$\int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S \right] dV = 0 \quad (24)$$



**Figure 7:** A finite volume of fluid, which has been isolated from the fluid continuum.



**Figure 8:** A 1D finite volume of fluid with volume  $V$ , which has been isolated from the bar.

The integration of each term can be considered separately, as addition and integration are commutative operations (it does not matter which order they are carried out in).

$$\int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) \right] dV + \int_V [S] dV = 0 \quad (25)$$

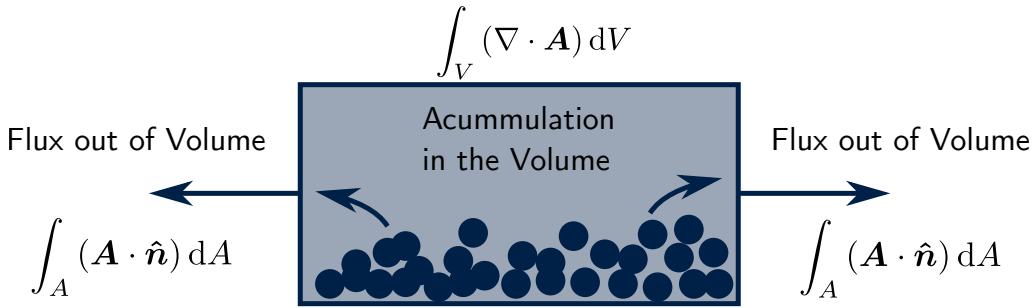
In one-dimension, the control volume forms a part of a one-dimensional geometry, as shown in Figure 8. This control volume can be thought of as a piece of a bar that is conducting heat from one-end to the other, with constant properties over its cross-section. The second term in equation 25 represents the heat source generated in the finite volume. Assume that the heat source is constant across the control volume, with a value of  $\bar{S}$  (the volume average heat source). The second term in the finite volume integral can now be simplified.

$$\int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) \right] dV + \bar{S} \int_V dV = 0 \quad (26)$$

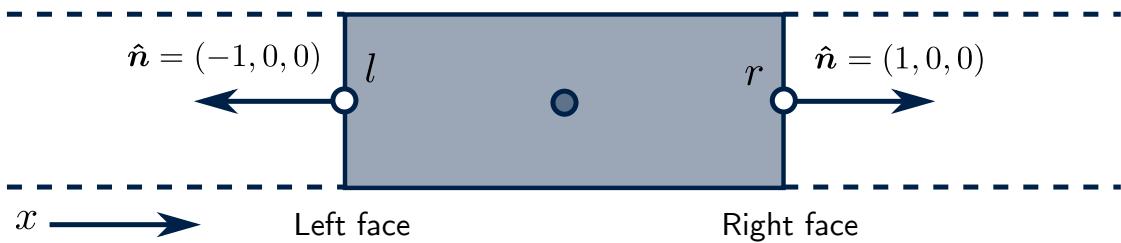
$$\int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) \right] dV + \bar{S}V = 0 \quad (27)$$

The source term  $\bar{S}$  has units of  $\text{W}/\text{m}^3$ . Hence, the product  $\bar{S}V$  has units of  $\text{W}$ .

The first term in equation 25 is the volume integral of the heat diffusion inside the control volume. To simplify and evaluate this term, the *divergence theorem* is required. Physically, the divergence theorem states that the rate of accumulation of a vector field inside a control volume is equal to the flux of the vector field across the surfaces of the control volume. When applied to the heat diffusion equation, this theorem can be thought of as an expression of conservation of energy. Heat accumulating inside the control volume by diffusion must cross the surfaces of the control volume if there are no additional sources of heat in the volume,



**Figure 9:** A diagram to show the physical significance of the divergence theorem applied to vector field  $\mathbf{A}$ . The accumulation of  $\mathbf{A}$  in the volume equals the flux of  $\mathbf{A}$  over the surfaces of the volume.



**Figure 10:** A diagram to show the face normal vectors on the left and right faces of the 1D cell. The cell normal vectors always point out of the cell.

as shown in Figure 9. Mathematically, the divergence theorem for a general vector field  $\mathbf{A}$  is written as:

$$\int_V (\nabla \cdot \mathbf{A}) dV = \int_A (\mathbf{A} \cdot \hat{\mathbf{n}}) dA \quad (28)$$

$$\int_V \left( \frac{\partial A_x}{\partial x} + \frac{\partial A_y}{\partial y} + \frac{\partial A_z}{\partial z} \right) dV = \int_A (A_x n_x + A_y n_y + A_z n_z) dA \quad (29)$$

where  $\hat{\mathbf{n}}$  is the unit normal vector pointing out of the control volume and  $A$  is the surface area of the control volume. In 1D, the divergence theorem can be written:

$$\int_V \left( \frac{dA_x}{dx} \right) dV = \int_A (A_x \hat{n}_x) dA \quad (30)$$

For the 1D heat diffusion equation  $\mathbf{A} = k \nabla T$ . Hence  $A_x = k dT/dx$ . Applying the 1D divergence theorem to the 1D heat diffusion equation leads to:

$$\int_A \left( \kappa \frac{dT}{dx} n_x \right) dA + \bar{S}V = 0 \quad (31)$$

Physically, equation 31 states that the flux of heat out of the cell by diffusion must balance the heat generated within the cell. To simplify this equation further, consider the 1D cell in Figure 10. The cell has a left face  $l$  and a right face  $r$ . Lower-case letters  $l$  and  $r$  are used in this course to refer to the left and right faces of the cell, while upper-case  $L$  and  $R$  are used to refer to the centroids of the neighbour cell that are on the left and right of the cell under consideration. The flow quantities (temperature, thermal conductivity etc.) are constant on the cell face. Hence, the first integral can be simplified:

$$\left( \kappa \frac{dT}{dx} n_x \right) \int_A dA + \bar{S}V = 0 \quad (32)$$



**Figure 11:** A comparison of interior cells (a) and boundary cells (b) in the mesh.

$$\left( k \frac{dT}{dx} n_x A \right)_r + \left( k \frac{dT}{dx} n_x A \right)_l + \bar{S}V = 0 \quad (33)$$

As shown in Figure 10,  $n_x$  is positive on the right face and negative on the left face. Hence:

$$\left( kA \frac{dT}{dx} \right)_r - \left( kA \frac{dT}{dx} \right)_l + \bar{S}V = 0 \quad (34)$$

This simplified form of the 1D heat-diffusion equation is valid for all cells in the mesh. However, it cannot be solved yet numerically, as the equation is written in terms of variables on the cell faces ( $l$  and  $r$ ). In the cell-centred finite volume method, the equation is solved in terms of variables at the cell centroids ( $L$ ,  $R$  and  $P$ ). To carry out the necessary simplification, *interior cells* and *boundary cells* need to be considered separately. As shown in Figure 11, interior cells are in the interior of the geometry and are connected to other cells. However, boundary cells are connected to a boundary of the domain (such as an inlet or wall) on one or more of their faces. In the sections that follow, the interior and boundary cells will be considered separately when simplifying equation 34.

## Interior Cells

Start with the general finite volume discretisation of the 1D heat-diffusion equation.

$$\left( kA \frac{dT}{dx} \right)_r - \left( kA \frac{dT}{dx} \right)_l + \bar{S}V = 0 \quad (35)$$

To simplify and solve this equation for the interior cells, the temperature gradient on the cell faces ( $l$  and  $r$ ) need to be expressed in terms of temperatures at the cell centroids ( $L$ ,  $R$  and  $P$ ). This simplification can be accomplished with linear interpolation, which is often called *central-differencing*. To help understand this simplification, remember that the spatial gradient of temperature can be thought of as:

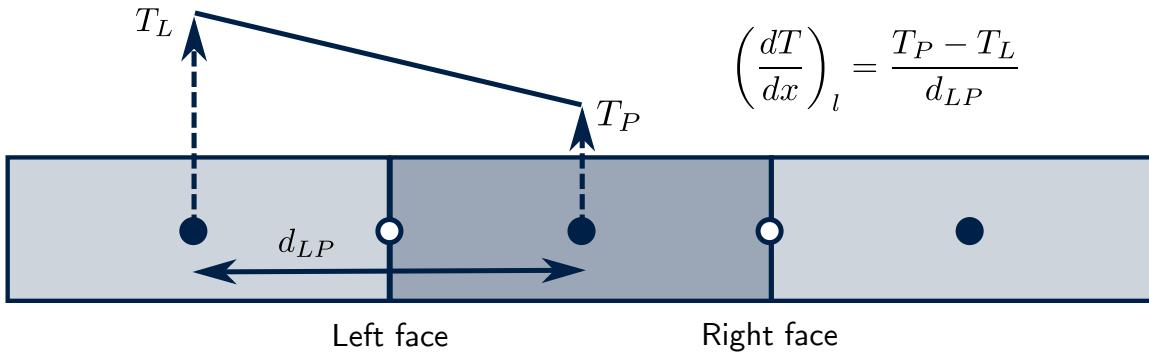
$$\frac{dT}{dx} \sim \frac{\Delta T}{\Delta x} = \frac{\text{Change in Temperature}}{\text{Distance}} \quad (36)$$

As shown in Figure 12, the temperature gradient on the left face can be expressed using central differencing as:

$$\left( \frac{dT}{dx} \right)_l = \frac{T_P - T_L}{d_{LP}} \quad (37)$$

where  $d_{LP}$  is the distance between the cell centroids  $L$  and  $P$ . In a similar manner, the temperature gradient on the right face can also be expressed using central differencing:

$$\left( \frac{dT}{dx} \right)_r = \frac{T_R - T_P}{d_{PR}} \quad (38)$$



**Figure 12:** Central differencing (linear interpolation) of the temperature gradient on the left face of the cell using the values at the cell centroids of the interior cell ( $T_P$ ) and the left cell ( $T_L$ ).

Substitute this simplification into the 1D heat-diffusion equation (equation 35).

$$\left( k_r A_r \frac{T_R - T_P}{d_{PR}} \right) - \left( k_l A_l \frac{T_P - T_L}{d_{LP}} \right) + \bar{S}V = 0 \quad (39)$$

The 1D diffusion equation can now be solved for the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ). To simplify this process, rearrange the equation and collect the terms in terms of temperature of the interior cell ( $T_P$ ), temperature of the left cell ( $T_L$ ) and the temperature of the right cell ( $T_R$ ).

$$T_P \left( \frac{k_l A_l}{d_{LP}} + \frac{k_r A_r}{d_{PR}} \right) = T_L \left( \frac{k_l A_l}{d_{LP}} \right) + T_R \left( \frac{k_r A_r}{d_{PR}} \right) + \bar{S}V \quad (40)$$

For convenience, introduce the notation  $D = k/d$ . This quantity can be thought of as the diffusive flux of heat per unit area through the cell face and has units of  $\text{W}/\text{m}^2\text{K}$ .

$$T_P (D_l A_l + D_r A_r) = T_L (D_l A_l) + T_R (D_r A_r) + \bar{S}V \quad (41)$$

For consistency with other equations that will be introduced later, write the above equation in the following form:

$$a_p T_P = a_L T_L + a_R T_R + S_u \quad (42)$$

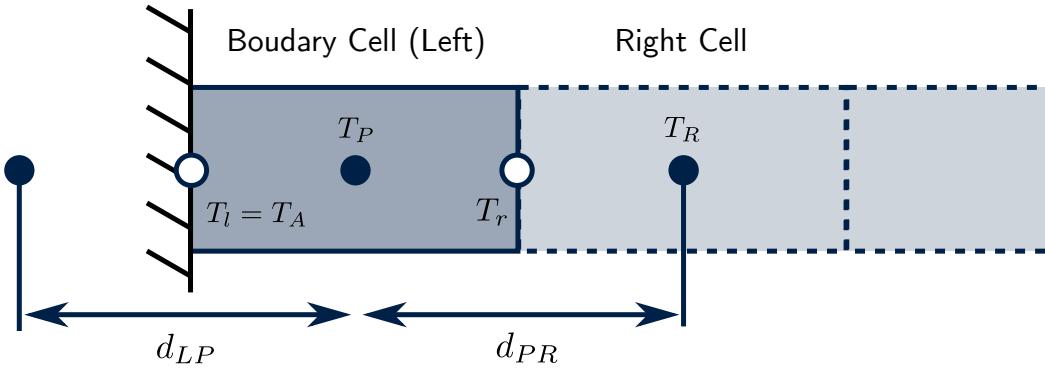
$$T_P \underbrace{(D_l A_l + D_r A_r + 0)}_{a_p} = T_L \underbrace{(D_l A_l)}_{a_L} + T_R \underbrace{(D_r A_r)}_{a_R} + \underbrace{\bar{S}V}_{S_u} \quad (43)$$

Hence, the following coefficients can be identified. These coefficients will be compared with other formulations of the convection-diffusion equation in the next two chapters.

$$a_p = a_L + a_R - S_u \quad a_L = D_l A_l \quad a_R = D_r A_r \quad (44)$$

$$S_p = 0 \quad S_u = \bar{S}V \quad (45)$$

At this stage, we now have an algebraic equation for the temperature at the centroid of the cell  $T_P$ . This is the unknown in the equation that we want to solve for. However, the temperature of the cells on the left and right of this cell ( $T_L$  and  $T_R$ ) are also unknown. To overcome this difficulty, one equation will be written for every cell in the mesh, with the unknown of each equation being the temperature of that cell centroid  $T_P$ . Each of these equations is coupled to the equations of the cells on the left and right of the cell through the variables  $T_L$  and  $T_R$ , as shown in equation 43. Before proceeding to assemble and solve these equations, separate treatment is required for the boundary cells.



**Figure 13:** The left boundary cell with temperature  $T_P$  at its centroid. The shared face between the boundary cell and the right cell is at a temperature  $T_r$  and the wall has a temperature  $T_l = T_A$ .

### Boundary Cell (Left)

Figure 13 shows the boundary cell at the left end of the bar. The cell is connected to the boundary (wall) at the left face, where a fixed temperature  $T_A$  is applied. The finite volume discretisation of the 1D heat-diffusion equation (equation 34) is:

$$\left( kA \frac{dT}{dx} \right)_r - \left( kA \frac{dT}{dx} \right)_l + \bar{S}V = 0 \quad (46)$$

The right face of the boundary cell is connected to an interior cell. Hence, the same central differencing scheme for the temperature gradient from the previous section can be used. However, the left face is connected to a boundary. As shown in Figure 13, the temperature gradient term for the left face is:

$$\left( \frac{dT}{dx} \right)_l = \frac{T_P - T_A}{d_{LP}/2} \quad (47)$$

The factor of  $1/2$  is required as the distance from the cell centroid to the face is  $1/2$  of  $d_{LP}$  (the distance from the cell centroid to the cell centroid of the adjacent cell). The finite volume discretisation of the 1D heat-diffusion equation for the left boundary cell is now:

$$\left( k_r A_r \frac{T_r - T_P}{d_{PR}} \right) - \left( k_l A_l \frac{T_P - T_A}{d_{LP}/2} \right) + \bar{S}V = 0 \quad (48)$$

Again, introduce the notation  $D = k/d$  for the diffusive heat flux per unit area.

$$T_P (2D_l A_l + D_r A_r) = T_R (D_r A_r) + T_A (2D_l A_l) + \bar{S}V \quad (49)$$

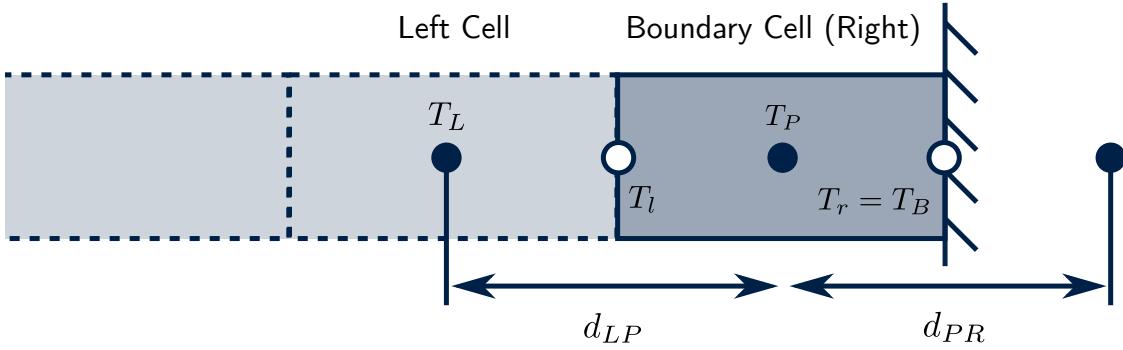
For consistency with the interior cell, write in the following form:

$$a_p T_P = a_L T_L + a_R T_R + S_u \quad (50)$$

$$T_P \underbrace{(0 + D_r A_r + 2D_l A_l)}_{a_P} = T_L \underbrace{(0)}_{a_L} + T_R \underbrace{(D_r A_r)}_{a_R} + \underbrace{T_A (2D_l A_l) + \bar{S}V}_{S_u} \quad (51)$$

For comparison with the interior cell, the boundary cell (left) has the following coefficients:

$$a_L = 0 \quad a_R = D_r A_r \quad a_p = a_L + a_R - S_u \quad (52)$$



**Figure 14:** The right boundary cell with temperature  $T_P$  at its centroid. The shared face between the boundary cell and the left cell is at a temperature  $T_l$  and the boundary has a temperature  $T_r = T_B$ .

$$S_p = -2D_l A_l \quad S_u = T_A(2D_l A_l) + \bar{S}V \quad (53)$$

Comparing these coefficients with the coefficients for the interior cell, it can be seen that the left coefficient  $a_L$  is zero. This makes sense physically, because the boundary cell is not connected to another cell on the left. The influence of the boundary condition is introduced into the equation through the source terms  $S_p$  and  $S_u$ .

### Boundary Cell (Right)

The boundary cell on the right of the domain is shown in Figure 14. The cell is connected to the boundary at the right face, where a fixed temperature  $T_B$  is applied. The finite volume discretisation of the 1D heat-diffusion equation from equation 34 is:

$$\left( kA \frac{dT}{dx} \right)_r - \left( kA \frac{dT}{dx} \right)_l + \bar{S}V = 0 \quad (54)$$

The left face of the boundary cell is connected to an interior cell. Hence, the same face interpolation schemes from the previous section can be used. However, the right face is connected to a boundary. As shown in Figure 14, the temperature gradient term for the right face is:

$$\left( \frac{dT}{dx} \right)_r = \frac{T_B - T_P}{d_{PR}/2} \quad (55)$$

The factor of  $1/2$  is required as the distance from the cell centroid to the face is  $1/2$  of  $d_{PR}$  (the distance from the cell centroid to the cell centroid of the adjacent cell). The finite volume discretisation of the 1D heat-diffusion equation is now:

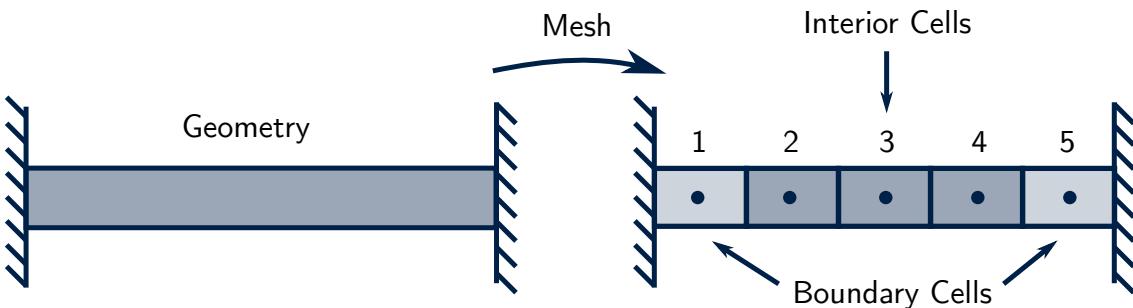
$$\left( k_r A_r \frac{T_B - T_P}{d_{PR}/2} \right) - \left( k_l A_l \frac{T_P - T_L}{d_{LP}} \right) + \bar{S}V = 0 \quad (56)$$

Again, introduce the notation  $D = k/d$  for the diffusive heat flux per unit area.

$$T_P (D_l A_l + 2D_r A_r) = T_L (D_l A_l) + T_B (2D_r A_r) + \bar{S}V \quad (57)$$

For consistency with the interior cell, write in the standard form:

$$a_p T_p = a_L T_L + a_R T_R + S_u \quad (58)$$



**Figure 15:** An example of the meshing process, where a 1D bar is divided into 5 cells/finite volumes. Cell 1 and cell 5 are boundary cells, whereas cells 2, 3 and 4 are interior cells.

$$T_P \underbrace{(D_l A_l + 0 + 2D_r A_r)}_{a_p} = T_L \underbrace{(D_l A_l)}_{a_L} + T_R \underbrace{0}_{a_R} + \underbrace{T_B (2D_r A_r) + \bar{S}V}_{S_u} \quad (59)$$

For comparison with the interior cell, the boundary cell (right) has the following coefficients:

$$a_L = D_l A_l \quad a_R = 0 \quad a_p = a_L + a_R - S_p \quad (60)$$

$$S_P = -2D_r A_r \quad S_u = T_B(2D_r A_r) + \bar{S}V \quad (61)$$

## Summary of Coefficients

A summary of the finite volume coefficients is provided in the table below for interior and boundary cells. Notice that the boundary cells have zero contribution from the cells that would extend outside of the domain. The boundary conditions are introduced through the source terms  $S_u$  and  $S_p$ .

	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (L)	0	$D_r A_r$	$a_L + a_R - S_p$	$-2D_l A_l$	$T_A(2D_l A_l) + \bar{S}V$
Interior	$D_l A_l$	$D_r A_r$	$a_L + a_R - S_p$	0	$\bar{S}V$
Boundary (R)	$D_l A_l$	0	$a_L + a_R - S_p$	$-2D_r A_r$	$T_B(2D_r A_r) + \bar{S}V$

## Meshing the Geometry

Before solving the finite volume equations, the physical geometry of interest needs to be divided into discrete cells/ volumes. This process is called *meshing* and is the most significant part of the CFD solution process because the quality of the mesh affects the accuracy and stability of the solution. The meshing process will not be examined here, as the primary aim of this course is the implementation of the finite volume method. An ideal quadrilateral mesh will be used for the 1D geometry, as shown in Figure 15. In the next section, the set of finite volume equations will be assembled for the all cells in the mesh.

## Write an Equation for Every Cell in the Mesh

Conceptually, the next stage in the finite volume method is to construct an equation for every cell in the mesh individually. The equation written for each cell is coupled to the equations

written for the neighbours of that cell. As an example, consider a 1D mesh with 5 cells, as shown in Figure 15. Cell 1 is the left boundary cell and cell 5 is the right boundary cell. Cells 2, 3 and 4 are interior cells. The individual finite volume equations are:

Cell 1	Boundary Cell (Left)	$a_{p1}T_1 = a_{R1}T_2 + S_{u1}$
Cell 2	Interior Cell	$a_{p2}T_2 = a_{L2}T_1 + a_{R2}T_3 + S_{u2}$
Cell 3	Interior Cell	$a_{p3}T_3 = a_{L3}T_2 + a_{R3}T_4 + S_{u3}$
Cell 4	Interior Cell	$a_{p4}T_4 = a_{L4}T_3 + a_{R4}T_5 + S_{u4}$
Cell 5	Boundary Cell (Right)	$a_{p5}T_5 = a_{L5}T_4 + S_{u5}$

where the coefficients  $a_p, a_L, a_R$  and  $S_u$  are given in the summary in the previous section. Notice that the interior cells are coupled to the temperature of the cells on the left and right hand side of them. In contrast, the boundary cells are only coupled to the temperature of a single cell centroid (the interior cell that they are in contact with). The boundary conditions enter the equations through the source terms  $S_u$ .

## Assemble the Matrices

To assemble the matrices, rearrange the equations by bringing all the temperature terms to the left hand side. Leave the source terms on the right hand side.

Cell 1	Boundary Cell (Left)	$a_{p1}T_1 - a_{R1}T_2 = S_{u1}$
Cell 2	Interior Cell	$-a_{L2}T_1 + a_{p2}T_2 - a_{R2}T_3 = S_{u2}$
Cell 3	Interior Cell	$-a_{L3}T_2 + a_{p3}T_3 - a_{R3}T_4 = S_{u3}$
Cell 4	Interior Cell	$-a_{L4}T_3 + a_{p4}T_4 - a_{R4}T_5 = S_{u4}$
Cell 5	Boundary Cell (Right)	$-a_{L5}T_4 + a_{p5}T_5 = S_{u5}$

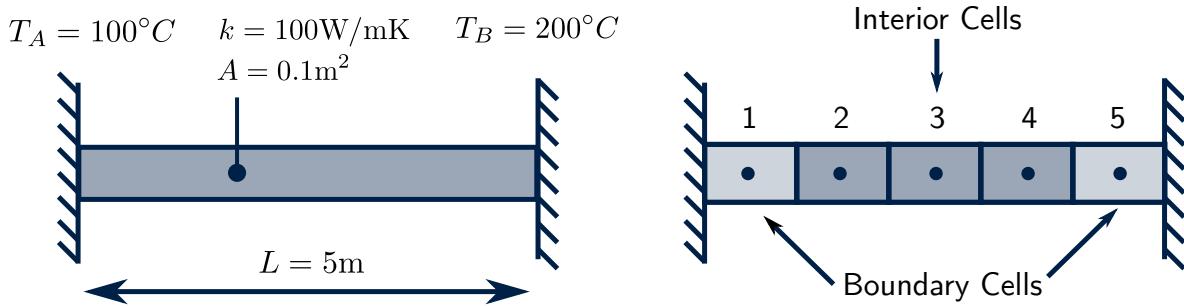
Add additional zero values for the missing temperatures in each equation.

Cell 1	Boundary Cell (Left)	$a_{p1}T_1 - a_{R1}T_2 + 0T_3 + 0T_4 + 0T_5 = S_{u1}$
Cell 2	Interior Cell	$-a_{L2}T_1 + a_{p2}T_2 - a_{R2}T_3 + 0T_4 + 0T_5 = S_{u2}$
Cell 3	Interior Cell	$0T_1 - a_{L3}T_2 + a_{p3}T_3 - a_{R3}T_4 + 0T_5 = S_{u3}$
Cell 4	Interior Cell	$0T_1 - a_{L4}T_2 + a_{p4}T_3 - a_{R4}T_4 + 0T_5 = S_{u4}$
Cell 5	Boundary Cell (Right)	$0T_1 + 0T_2 + 0T_3 - a_{L5}T_4 + a_{p5}T_5 = S_{u5}$

Write the equations in matrix form:

$$\begin{bmatrix} a_{p1} & -a_{R1} & 0 & 0 & 0 \\ -a_{L2} & a_{p2} & -a_{R2} & 0 & 0 \\ 0 & -a_{L3} & a_{p3} & -a_{R3} & 0 \\ 0 & 0 & -a_{L4} & a_{p4} & -a_{R4} \\ 0 & 0 & 0 & -a_{L5} & a_{p5} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} S_{u1} \\ S_{u2} \\ S_{u3} \\ S_{u4} \\ S_{u5} \end{bmatrix} \quad \mathbf{AT} = \mathbf{B} \quad (62)$$

which is the standard form used in linear algebra. Commercial CFD solvers populate the matrices by calculating the coefficients ( $a_l, a_p$  and  $a_r$ ) automatically for the user and then solve the matrix equations. In the next section, the entire process will be demonstrated with an example problem. A mesh will be defined, the coefficients will be calculated and then the matrices will be constructed and solved.



**Figure 16:** An example problem to demonstrate 1D heat-diffusion in a bar.

### Example Problem: Heat Diffusion in a Bar

Consider 1D steady-state diffusion of heat in a bar, as shown in Figure 16. The bar has a length of 5m, a cross-sectional area of  $0.1 \text{ m}^2$  and a thermal conductivity of  $100 \text{ W/mK}$ . The temperature at the left end of the bar ( $T_A$ ) is  $100^\circ\text{C}$  and the temperature at the right end ( $T_B$ ) is  $200^\circ\text{C}$ . There is a constant heat source of  $1000 \text{ W/m}^3$  in the bar. The temperature field in the bar is governed by the 1D steady-state diffusion equation.

$$\frac{d}{dx} \left( k \frac{dT}{dx} \right) + S = 0 \quad (63)$$

### Step 1: Divide the Geometry into a Mesh

For the example in Figure 16, divide the geometry into a mesh of 5 cells of equal length. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{5}{5} = 1\text{m} \quad (64)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d = 1\text{m} \quad (65)$$

### Step 2: Assign Material Properties

The thermal conductivity  $k$  and the cross-sectional area  $A$  are the same for every cell in the mesh. Hence, the parameter  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 \text{ [W/K]} \quad (66)$$

$$D_l A_l = D_r A_r = DA = 10 \text{ [W/K]} \quad (67)$$

The heat source per unit volume in each cell is given by:

$$\bar{SV} = \bar{S} A L_{\text{cell}} = 1000 * 0.1 * 1 = 100 \text{ [W]} \quad (68)$$

### Step 3: Calculate Matrix Coefficients

Now calculate the coefficients required to assemble the matrices. The most straightforward way is to fill in the table of coefficients:

	$a_L$	$a_R$	$S_p$	$S_u$	$a_p$
Boundary (Left)	0	10	-20	2100	30
Interior	10	10	0	100	20
Boundary (Right)	10	0	-20	4100	30

### Step 4: Assemble the Matrices

Assign the coefficients to their correct location in the matrix.

$$\begin{bmatrix} 30 & -10 & 0 & 0 & 0 \\ -10 & 20 & -10 & 0 & 0 \\ 0 & -10 & 20 & -10 & 0 \\ 0 & 0 & -10 & 20 & -10 \\ 0 & 0 & 0 & -10 & 30 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 2100 \\ 100 \\ 100 \\ 100 \\ 4100 \end{bmatrix} \quad (69)$$

### Step 5: Solve the Equations

Now that the matrices have been assembled, the matrix equation can be solved with an appropriate *iterative method*. An iterative method (such as Gauss-Seidel or Pre-conditioned Conjugate Gradient) is usually chosen by modern CFD codes, as the equations are usually too large for a *direct method* (like Gaussian Elimination) to be feasible. For example, a mesh with 1 million cells will require the solution of a matrix equation with 1 million unknowns. This is not feasible to solve in a reasonable time with a direct method. In this course, different algorithms to solve the matrix equation  $\mathbf{AT} = \mathbf{B}$  will not be considered, as details can be found in any comprehensive text on linear algebra. The default solvers for linear algebra will be used instead.

### Run the Example Problem Yourself!

Now, open either the Excel spreadsheet, the Python source code or the MATLAB source code and solve the problem yourself.

Excel	<code>solve1DDiffusionEquation.xlsx</code>
Python	<code>solve1DDiffusionEquation.py</code>
MATLAB	<code>solve1DDiffusionEquation.m</code>

Examine the calculation of the coefficients, the assembly of the matrices and run the code. You can even try changing some of the geometric and material properties of the problem

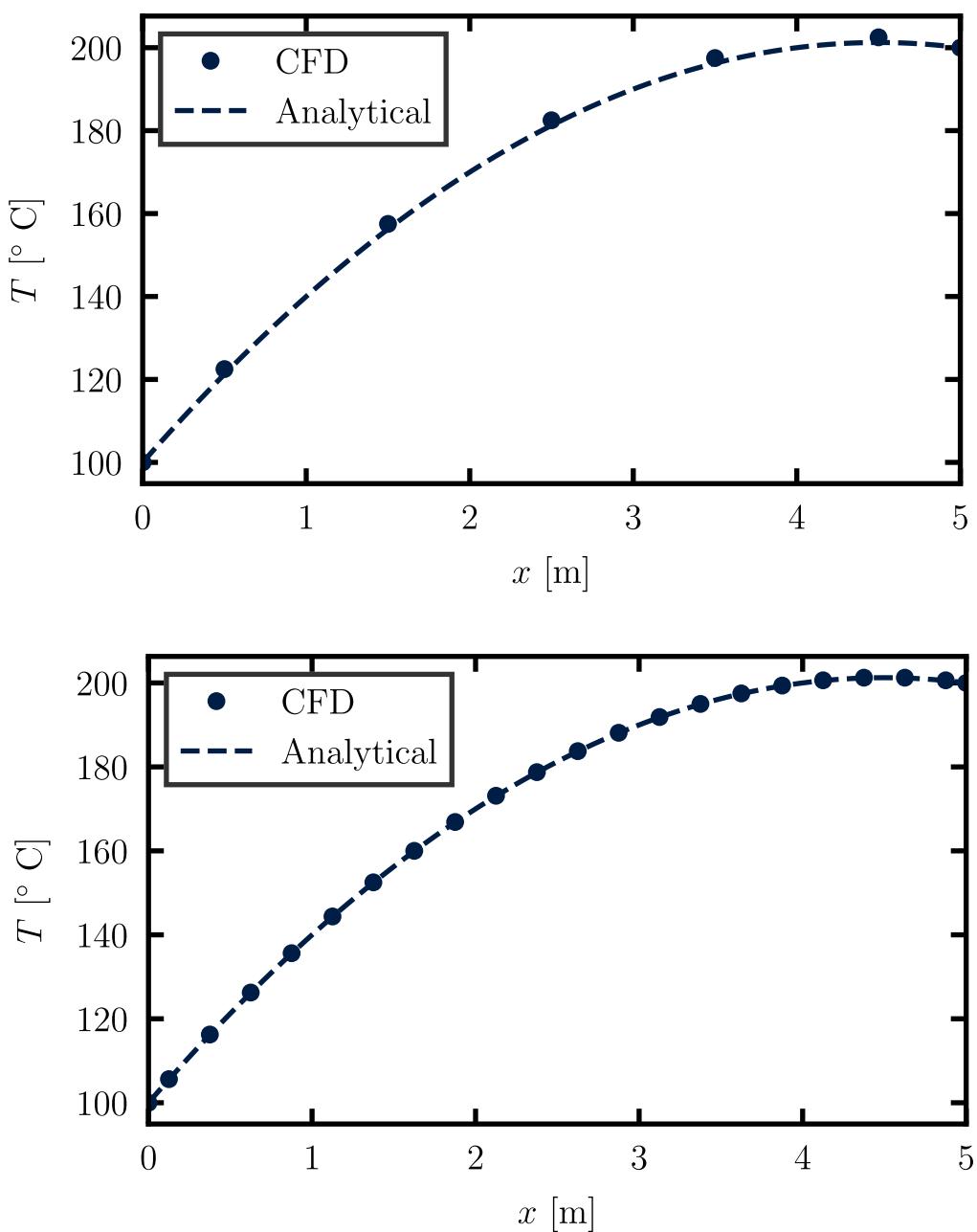
(such as the thermal conductivity or the length of the bar) and examine the changes in the solution.

### Results

The blue circles in Figure 17 show the temperature variation in the 1D bar computed with the CFD code. The dashed line shows the analytical solution of the 1D heat diffusion equation with a constant heat source ( $S$ ) which is given by:

$$T = T_A + \frac{x}{L} (T_B - T_A) + \frac{S}{2k} x (L - x) \quad (70)$$

As shown in Figure 17 (a), there is a small error between the CFD solution and the analytical solution. This is because the finite volume method assumes a *linear variation* between cells, whereas the analytical solution (for this flow scenario) is quadratic in nature. To reduce the error in the CFD solution, the mesh needs to be refined by increasing the number of cells. Figure 17 (b) shows the CFD solution of the same problem, with the number of cells increased from 5 to 20. The error in the CFD solution is reduced. However, the computational cost of the simulation has increased significantly. Hence, for practical CFD applications, a careful balance must be made between increased accuracy and increased cost of the simulations.



**Figure 17:** Temperature variation along the 1D bar for a mesh of (a) 5 cells and (b) 20 cells.

### 3 The Convection-Diffusion Equation

In the previous chapter, the steady-state diffusion equation for heat transfer was analysed.

$$0 = \nabla \cdot (k \nabla T) + S \quad (71)$$

In this chapter, the analysis will be extended to also include convective heat transfer. The steady-state equation for convective and diffusive heat transfer is:

$$\nabla \cdot (\rho c_p \mathbf{U} T) = \nabla \cdot (k \nabla T) + S \quad (72)$$

Following the same steps as the previous chapter, integrate the equation over a control volume  $V$ .

$$\int_V \nabla \cdot (\rho c_p \mathbf{U} T) dV = \int_V \nabla \cdot (k \nabla T) dV + \int_V S dV \quad (73)$$

Recall the divergence theorem from the previous chapter. For a vector field  $\mathbf{A}$ :

$$\int_V (\nabla \cdot \mathbf{A}) dV = \int_A (\mathbf{A} \cdot \hat{\mathbf{n}}) dA \quad (74)$$

Apply the divergence theorem to the convection term and the diffusion term.

$$\int_A ((\rho c_p \mathbf{U} T) \cdot \hat{\mathbf{n}}) dA = \int_A ((k \nabla T) \cdot \hat{\mathbf{n}}) dA + \int_V S dV \quad (75)$$

In the same manner as the previous chapter, the analysis will only be considered in 1D (the  $x$  direction).

$$\int_A (\rho c_p U_x T) n_x dA = \int_A k \frac{dT}{dx} n_x dA + \int_V S dV \quad (76)$$

Take the volume average of the source term over the control volume ( $\bar{S}$ ):

$$\int_A (\rho c_p U_x T) n_x dA = \int_A k \frac{dT}{dx} n_x dA + \bar{S} V \quad (77)$$

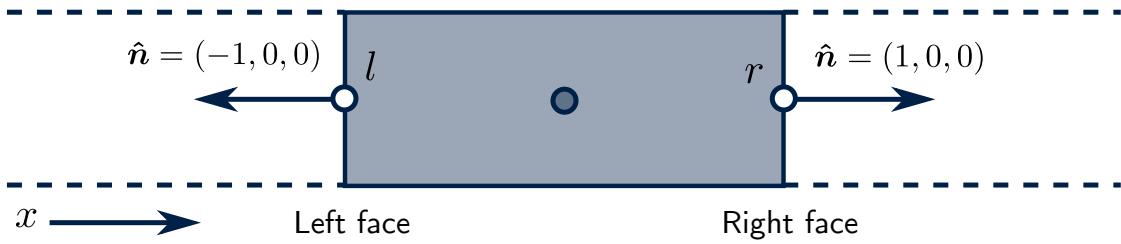
As shown in Figure 18, the 1D cells have two faces: a left face ( $l$ ) and a right face ( $r$ ). The surface integrals can be evaluated directly on the left and right faces of the cell as the fluid properties are constant across the face of the cell.

$$[\rho c_p U T n_x A]_r + [\rho c_p U T n_x A]_l = \left[ k \frac{dT}{dx} n_x A \right]_r + \left[ k \frac{dT}{dx} n_x A \right]_l + \bar{S} V \quad (78)$$

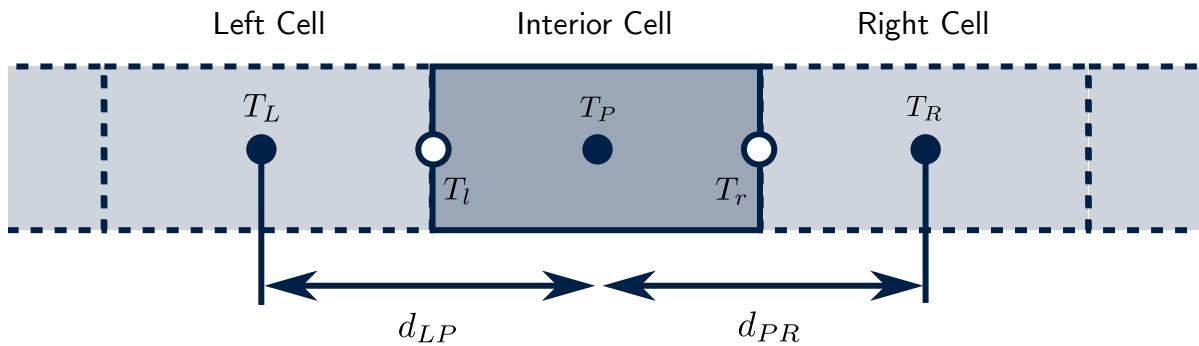
The unit normal vectors always point out of the cell. Hence  $n_x = -1$  on the left face and  $n_x = 1$  on the right face.

$$\underbrace{[\rho c_p U T A]_r - [\rho c_p U T A]_l}_{\text{Convection}} = \underbrace{\left[ k A \frac{dT}{dx} \right]_r - \left[ k A \frac{dT}{dx} \right]_l}_{\text{Diffusion}} + \bar{S} V \quad (79)$$

This simplified form of the equation is valid for all cells in the mesh. However, it cannot be solved yet numerically, as the equation is written in terms of variables on the cell faces ( $l$  and  $r$ ). In the cell-centred finite volume method, the equation is solved in terms of variables at the cell centroids ( $L$ ,  $R$  and  $P$ ). Hence, further simplification is necessary. To carry out the simplification, the *interior cells* and *boundary cells* need to be considered separately.



**Figure 18:** A diagram to show the face normal vectors on the left and right faces of the 1D cell. The cell normal vectors always point out of the cell.



**Figure 19:** An interior cell in the 1D finite volume method. The cell has two neighbours: a left cell and a right cell. The interior cell has a temperature  $T_P$  at its centroid, the left cell has a temperature  $T_L$  and the right cell has a temperature  $T_R$ . The shared face between the interior cell and the left cell is at a temperature  $T_l$  and the shared face between the interior cell and the right cell is at a temperature  $T_r$ .

## Interior Cells

Figure 19 shows an interior cell in the 1D mesh. The finite volume discretisation of the 1D convection-diffusion equation for the interior cell can be expressed as:

$$\underbrace{\rho_r c_{pr} U_r T_r A_r - \rho_l c_{pl} U_l T_l A_l}_{\text{Convection}} = \underbrace{k_r A_r \frac{T_R - T_P}{d_{PR}} - k_l A_l \frac{T_P - T_L}{d_{LP}} + \bar{S}V}_{\text{Diffusion}} \quad (80)$$

The diffusion terms and the source term (the right hand side of the equation) are identical to the previous chapter. The convection terms (the left hand side of the equation) are new in this chapter. For convenience in the analysis that follows, define the following quantities:

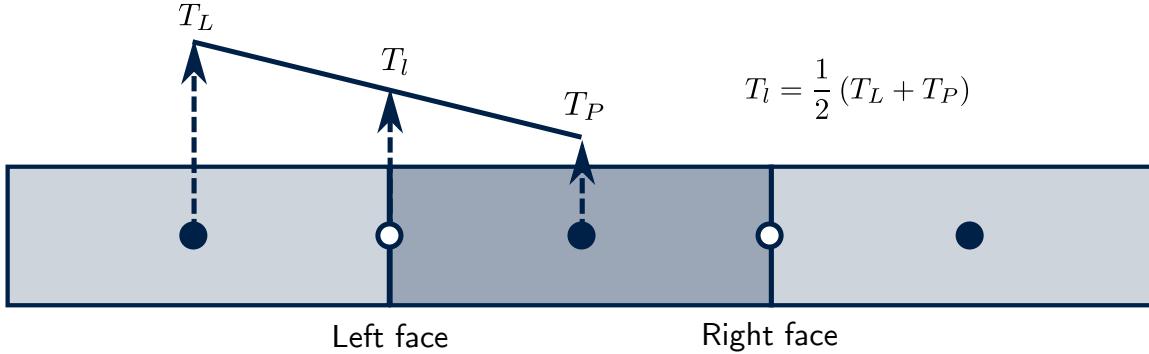
$$D = k/d \quad [\text{W}/\text{m}^2\text{K}] \quad (81)$$

$$F = \rho c_p U A \quad [\text{W}/\text{K}] \quad (82)$$

$D$  is the diffusive flux of heat through the cell face and  $F$  is the convective flux of heat through the cell face. With these new quantities, Equation 80 can simplified:

$$F_r T_r - F_l T_l = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_L) + \bar{S}V \quad (83)$$

As the unknowns in the analysis are the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ), the temperatures on the cell faces that arise in the convection term ( $T_l$ ,  $T_r$ ) are currently undefined. Hence, an interpolation scheme is required to calculate the face temperatures



**Figure 20:** Central differencing (linear interpolation) of  $T$  on the left face of the cell using the values at the cell centroids of the interior cell  $T_P$  and the left cell  $T_L$ .

from the temperatures at the cell centroids. In this chapter, *central differencing* will be used for the temperatures in the convection term. It will be shown later that in some scenarios central differencing is not appropriate for the convection term and an alternative interpolation scheme is required.

When using central differencing for the convection term, the temperatures on the left and right faces of the cell are given by (see Figure 20):

$$T_l = \frac{1}{2} (T_L + T_P) \quad (84)$$

$$T_r = \frac{1}{2} (T_P + T_R) \quad (85)$$

Substitute  $T_l$  and  $T_r$  into equation 83.

$$\frac{F_r}{2} (T_P + T_R) - \frac{F_l}{2} (T_L + T_P) = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_L) + \bar{S}V \quad (86)$$

Rearrange and collect the terms that are associated with the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_p \left[ D_l A_l + D_r A_r + \frac{F_r}{2} - \frac{F_l}{2} \right] = T_L \left[ D_l A_l + \frac{F_l}{2} \right] + T_R \left[ D_r A_r - \frac{F_r}{2} \right] + \bar{S}V \quad (87)$$

Rearrange the first term on the left-hand side slightly, for convenience:

$$\begin{aligned} & T_P \underbrace{\left[ \left( D_l A_l + \frac{F_l}{2} \right) + \left( D_r A_r - \frac{F_r}{2} \right) + (F_r - F_l) \right]}_{a_p} = \\ & T_L \underbrace{\left[ D_l A_l + \frac{F_l}{2} \right]}_{a_L} + T_R \underbrace{\left[ D_r A_r - \frac{F_r}{2} \right]}_{a_R} + \bar{S}V \end{aligned} \quad (88)$$

The finite volume equation is now in the standard form:

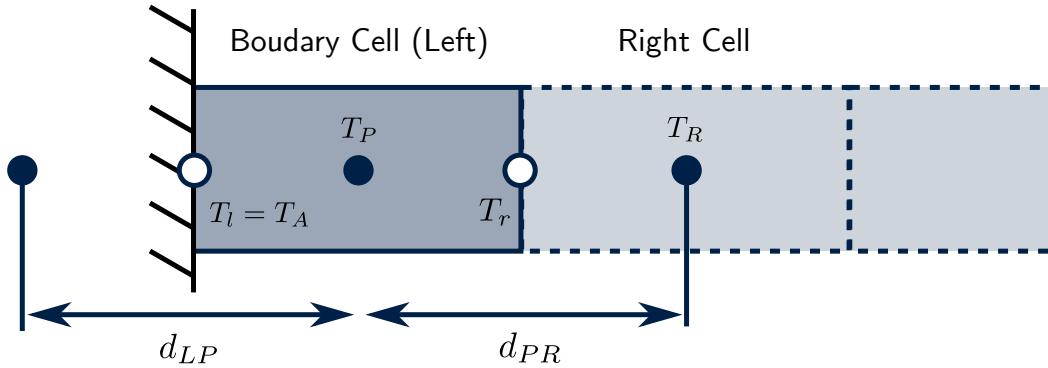
$$a_p T_P = a_L T_L + a_R T_R + S_u \quad (89)$$

with the following coefficients:

$$a_L = D_l A_l + \frac{F_l}{2} \quad a_R = D_r A_r - \frac{F_r}{2} \quad (90)$$

$$a_p = a_L + a_R + (F_r - F_l) - S_p \quad S_p = 0 \quad S_u = \bar{S}V \quad (91)$$

As a quick check, setting  $F_r$  and  $F_l = 0$  (no convection) results in the same coefficients that were derived in the previous chapter for the diffusion equation.



**Figure 21:** The left boundary cell with a temperature  $T_P$  at its centroid. The shared face between the boundary cell and the right cell is at a temperature  $T_r$  and the boundary has a temperature  $T_A$ .

## Boundary Cell (Left)

All cells in the mesh (boundary cells and interior cells) obey the same discretised form of the convection-diffusion equation:

$$[\rho c_p U T A]_r - [\rho c_p U T A]_l = \left[ kA \frac{dT}{dx} \right]_r - \left[ kA \frac{dT}{dx} \right]_l + \bar{S}V \quad (92)$$

In the same manner as the previous chapter, the temperature of the left wall is fixed at  $T_A$ . Furthermore, the distance between the cell centroids ( $d_{LP}$ ) that is used in the diffusion term is halved, as the distance from the cell centroid to the boundary is half the distance to the next cell (see Figure 21). It follows that the finite volume discretisation of the convection-diffusion equation for the left boundary cell is:

$$F_r T_r - F_l T_A = D_r A_r (T_r - T_P) - 2D_l A_l (T_P - T_A) + \bar{S}V \quad (93)$$

Use central differencing for the temperature on the right face:

$$T_r = \frac{1}{2} (T_P + T_R) \quad (94)$$

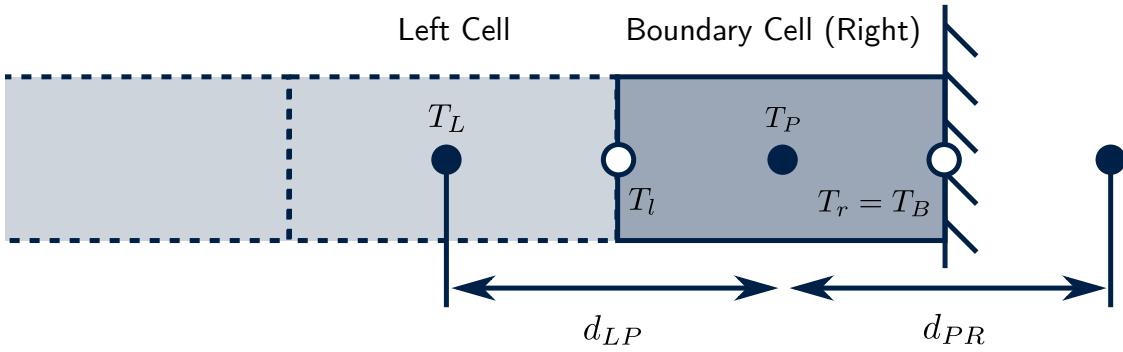
$$\frac{F_r}{2} (T_P + T_R) - F_l T_A = D_r A_r (T_r - T_P) - 2D_l A_l (T_P - T_A) + \bar{S}V \quad (95)$$

Rearrange in terms of temperatures at the cell-centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P \left[ 2D_l A_l + D_r A_r + \frac{F_r}{2} \right] = T_R \left[ D_r A_r - \frac{F_r}{2} \right] + T_A [2D_l A_l + F_l] + \bar{S}V \quad (96)$$

Rearrange the first term on the left-hand side slightly for convenience.

$$\begin{aligned} T_P \underbrace{\left[ 0 + \left( D_r A_r - \frac{F_r}{2} \right) + (F_r - F_l) + (2D_l A_l + F_l) \right]}_{a_p} &= \\ T_L \underbrace{[0]}_{a_L} + T_R \underbrace{\left[ D_r A_r - \frac{F_r}{2} \right]}_{a_R} + \underbrace{T_A [2D_l A_l + F_l] + \bar{S}V}_{S_u} & \end{aligned} \quad (97)$$



**Figure 22:** The right boundary cell with a temperature  $T_P$  at its centroid. The shared face between the boundary cell and the left cell is at a temperature  $T_l$  and the boundary has a temperature  $T_B$ .

The equation is now in standard form:

$$a_p T_P = a_l T_L + a_r T_R + S_u \quad (98)$$

with the following coefficients:

$$a_L = 0 \quad a_R = D_r A_r - \frac{F_r}{2} \quad a_p = a_l + a_r + (F_r - F_l) - S_p \quad (99)$$

$$S_p = -(2D_l A_l + F_l) \quad S_u = T_A (2D_l A_l + F_l) + \bar{S}V \quad (100)$$

## Boundary Cell (Right)

The boundary cell on the right side of the domain is shown in Figure 22. Start with the 1D finite volume discretisation of the convection-diffusion equation:

$$[\rho c_p UTA]_r - [\rho c_p UTA]_l = \left[ kA \frac{dT}{dx} \right]_r - \left[ kA \frac{dT}{dx} \right]_l + \bar{S}V \quad (101)$$

The temperature on the right boundary is fixed at  $T_B$ . Hence, the temperature on the right face  $T_r = T_B$ . Furthermore, the distance between the cell centroids ( $d_{PR}$ ) that is used in the diffusion term is halved, as the distance from the cell centroid to the boundary is half the distance to the next cell (see Figure 22). It follows that the finite volume discretisation of the convection-diffusion equation for the right boundary cell is:

$$F_r T_B - F_l T_l = 2D_r A_r (T_B - T_P) - D_l A_l (T_P - T_l) + \bar{S}V \quad (102)$$

Use central differencing for the temperature on the left face:

$$T_l = \frac{1}{2} (T_L + T_P) \quad (103)$$

$$F_r T_B - \frac{F_l}{2} (T_L + T_P) = 2D_r A_r (T_B - T_P) - D_l A_l (T_P - T_l) + \bar{S}V \quad (104)$$

Rearrange in terms of temperatures at the cell-centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P \left[ D_l A_l - \frac{F_l}{2} - 2D_r A_r \right] = T_L \left[ D_l A_l + \frac{F_l}{2} \right] + T_B [2D_r A_r - F_r] + \bar{S}V \quad (105)$$

Rearrange the first term on the left-hand side slightly for convenience.

$$T_P \underbrace{\left[ \left( D_l A_l + \frac{F_l}{2} \right) + 0 + (F_r - F_l) + (2D_r A_r - F_r) \right]}_{a_p} = \quad (106)$$

$$T_L \underbrace{D_l A_l + \frac{F_l}{2}}_{a_L} + T_R \underbrace{[0]}_{a_R} + \underbrace{T_B [2D_r A_r - F_r] + \bar{S}V}_{S_u} \quad (107)$$

The equation is now in standard form:

$$a_p T_P = a_L T_L + a_R T_R + S_u \quad (108)$$

with the following coefficients:

$$a_L = D_l A_l + \frac{F_l}{2} \quad a_R = 0 \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (109)$$

$$S_p = -(2D_r A_r - F_r) \quad S_u = T_B (2D_r A_r - F_r) + \bar{S}V \quad (110)$$

## Coefficient Summary

	$a_L$	$a_R$	$S_p$	$S_u$
Interior	$D_l A_l + \frac{F_l}{2}$	$D_r A_r - \frac{F_r}{2}$	0	$\bar{S}V$
Boundary (L)	0	$D_r A_r - \frac{F_r}{2}$	$-(2D_l A_l + F_l)$	$T_A (2D_l A_l + F_l) + \bar{S}V$
Boundary (R)	$D_l A_l + \frac{F_l}{2}$	0	$-(2D_r A_r - F_r)$	$T_B (2D_r A_r - F_r) + \bar{S}V$

And the central coefficient is given by:

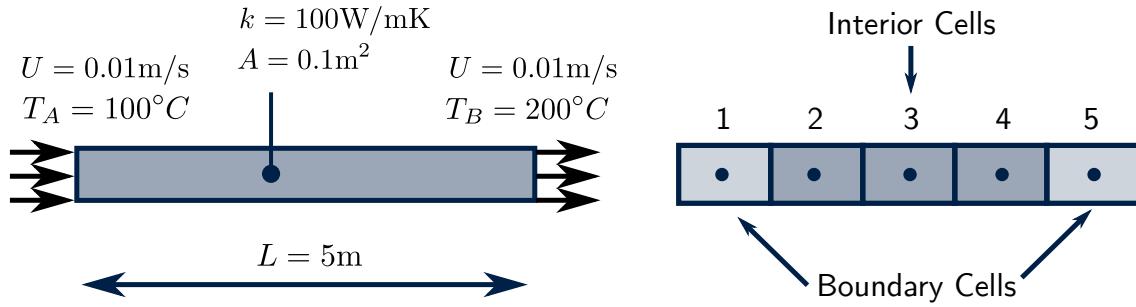
$$a_p = a_L + a_R + (F_r - F_l) - S_p \quad (111)$$

These coefficients are identical to the coefficients for the diffusion equation from the previous chapter, with the exception of the mass fluxes across the left ( $F_l$ ) and right ( $F_r$ ) faces. Setting the mass fluxes to zero results in identical coefficients to the 1D diffusion equation from the previous chapter.

## Example Problem: Conduction and Convection in a bar

For the worked example in this chapter, the same example from the previous chapter will be used. However, the temperature field is now also convected from left to right with a constant velocity of 0.01 m/s. Figure 23 shows a diagram of 1D convection and diffusion of heat in the bar. The bar has a length of 5m, a cross-sectional area of 0.1 m<sup>2</sup>, a thermal conductivity of 100 W/mK and a specific heat capacity of 1000 J/Kg K. The temperature at the left end of the bar ( $T_A$ ) is 100°C and the temperature at the right end ( $T_B$ ) is 200°C. There is a constant source of heat of 1000 W/m<sup>3</sup> in the bar. The temperature field in the bar is governed by the 1D steady-state convection-diffusion equation.

$$\frac{d}{dx} (\rho c_p U T) = \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S \quad (112)$$



**Figure 23:** An example problem to demonstrate 1D convection and diffusion in a bar.

### Step 1: Divide the Geometry into a Mesh

For the example in Figure 23, divide the geometry into a mesh of  $N = 5$  cells of equal length. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{5}{5} = 1\text{m} \quad (113)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d = 1\text{m} \quad (114)$$

### Step 2: Assign Material Properties

The thermal conductivity  $k$  and the cross-sectional area  $A$  are the same for every cell in the mesh. Hence, the parameter  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 \text{ [W/K]} \quad (115)$$

$$D_l A_l = D_r A_r = DA = 10 \text{ [W/K]} \quad (116)$$

The convective heat flux through the cell faces is given by:

$$F = \rho c_p U A = 1.0 * 1000 * 0.01 * 0.1 = 1.0 \text{ [W/K]} \quad (117)$$

$$F_l = F_r = F = 1.0 \text{ [W/K]} \quad (118)$$

The heat source in each cell is given by:

$$\bar{S}V = \bar{S}A L_{\text{cell}} = 1000 * 0.1 * 1 = 100 \text{ [W]} \quad (119)$$

### Step 3: Calculate Matrix Coefficients

Now calculate the coefficients required to assemble the matrices. The most straightforward way is to fill in the table of coefficients:

	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (Left)	0	9.5	30.5	-21	2200
Interior	10.5	9.5	20	0	100
Boundary (Right)	10.5	0	29.5	-19	3900

## Step 4: Assemble the Matrices

Assign the coefficients to their correct location in the matrix.

$$\begin{bmatrix} 30.5 & -9.5 & 0 & 0 & 0 \\ -10.5 & 20 & -9.5 & 0 & 0 \\ 0 & -10.5 & 20 & -9.5 & 0 \\ 0 & 0 & -10.5 & 20 & -9.5 \\ 0 & 0 & 0 & -10.5 & 29.5 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 2200 \\ 100 \\ 100 \\ 100 \\ 3900 \end{bmatrix} \quad (120)$$

## Step 5: Solve the Equations

Now that the matrices have been assembled, the matrix equation can be solved with an appropriate *iterative method*. Popular algorithms include Geometric Algebraic Multi-grid (GAMG) and Preconditioned Conjugate Gradient (PCG). However, as with the previous chapter, these algorithms will not be considered in detail here.

## Run the Example Problem Yourself!

Now, open either the Excel spreadsheet, the Python source code or the MATLAB code and solve the problem yourself.

Excel      *solve1DConvectionDiffusionEquation.xlsx*

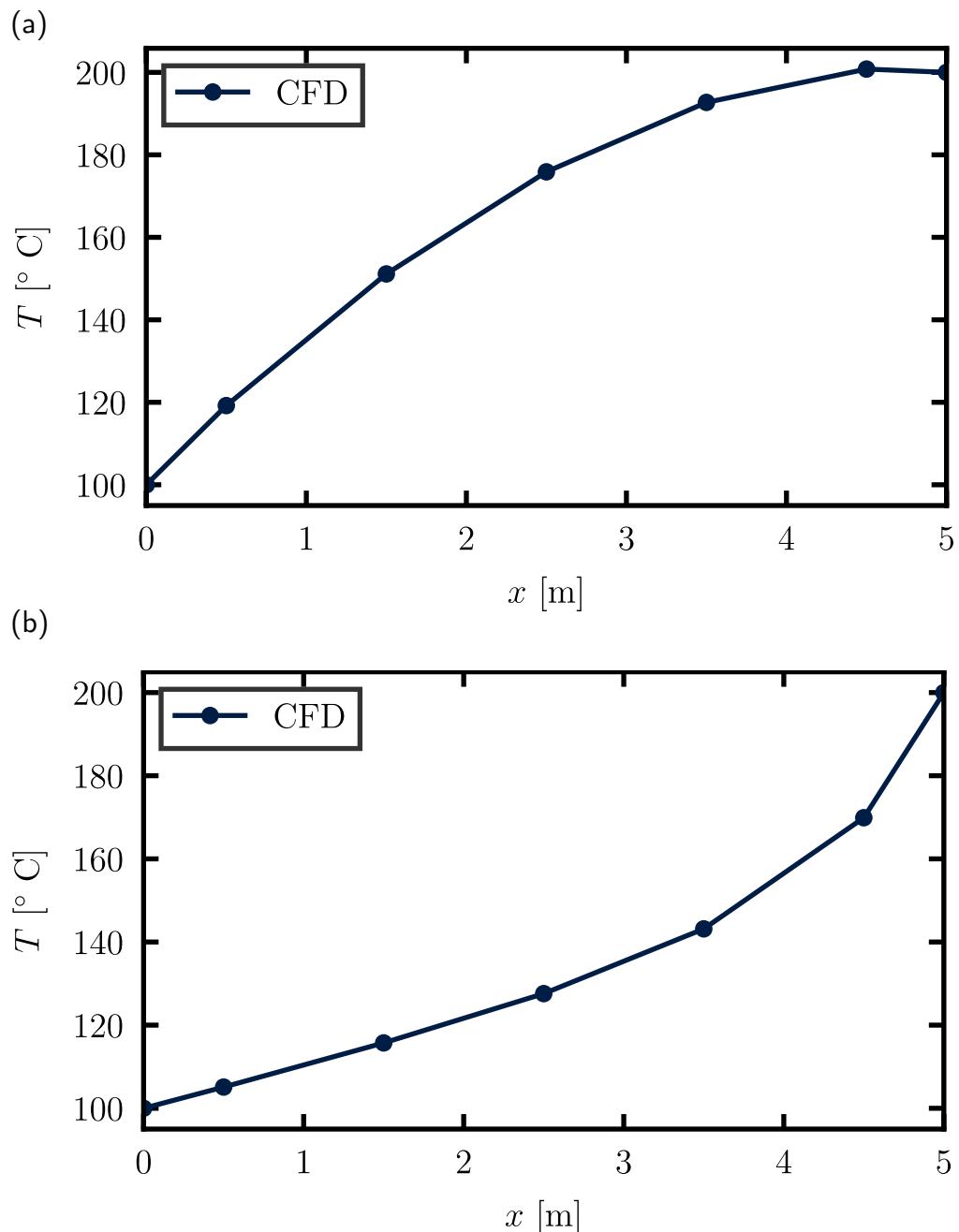
Python      *solve1DConvectionDiffusionEquation.py*

MATLAB      *solve1DConvectionDiffusionEquation.m*

Examine the calculation of the coefficients, the assembly of the matrices and run the code. You can even try changing some of the geometric and material properties of the problem (such as the thermal conductivity or the length of the bar) and examine the changes in the solution.

## Results

The temperature variation in the bar with a flow velocity of 0.01 m/s is shown in Figure 24 (a). The temperature profile is almost identical to the temperature profile from the previous chapter (where convection was not considered). This is because the strength of the convection is relatively small in comparison with the strength of diffusion in the bar.



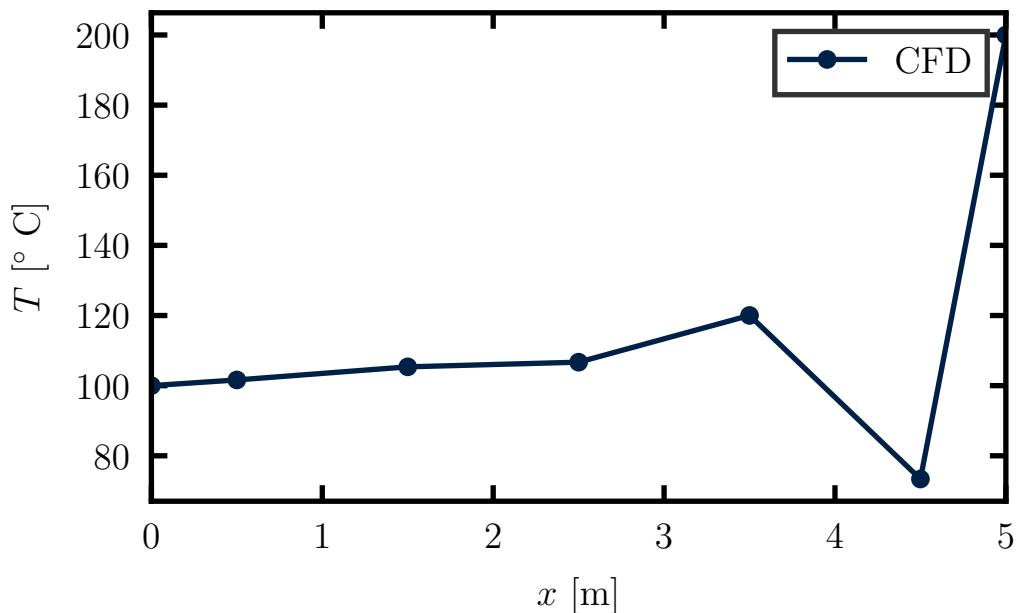
**Figure 24:** Temperature variation along the bar for a flow velocity of (a) 0.01 m/s and (b) 0.1 m/s.

Figure 24 (b) shows the effect of increasing the flow velocity from 0.01 m/s to 0.1 m/s. The effect of convection is now much stronger and the profile is noticeably shifted to the right by the flow velocity (which is flowing from left to right). The relative strength of the convective and diffusive flux of heat through the cell is given by the cell Peclet number ( $Pe$ ).

$$Pe = \frac{\text{Convective Flux}}{\text{Diffusive Flux}} = \frac{F}{DA} = \frac{\rho c_p U A}{k A / d} \quad (121)$$

For a flow velocity of 0.01 m/s,  $Pe$  is 0.1. This indicates that the diffusive heat flux through the cell is much stronger than the convective heat flux. Increasing the flow velocity to 0.1

m/s increases  $Pe$  to 1. This indicates that the relative strength of convective and diffusive heat flux through the cell are approximately the same and the effect on the temperature profile is noticeable in Figure 24 (b). However, further increasing the flow velocity to 0.3 m/s increases  $Pe$  to 3. At this stage, the solution exhibits non-physical oscillations, as shown in Figure 25. The reason for these oscillations is the *central-differencing* scheme that has been used for the convection term is unstable when  $Pe > 2$ . Hence, an alternative discretisation scheme is required that does not lead to these instabilities. One common choice is the *upwind differencing scheme*, which will be considered in the next chapter.



**Figure 25:** Temperature variation along the bar with a flow velocity of 0.3 m/s ( $Pe = 3$ ).

## 4 Upwind Differencing

In this chapter, the 1D steady-state convection-diffusion equation will be considered again. However, an *upwind differencing* scheme will be applied to the convection term rather than a *central differencing* scheme (which was applied in the previous chapter). As a reminder, the 1D convection-diffusion equation is:

$$\frac{d}{dx} (\rho c_p U T) = \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S \quad (122)$$

The finite volume discretisation of this equation for a 1D cell is:

$$[\rho c_p U T A]_r - [\rho c_p U T A]_l = \left[ k A \frac{dT}{dx} \right]_r - \left[ k A \frac{dT}{dx} \right]_l + \bar{S}V \quad (123)$$

In the previous chapter, central differencing (linear interpolation) was used to calculate the temperature of the left ( $T_l$ ) and right ( $T_r$ ) faces of the cell.

$$T_l = \frac{1}{2} (T_L + T_P) \quad (124)$$

$$T_r = \frac{1}{2} (T_P + T_R) \quad (125)$$

However, when the Peclet number  $Pe$  is  $> 2$ , non-physical oscillations are generated in the solution. An alternative to the central differencing scheme which does not lead to non-physical oscillations is an *upwind* scheme. With an upwind scheme, the temperature on the cell face takes the value of the *upwind* cell centroid. More specifically, this means that the value on the cell face will take the value at the cell centroid in the direction that the flow is coming from. As an example, consider the left face of an interior cell in the mesh, as shown in Figure 26. When the flow is left to right,  $U > 0$  and therefore  $F > 0$ . It follows that the centroid of the left cell is upwind of the face and the temperature on the face  $T_l = T_L$ . Conversely, when the flow is right to left,  $U < 0$  and therefore  $F < 0$ . It follows that the centroid of the interior cell is upwind of the left face and the temperature of the face  $T_l = T_P$ . These two cases can be written concisely as:

$$T_l = \begin{cases} T_L & F_l > 0 \\ T_P & F_l < 0 \end{cases} \quad (126)$$

Now consider the right face of the cell, as shown in Figure 27. The temperature on this face of the cell is:

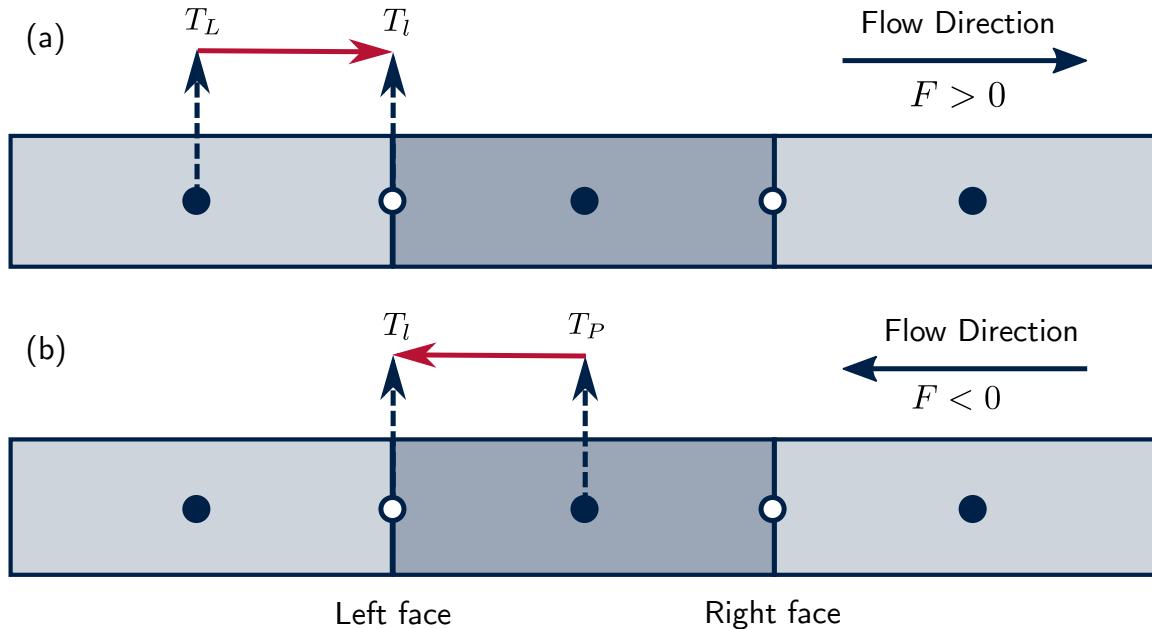
$$T_r = \begin{cases} T_P & F_l > 0 \\ T_R & F_l < 0 \end{cases} \quad (127)$$

As with the previous chapters, separate discretised equations will now be written for the interior and boundary cells. Once these equations have been written, the coefficients will be calculated and the matrices assembled, ready for solving.

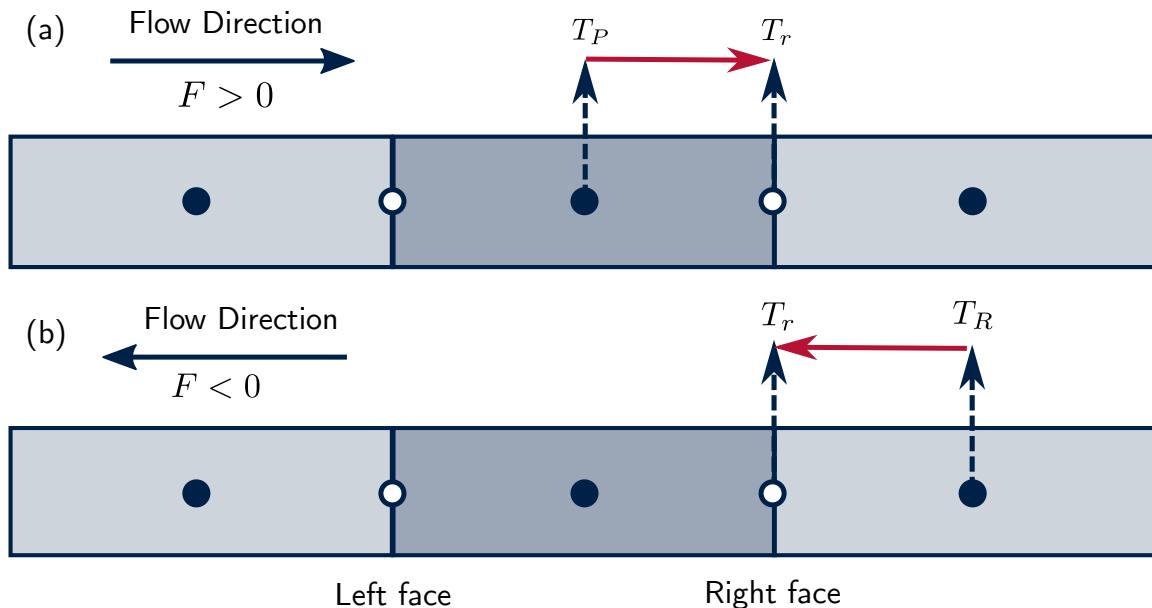
### Interior Cells

Start with the general finite volume discretisation of the 1D steady-state convection-diffusion equation.

$$[\rho c_p U T A]_r - [\rho c_p U T A]_l = \left[ k A \frac{dT}{dx} \right]_r - \left[ k A \frac{dT}{dx} \right]_l + \bar{S}V \quad (128)$$



**Figure 26:** A diagram to show the temperature on the left face of the cell ( $T_l$ ) computed using upwind differencing when the flow direction is (a) left to right and (b) right to left.



**Figure 27:** A diagram to show the temperature on the right face of the cell  $T_r$  computed using upwind differencing when the flow direction is (a) left to right and (b) right to left.

As before, introduce the notation  $D = k/d$  and  $F = \rho c_p U A$  for convenience.

$$F_r T_r - F_l T_l = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_l) + \bar{S}V \quad (129)$$

As an upwind differencing scheme is being used in this chapter for the convection term, two different cases will be considered. Firstly, the case where the flow is left to right will be considered, then the case where the flow is right to left.

### (a) Flow from Left to Right ( $F > 0$ )

Using Figure 26 (a) and Figure 27 (a), the finite volume discretisation becomes:

$$F_r T_P - F_l T_L = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_L) + \bar{S}V \quad (130)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_l A_l + D_r A_r + F_r] = T_L [D_l A_l + F_l] + T_R [D_l A_l] + \bar{S}V \quad (131)$$

Rearrange the first term on the left-hand side slightly.

$$T_P \underbrace{[D_l A_l + F_l + D_r A_r + (F_r - F_l)]}_{a_p} = T_L \underbrace{[D_l A_l + F_l]}_{a_L} + T_R \underbrace{[D_r A_r]}_{a_R} + \underbrace{\bar{S}V}_{S_u} \quad (132)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (133)$$

$$a_L = D_l A_l + F_l \quad a_R = D_r A_r \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (134)$$

$$S_p = 0 \quad S_u = \bar{S}V \quad (135)$$

### (b) Flow from Right to Left ( $F < 0$ )

Using Figure 26 (b) and Figure 27 (b), the finite volume discretisation becomes:

$$F_r T_R - F_l T_P = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_L) + \bar{S}V \quad (136)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_l A_l + D_r A_r - F_l] = T_L [D_l A_l] + T_R [D_r A_r - F_r] + \bar{S}V \quad (137)$$

Rearrange the first term on the left-hand side slightly.

$$T_P \underbrace{[D_l A_l + D_r A_r - F_r + (F_r - F_l)]}_{a_P} = T_L \underbrace{[D_l A_l]}_{a_L} + T_R \underbrace{[D_r A_r - F_r]}_{a_R} + \underbrace{\bar{S}V}_{S_u} \quad (138)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (139)$$

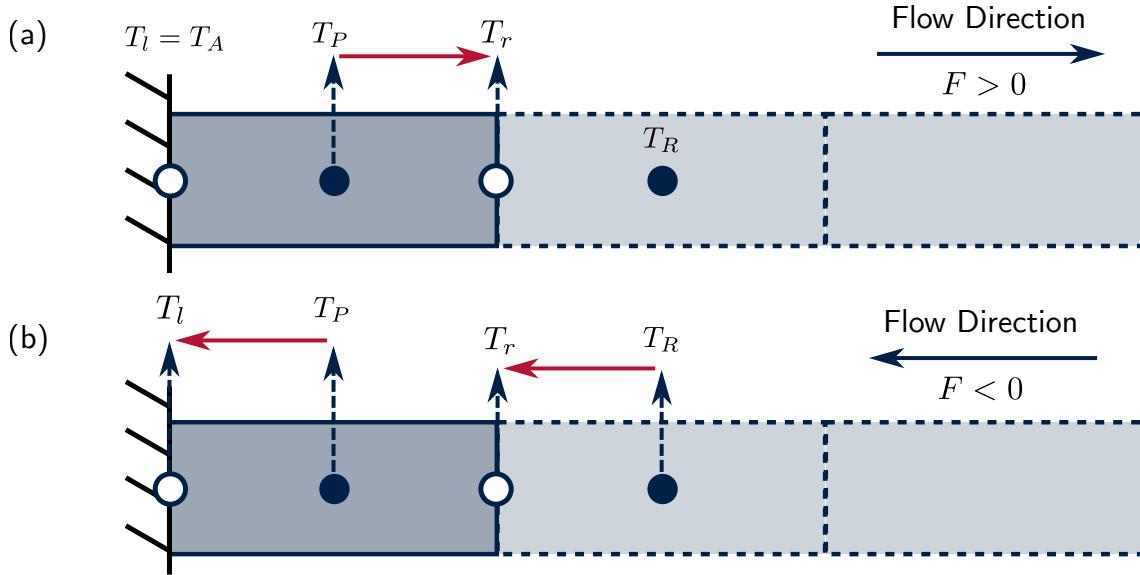
$$a_L = D_l A_l \quad a_R = D_r A_r - F_r \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (140)$$

$$S_p = 0 \quad S_u = \bar{S}V \quad (141)$$

### (c) Flow From Either Direction

The finite volume discretisation for the interior cell can be written concisely (accounting for flow from either direction) as:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (142)$$



**Figure 28:** A diagram of the temperature on the left and right faces of the boundary cell (left) computed using upwind differencing when the flow direction is (a) left to right and (b) right to left.

with the following coefficients:

$$a_L = D_l A_l + \max(F_l, 0) \quad (143)$$

$$a_R = D_R A_R + \max(-F_r, 0) \quad (144)$$

$$a_P = a_L + a_R + (F_r - F_l) - S_p \quad (145)$$

$$S_P = 0 \quad (146)$$

$$S_u = \bar{S}V \quad (147)$$

The coefficients  $a_L$  and  $a_R$  only receive a convective flow contribution ( $F$ ) when the flow is into the cell through that face. Now that the interior cell has been considered, the boundary cells on the left and right need to be considered.

## Boundary Cell (Left)

In the same manner as the interior cell, two different cases will be considered for the boundary cell. Firstly, the case where the flow is left to right will be considered, then the case where the flow is right to left.

### (a) Flow From Left to Right ( $F > 0$ )

For the left boundary cell, consider the case where the flow direction is left to right, as shown in Figure 28 (a). The general finite volume discretisation for the left boundary cell is:

$$F_r T_r - F_l T_l = D_r A_r (T_R - T_P) - 2D_l A_l (T_P - T_A) \quad (148)$$

In the same manner as the previous chapters, the diffusive flux through the left face contains a factor of 2 because the distance to the boundary face is half the distance to the boundary

cell centroid. As the flow direction is left to right, the temperature on the cell faces ( $T_l$  and  $T_r$ ) are assigned the following values when an upwind scheme is used.

$$T_l = T_A \quad T_r = T_P \quad (149)$$

Hence, the finite volume discretisation for the left boundary cell becomes:

$$F_r T_P - F_l T_A = D_r A_r (T_R - T_P) - 2D_l A_l (T_P - T_A) \quad (150)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_r A_r + 2D_l A_l + F_r] = T_R [D_r A_r] + T_A [2D_l A_l + F_l] \quad (151)$$

Rearrange the first term on the left-hand side slightly.

$$T_P \underbrace{[0 + D_r A_r + (F_r - F_l) + 2D_l A_l + F_l]}_{a_p} = T_L \underbrace{[0]}_{a_L} + T_R \underbrace{[D_r A_r]}_{a_R} + T_A \underbrace{[2D_l A_l + F_l]}_{S_u} \quad (152)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (153)$$

$$a_L = 0 \quad a_R = D_r A_r \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (154)$$

$$S_u = T_A (2D_l A_l + F_l) \quad S_p = -(2D_l A_l + F_l) \quad (155)$$

## (b) Flow From Right to Left ( $F < 0$ )

Now consider the case for the left boundary cell where the flow direction is right to left, as shown in Figure 28 (b). The general finite volume discretisation for the left boundary cell is:

$$F_r T_r - F_l T_l = D_r A_r (T_R - T_P) - 2D_l A_l (T_P - T_A) \quad (156)$$

As the flow direction is right to left, the temperature on the cell faces ( $T_l$  and  $T_r$ ) are assigned the following values when an upwind scheme is used.

$$T_l = T_P \quad T_r = T_R \quad (157)$$

Hence, the finite volume discretisation for the left boundary cell becomes:

$$F_r T_R - F_l T_P = D_r A_r (T_R - T_P) - 2D_l A_l (T_P - T_A) \quad (158)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_r A_r - F_l + 2D_l A_l] = T_R [D_r A_r - F_r] + T_A [2D_l A_l] \quad (159)$$

Rearrange the first term on the left-hand side slightly.

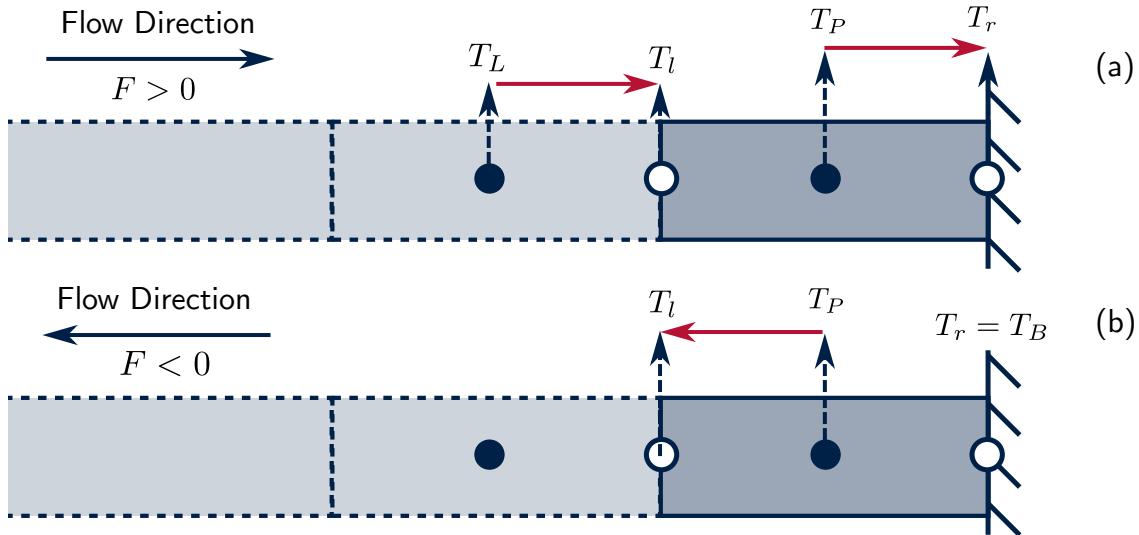
$$T_P \underbrace{[0 + D_r A_r - F_r + (F_r - F_l) + 2D_l A_l]}_{a_p} = T_L \underbrace{[0]}_{a_L} + T_R \underbrace{[D_r A_r - F_r]}_{a_R} + T_A \underbrace{[2D_l A_l]}_{S_u} \quad (160)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (161)$$

$$a_L = 0 \quad a_R = D_r A_r - F_r \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (162)$$

$$S_p = -2D_l A_l \quad S_u = T_A (2D_l A_l) \quad (163)$$



**Figure 29:** A diagram of the temperature on the left and right faces of the boundary cell (right) computed using upwind differencing when the flow direction is (a) left to right and (b) right to left.

### (c) Flow From Either Direction

The finite volume discretisation for the boundary cell (left) can be written concisely (accounting for flow from either direction) as:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (164)$$

With the following coefficients:

$$a_L = 0 \quad (165)$$

$$a_R = D_r A_r + \max(-F_r, 0) \quad (166)$$

$$a_P = a_L + a_R + (F_r - F_l) - S_p \quad (167)$$

$$S_p = -(2D_l A_l + \max(F_l, 0)) \quad (168)$$

$$S_u = T_A (2D_l A_l + \max(F_l, 0)) \quad (169)$$

The source terms ( $S_u$  and  $S_p$ ) only receive a contribution when the convective flux from the left boundary is into the cell ( $F_l$  is positive). Conversely, the convective flux over the right face of the cell ( $a_R$ ) only receives a contribution when the flow is into the cell ( $F_r$  is negative).

### Boundary Cell (Right)

In the same manner as the boundary cell (left), two different cases will be considered for the boundary cell (right). Firstly, the case where the flow is left to right will be considered, then the case where the flow is right to left.

#### (a) Flow From Left to Right ( $F > 0$ )

For the right boundary cell, consider the case where the flow is left to right, as shown in Figure 29 (a). The temperature of the wall is  $T_B$  and the general finite volume discretisation of the right boundary cell is:

$$F_r T_r - F_l T_l = 2D_r A_r (T_B - T_P) - D_r A_r (T_P - T_L) \quad (170)$$

As with the previous treatment of boundary cells, a factor of 2 is introduced into the diffusive flux through the right face because the distance to the boundary face is half of the distance between cell centroids. As the flow direction is left to right, the temperature on the cell faces ( $T_l$  and  $T_r$ ) are assigned the following values when an upwind scheme is used:

$$T_l = T_L \quad T_r = T_P \quad (171)$$

Hence, the finite volume discretisation for the right boundary cell becomes:

$$F_r T_P - F_l T_L = 2D_r A_r (T_B - T_P) - D_l A_l (T_P - T_L) \quad (172)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_l A_l + F_l + 2D_r A_r] = T_L [D_l A_l + F_l] + T_B [2D_r A_r] \quad (173)$$

Rearrange the first term on the left-hand side slightly.

$$T_P \underbrace{[D_l A_l + F_l + 0 + (F_r - F_l) + 2D_r A_r]}_{a_p} = T_L \underbrace{[D_l A_l + F_l]}_{a_L} + T_R \underbrace{[0]}_{a_R} + T_B \underbrace{[2D_r A_r]}_{S_u} \quad (174)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (175)$$

$$a_L = D_l A_l + F_l \quad a_R = 0 \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (176)$$

$$S_p = -2D_r A_r \quad S_u = 2D_r A_r T_B \quad (177)$$

## (b) Flow From Right to Left ( $F < 0$ )

Now consider the case on the right boundary cell where the flow direction is right to left, as shown in Figure 29 (b). The general finite volume discretisation of the right boundary cell is:

$$F_r T_r - F_l T_l = 2D_r A_r (T_B - T_P) - D_l A_l (T_P - T_L) \quad (178)$$

As the flow direction is right to left, the temperature on the cell faces ( $T_l$  and  $T_r$ ) are assigned the following values when an upwind scheme is used:

$$T_l = T_P \quad T_r = T_B \quad (179)$$

Hence, the finite volume discretisation for the right boundary cell becomes:

$$F_r T_B - F_l T_P = 2D_r A_r (T_B - T_P) - D_l A_l (T_P - T_L) \quad (180)$$

Rearrange the equation in terms of the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ).

$$T_P [D_l A_l - F_l + 2D_r A_r] = T_L [D_l A_l] + T_B [2D_r A_r - F_r] \quad (181)$$

Rearrange the first term on the left-hand side slightly.

$$T_P \underbrace{[D_l A_l + 0 + (F_r - F_l) + 2D_r A_r - F_r]}_{a_p} = T_L \underbrace{[D_l A_l]}_{a_L} + T_R \underbrace{[0]}_{a_R} + T_B \underbrace{[2D_r A_r - F_r]}_{S_u} \quad (182)$$

The equation is now written in standard form, with the following coefficients:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (183)$$

$$a_L = D_l A_l \quad a_R = 0 \quad a_p = a_L + a_R + (F_r - F_l) - S_p \quad (184)$$

$$S_p = -(2D_r A_r - F_r) \quad S_u = T_B (2D_r A_r - F_r) \quad (185)$$

### (c) Flow From Either Direction

The finite volume discretisation for the boundary cell (left) can be written concisely (accounting for flow from either direction) as:

$$a_P T_p = a_L T_L + a_R T_R + S_u \quad (186)$$

With the following coefficients:

$$a_L = D_l A_l + \max(F_l, 0) \quad (187)$$

$$a_R = 0 \quad (188)$$

$$a_P = a_L + a_R + (F_r - F_l) - S_p \quad (189)$$

$$S_p = -(2D_r A_r + \max(-F_r, 0)) \quad (190)$$

$$S_u = T_B(2D_r A_r + \max(-F_r, 0)) \quad (191)$$

The source terms ( $S_u$  and  $S_p$ ) only receive a contribution from convection when the convective flux from the right boundary is into the cell ( $F_r$  is negative). Conversely, the convective flux over the left face of the cell ( $a_L$ ) only receives a contribution when the flow is into the cell ( $F_l$  is positive).

### Table of Coefficients

Assemble all of the coefficients into a table for ease of reference:

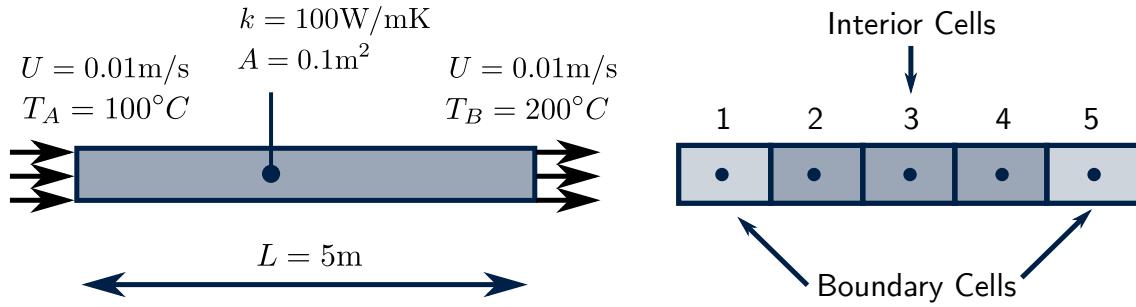
	$a_L$	$a_R$	$S_p$	$S_u$
Left	0	$D_r A_r$ + $\max(-F_r, 0)$	$-(2D_l A_l$ + $\max(F_l, 0))$	$T_A(2D_l A_l$ + $\max(F_l, 0)) + \bar{S}V$
Interior	$D_l A_l$ + $\max(F_l, 0)$	$D_r A_r$ + $\max(-F_r, 0)$	0	$\bar{S}V$
Right	$D_l A_l + \max(0, F_l)$	0	$-(2D_r A_r$ + $\max(-F_r, 0))$	$T_B(2D_r A_r$ + $\max(-F_r, 0)) + \bar{S}V$

Along with the relationship for the central coefficients:

$$a_P = a_L + a_R + (F_r - F_l) - S_p \quad (192)$$

### Example Problem: Conduction and Convection in a 1D Bar

For the worked example in this chapter, the same example from the previous chapter will be used. However, it will be shown that the upwind differencing scheme can achieve stable solutions at higher cell Peclet numbers which were not possible with the central differencing scheme in the previous chapter. As a reminder, Figure 30 shows a diagram of 1D convection and diffusion of heat in the bar (or channel). The bar has a length of 5m, a cross-sectional



**Figure 30:** An example problem to demonstrate 1D convection and diffusion in a bar.

area of  $0.1 \text{ m}^2$ , a thermal conductivity of  $100 \text{ W/mK}$  and a specific heat capacity of  $1000 \text{ J/Kg K}$ . The temperature at the left end of the bar ( $T_A$ ) is  $100^\circ\text{C}$  and the temperature at the right end ( $T_B$ ) is  $200^\circ\text{C}$ . There is a constant source of heat of  $1000 \text{ W/m}^3$  in the bar. Fluid flows through the bar from left to right at  $0.01 \text{ m/s}$ .

### Step 1: Divide the Geometry into a Mesh

Divide the geometry into a mesh of  $N = 5$  cells of equal length. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{5}{5} = 1\text{m} \quad (193)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d = 1\text{m} \quad (194)$$

### Step 2: Assign Material Properties

The thermal conductivity  $k$  and the cross-sectional area  $A$  are the same for every cell in the mesh. Hence, the parameter  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 \text{ [W/K]} \quad (195)$$

$$D_l A_l = D_r A_r = DA = 10 \text{ [W/K]} \quad (196)$$

The convective heat flux through the cell faces is given by:

$$F = \rho c_p U A = 1.0 * 1000 * 0.01 * 0.1 = 1 \text{ [W/K]} \quad (197)$$

$$F_l = F_r = F = 1 \text{ [W/K]} \quad (198)$$

The heat source in each cell is given by:

$$\bar{S}V = \bar{S}A L_{\text{cell}} = 1000 * 0.1 * 1 = 100 \text{ [W]} \quad (199)$$

### Step 3: Calculate Matrix Coefficients

Now calculate the coefficients required to assemble the matrices. The most straightforward way is to fill in the table of coefficients:

	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (Left)	0	10	31	-21	2200
Interior	11	10	21	0	100
Boundary (Right)	11	0	31	-20	4100

Notice that the calculated coefficients are different to the calculated coefficients from the previous chapter, which are shown in the table below.

	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (Left)	0	9.5	30.5	-21	2200
Interior	10.5	9.5	20	0	100
Boundary (Right)	10.5	0	29.5	-19	3900

The difference in the coefficients arises because upwind differencing has been used to discretise the convection term in this chapter, rather than central differencing.

### Step 4: Assemble the Matrices

Assign the coefficients to their correct location in the matrix.

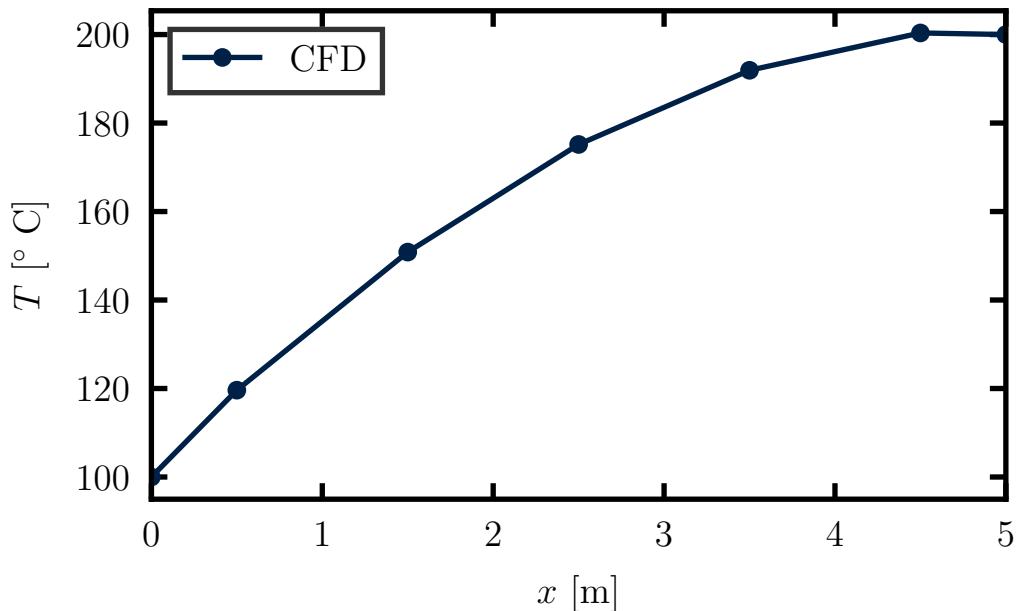
$$\begin{bmatrix} 31 & -10 & 0 & 0 & 0 \\ -11 & 21 & -10 & 0 & 0 \\ 0 & -11 & 21 & -10 & 0 \\ 0 & 0 & -11 & 21 & -10 \\ 0 & 0 & 0 & -11 & 31 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 2200 \\ 100 \\ 100 \\ 100 \\ 4100 \end{bmatrix} \quad (200)$$

### Step 5: Solve the Equations

Now that the matrices have been assembled, the matrix equation can be solved with an appropriate *iterative method*. Popular algorithms include Geometric Algebraic Multi-grid (GAMG) and Preconditioned Conjugate Gradient (PCG). However, as with the previous chapter, these algorithms will not be considered in detail here.

### Run the Example Problem Yourself!

Now, open either the Excel spreadsheet, the Python source code or the MATLAB source code and solve the problem yourself.



**Figure 31:** Temperature variation along the bar with a flow velocity is 0.01 m/s ( $Pe = 0.1$ ). Upwind differencing has been used for the convection term.

Excel	<i>solve1DConvectionDiffusionEquationUpwind.xlsx</i>
Python	<i>solve1DConvectionDiffusionEquationUpwind.py</i>
MATLAB	<i>solve1DConvectionDiffusionEquationUpwind.m</i>

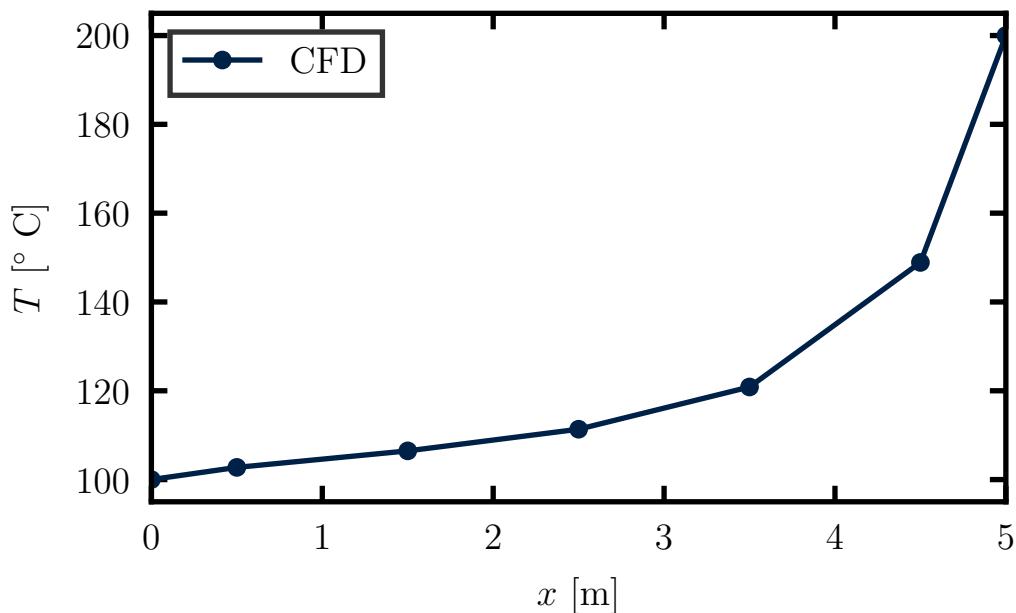
Examine the calculation of the coefficients, the assembly of the matrices and run the code. You can even try changing some of the geometric and material properties of the problem (such as the thermal conductivity or the length of the bar) and examine the changes in the solution.

## Results

The temperature variation in the bar when upwind differencing is used is shown in Figure 31. The temperature profile is almost identical to the temperature profile from the previous chapter (where central differencing was used). This is because the strength of convection is relatively small in comparison with the strength of diffusion in the bar. However, comparing the solution vectors, it can be seen that the upwind differencing scheme computes a slightly different solution to the central differencing scheme.

$$T = \underbrace{\begin{bmatrix} 119.6 \\ 150.8 \\ 175.2 \\ 191.9 \\ 200.4 \end{bmatrix}}_{\text{Upwind}} \quad T = \underbrace{\begin{bmatrix} 119.2 \\ 151.1 \\ 175.9 \\ 192.7 \\ 200.8 \end{bmatrix}}_{\text{Central}} \quad (201)$$

The upwind scheme is less accurate than the central differencing scheme because the upwind scheme assumes a constant variation of temperature between the cell face and the cell centroid, rather than a linear variation (see Figure 26 for example). For this reason, the upwind differencing scheme is often referred to as *first-order accurate* in the literature.



**Figure 32:** Temperature variation along the bar with a flow velocity of 0.3 m/s ( $Pe = 3$ ). Upwind differencing has been used for the convection term.

### Increasing the Flow Velocity

Following the previous chapter, the flow velocity will now be increased to 0.3 m/s. This results in a Peclet number of 3. As shown in Figure 32, the upwind scheme is able to compute a solution that does not contain any non-physical oscillations. This is a significant improvement on the central differencing scheme from the previous chapter.

### Choice of Convection Scheme for RANS Computations

While the upwind scheme does not lead to non-physical oscillations at high  $Pe$ , upwind schemes reduce the accuracy of the solution as they assume a constant variation between the cell face and the cell centroid. To achieve a balance between accuracy and stability, a variety of discretisation schemes are available in modern CFD codes, such as *linear-upwind differencing*. These schemes are preferable for the convection term and should be selected (for RANS simulations) where possible, to achieve a balance between accuracy and stability.

# 1 Dirichlet and Neumann Boundary Conditions

In this chapter, the finite volume discretisation of the 1D diffusion equation from the previous course will be revisited. The diffusion equation will be used to introduce *Neumann* (fixed gradient) boundary conditions for the first time and revisit *Dirichlet* (fixed value) boundary conditions. While a general diffusion equation can be used to describe many transported quantities, the diffusion equation for thermal energy (temperature) will be considered in this course, as it is the easiest to physically interpret. Mathematically, the general scalar transport equation for thermal energy (temperature) for incompressible (low speed) flows is:

$$\underbrace{\frac{\partial(\rho c_p T)}{\partial t}}_{\text{Unsteady}} + \underbrace{\nabla \cdot (\rho c_p T \mathbf{U})}_{\text{Convection}} = \underbrace{\nabla \cdot (k \nabla T)}_{\text{Diffusion}} + S \quad (1)$$

where  $T$  is the temperature,  $\rho$  is the density,  $c_p$  is the specific heat capacity at constant pressure,  $k$  is the thermal conductivity,  $\mathbf{U}$  is the velocity vector and  $S$  is a source of thermal energy. Some readers may recognise this equation as the transport equation for *enthalpy*. To observe the transport equation for enthalpy ( $h$ ), define  $h = c_p T$  and substitute in to equation 1:

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho \mathbf{U} h) = \nabla \cdot (k \nabla T) + S \quad (2)$$

However, this form of the thermal energy equation will not be used in this course, as temperature is an easier variable to interpret and follow than enthalpy.

The focus of this chapter will be the diffusion and source terms in equation 1, since the convection term was considered in the previous course and the unsteady term will be considered in a later course. Neglecting the convection and unsteady terms in equation 1, the diffusion equation for thermal energy is:

$$\cancel{\frac{\partial(\rho c_p T)}{\partial t}}^0 + \cancel{\nabla \cdot (\rho c_p T \mathbf{U})}^0 = \nabla \cdot (k \nabla T) + S \quad (3)$$

$$0 = \nabla \cdot (k \nabla T) + S \quad (4)$$

Expanding the gradient ( $\nabla$ ) and divergence ( $\nabla \cdot$ ) operators into Cartesian coordinates ( $x, y, z$ ):

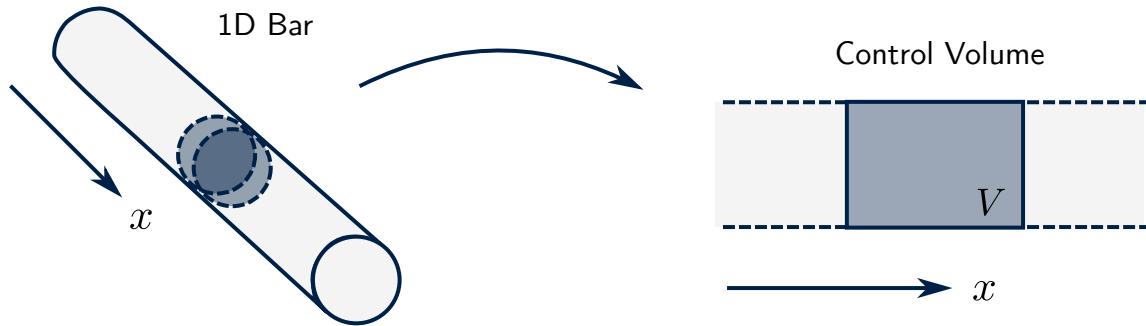
$$0 = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) + S \quad (5)$$

To clearly demonstrate how Dirichlet and Neumann boundary conditions are implemented by CFD solvers, the diffusion equation will only be considered in 1D (the  $x$  direction) in this chapter. 2D geometries will be considered later in Chapter 2 of this course. In 1D, the diffusion equation for thermal energy (temperature) is:

$$0 = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \cancel{\frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right)}^0 + \cancel{\frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right)}^0 + S \quad (6)$$

$$0 = \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S \quad (7)$$

The majority of modern CFD codes use the **finite volume method** to solve the transport equations for the various flow variables (velocity, temperature, pressure etc.). In the previous



**Figure 1:** A 1D finite volume of fluid with volume  $V$ , which has been isolated from the bar.

course, the finite volume method was reviewed in detail. To avoid repetition, only a concise overview of the finite volume method will be provided here, to ensure that the derivation and solution presented is continuous and coherent. Further detail can be found in the previous course. The first stage in the finite volume method is to integrate the transport equation over a finite-sized control volume (cell). As shown in Figure 1, a 1D cell can be thought of as a slice through a thin 1D bar or rod. Integrating the 1D diffusion equation (equation 7) over this cell:

$$0 = \int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) + S \right] dV \quad (8)$$

Integration and addition are commutative operations (it doesn't matter what order they are carried out in). Hence, the finite volume integral can be split into two separate integrals and each one can then be considered in turn.

$$0 = \int_V \left[ \frac{d}{dx} \left( k \frac{dT}{dx} \right) \right] dV + \int_V [S] dV \quad (9)$$

As shown in the previous course, Gauss's divergence theorem is used to replace the volume integral of the diffusion term with a surface integral. Recall (from the previous course) that the divergence theorem for a general vector field  $\mathbf{B}$  is written as:

$$\int_V (\nabla \cdot \mathbf{B}) dV = \int_A (\mathbf{B} \cdot \hat{\mathbf{n}}) dA \quad (10)$$

where  $\hat{\mathbf{n}}$  is the unit normal vector pointing out of the control volume and  $A$  is the surface area of the control volume. In 1D Cartesian coordinates, the divergence theorem can be written:

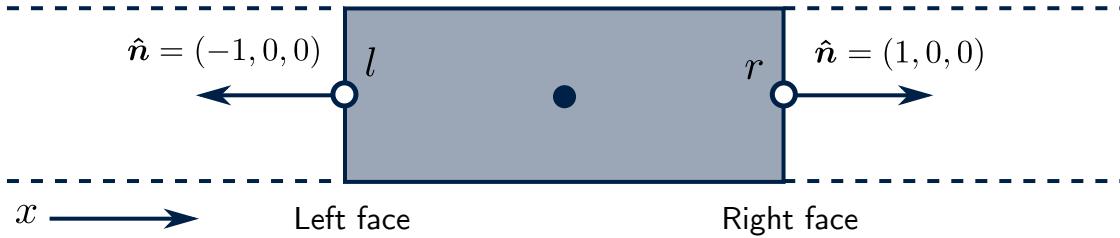
$$\int_V \left( \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \right) dV = \int_A (B_x n_x + B_y n_y + B_z n_z) dA \quad (11)$$

$$\int_V \left( \frac{\partial B_x}{\partial x} + \cancel{\frac{\partial B_y}{\partial y}}^0 + \cancel{\frac{\partial B_z}{\partial z}}^0 \right) dV = \int_A \left( B_x n_x + \cancel{B_y n_y}^0 + \cancel{B_z n_z}^0 \right) dA \quad (12)$$

$$\int_V \left( \frac{\partial B_x}{\partial x} \right) dV = \int_A (B_x n_x) dA \quad (13)$$

For the heat diffusion equation  $\mathbf{B} = k \nabla T$ . Hence  $B_x = k \partial T / \partial x$  in 1D. Applying the 1D divergence theorem to the 1D heat diffusion equation leads to:

$$0 = \int_A \left[ k \frac{dT}{dx} n_x \right] dA + \int_V [S] dV \quad (14)$$



**Figure 2:** A diagram to show the face normal vectors on the left and right faces of the 1D cell. The cell normal vectors always point out of the cell.

The source term is averaged over the control volume, so it can be moved outside the volume integral.

$$0 = \int_A \left[ k \frac{dT}{dx} n_x \right] dA + \bar{S} \int_V dV \quad (15)$$

$$0 = \int_A \left[ k \frac{dT}{dx} n_x \right] dA + \bar{S} V \quad (16)$$

Recall that the unit normal vector ( $n_x$ ) always points out of the cell. As shown in Figure 2, on the left face of the cell ( $l$ ), the unit normal vector is negative. Conversely, the unit normal vector is positive on the right face of the cell ( $r$ ).

$$0 = \left[ kA \frac{dT}{dx} \right]_r - \left[ kA \frac{dT}{dx} \right]_l + \bar{S} V \quad (17)$$

This finite volume discretisation is valid for all cells in the mesh. However, different simplifications are required for interior and boundary cells before the equations can be solved.

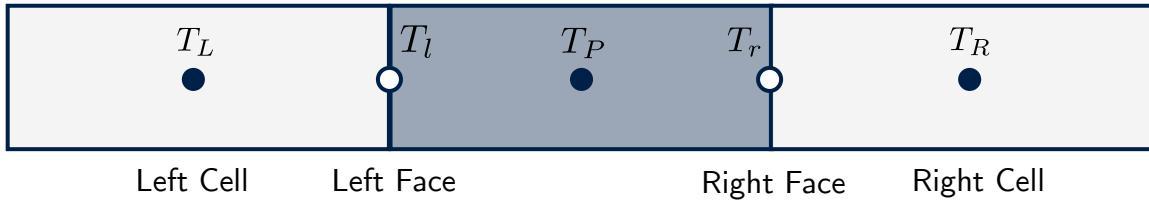
## Interior Cells

Start with the general finite volume discretisation of the 1D diffusion equation.

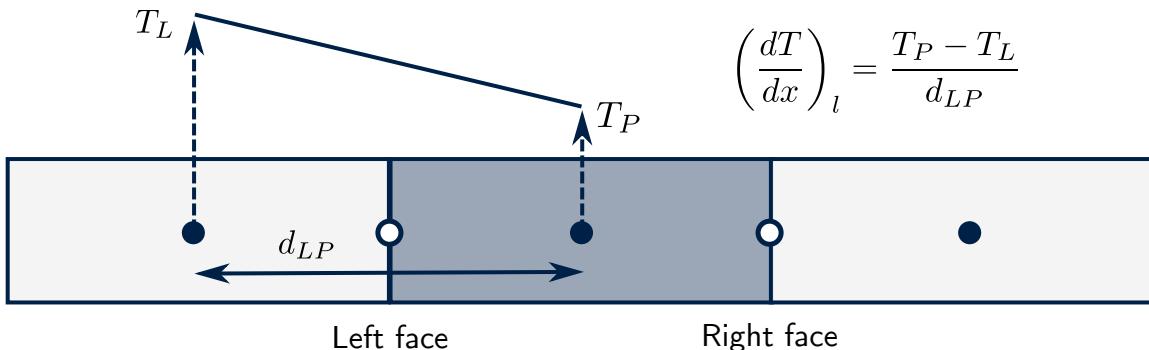
$$0 = \left[ kA \frac{dT}{dx} \right]_r - \left[ kA \frac{dT}{dx} \right]_l + \bar{S} V \quad (18)$$

To simplify and solve this equation for the interior cells, the temperature gradient on the cell faces ( $l$  and  $r$ ) needs to be expressed in terms of temperatures at the cell centroids ( $L$ ,  $R$  and  $P$ ). This is because the unknowns in the matrix equations are the temperatures at the cell centroids and other variables need to be expressed in terms of these temperatures. Hence, the temperature (and other variables) are calculated and stored at the cell centroids, rather than the nodes or faces. This approach is called a **cell-centred** finite volume method and is used by the CFD codes OpenFOAM, ANSYS Fluent and Star CCM+ but not ANSYS CFX (which uses a **node based** finite volume method).

Throughout this course, the lower case subscripts ( $l$  and  $r$ ) will be used to refer to cell faces, while upper-case subscripts ( $L$ ,  $R$  and  $P$ ) will be used to refer to cell centroids, as shown in Figure 3. This notation has been chosen because the temperatures at the cell centroids are the unknowns in the system. The necessary simplification of the gradients on the cell faces can be accomplished with linear interpolation, which is often called **central-differencing**. To



**Figure 3:** A diagram to show the difference between the temperatures on the cell faces ( $T_l$  and  $T_r$ ) and the temperature at the cell centroids ( $T_L$  and  $T_R$ ).



**Figure 4:** Central differencing (linear interpolation) of the temperature gradient on the left face of the cell using the values at the cell centroids of the interior cell ( $T_P$ ) and the left cell ( $T_L$ ).

help understand this simplification, remember that the spatial gradient of temperature can be thought of as:

$$\frac{dT}{dx} \sim \frac{\Delta T}{\Delta x} = \frac{\text{Change in Temperature}}{\text{Distance}} \quad (19)$$

As shown in Figure 4, the temperature gradient on the left face can be expressed using central differencing as:

$$\left( \frac{dT}{dx} \right)_l = \frac{T_P - T_L}{d_{LP}} \quad (20)$$

where  $d_{LP}$  is the distance between the cell centroids  $L$  and  $P$ . In a similar manner, the temperature gradient on the right face can also be expressed using central differencing:

$$\left( \frac{dT}{dx} \right)_r = \frac{T_R - T_P}{d_{PR}} \quad (21)$$

Substitute this simplification into the 1D diffusion equation (equation 18).

$$\left( k_r A_r \frac{T_R - T_P}{d_{PR}} \right) - \left( k_l A_l \frac{T_P - T_L}{d_{LP}} \right) + \bar{S}V = 0 \quad (22)$$

The 1D diffusion equation can now be solved for the temperatures at the cell centroids ( $T_L$ ,  $T_R$  and  $T_P$ ). To simplify this process, rearrange the equation and collect the terms in terms of temperature of the interior cell ( $T_P$ ), temperature of the left cell ( $T_L$ ) and the temperature of the right cell ( $T_R$ ).

$$T_P \left( \frac{k_l A_l}{d_{LP}} + \frac{k_r A_r}{d_{PR}} \right) = T_L \left( \frac{k_l A_l}{d_{LP}} \right) + T_R \left( \frac{k_r A_r}{d_{PR}} \right) + \bar{S}V \quad (23)$$

At this stage, it is useful to introduce some new notation to simplify the finite volume discretisation. This new notation makes it easier to compare the finite volume discretisation of interior cells, boundary cells and add additional terms to the equation later on. The approach adopted in the previous course was to introduce the notation  $D = k/d$ . This quantity can be thought of as the diffusive flux of heat per unit area through the cell face and has units of  $\text{W/m}^2\text{K}$ . Using this notation, the finite volume discretiation becomes:

$$T_P(D_l A_l + D_r A_r) = T_L(D_l A_l) + T_R(D_r A_r) + \bar{S}V \quad (24)$$

For consistency with other equations that will be introduced later, write the above equation in the following form:

$$\begin{aligned} a_p T_P &= a_L T_L + a_R T_R + S_u \\ T_P \underbrace{(D_l A_l + D_r A_r + 0)}_{a_p} &= T_L \underbrace{(D_l A_l)}_{a_L} + T_R \underbrace{(D_r A_r)}_{a_R} + \underbrace{\bar{S}V}_{S_u} \end{aligned} \quad (25)$$

This equation is valid for all cells in the mesh, except for the boundary cells. The boundary cells have to be considered separately.

Equation 23 can also be written using summation notation:

$$\begin{aligned} a_p T_P &= \sum_N a_N T_N + S_u \\ a_p &= \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| & a_N &= \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| & S_u &= \bar{S}V \end{aligned} \quad (26)$$

where the summation is taken over the  $N$  cell neighbours. For 1D cells,  $N = 2$ . However, for 2D and 3D meshes, the summation notation becomes more useful as each cell has multiple cell neighbours and the finite volume discretisation can become quite long (this will be shown in Chapter 2)! Also notice that in the summation notation, the magnitude of the unit normal vector  $|\hat{\mathbf{n}}|$  and the vector  $|\mathbf{d}|$  are used. This is permissible because the unit normal vector and the vector  $\mathbf{d}$  are parallel with eachother. In turn this is because the mesh used in this course is *structured* and *orthogonal*. Special treatment is required for *non-orthogonal* and *unstructured* meshes. However, the special treatment required for these meshes will not be considered in this course (for brevity).

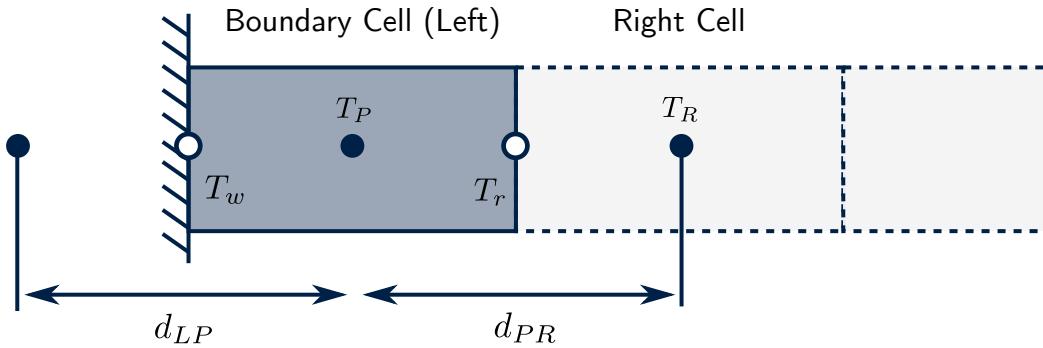
## Boundary Cell (Left) - Dirichlet Boundary Condition

Now the boundary conditions on the left hand boundary cell will be considered, starting with the Dirichlet (fixed temperature) condition. Figure 5 shows a schematic diagram of this cell, with a fixed temperature  $T_w$  applied at the wall. The general finite volume discretisation for the 1D diffusion equation is:

$$\left( kA \frac{dT}{dx} \right)_r - \left( kA \frac{dT}{dx} \right)_l + \bar{S}V = 0 \quad (27)$$

The right face of the boundary cell is connected to an interior cell. Hence, the same central differencing scheme for the temperature gradient from the previous section can be used for the right face. However, the left face is connected to a boundary. As shown in Figure 5, the temperature gradient term for the left face is:

$$\left( \frac{dT}{dx} \right)_l = \frac{T_P - T_w}{d_{LP}/2} \quad (28)$$



**Figure 5:** The left boundary cell with temperature  $T_P$  at its centroid. The shared face between the boundary cell and the right cell is at a temperature  $T_r$  and the wall has a temperature  $T_l = T_w$ .

The factor of  $1/2$  is required as the distance from the cell centroid to the face is  $1/2$  of  $d_{LP}$  (the distance from the cell centroid to the cell centroid of the adjacent cell). The finite volume discretisation of the 1D heat-diffusion equation for the left boundary cell is now:

$$\left( k_r A_r \frac{T_r - T_P}{d_{PR}} \right) - \left( k_l A_l \frac{T_P - T_w}{d_{LP}/2} \right) + \bar{S}V = 0 \quad (29)$$

Again, introduce the notation  $D = k/d$  for the diffusive heat flux per unit area.

$$T_P (2D_l A_l + D_r A_r) = T_R (D_r A_r) + T_w (2D_l A_l) + \bar{S}V \quad (30)$$

For consistency with the interior cell, write in the standard form:

$$a_p T_p = a_L T_L + a_R T_R + S_u \quad (31)$$

$$T_P \underbrace{(0 + D_r A_r + 2D_l A_l)}_{a_P} = T_L \underbrace{(0)}_{a_L} + T_R \underbrace{(D_r A_r)}_{a_R} + \underbrace{T_w (2D_l A_l)}_{S_u} + \bar{S}V \quad (32)$$

For comparison with the interior cell, the boundary cell (left) has the following coefficients:

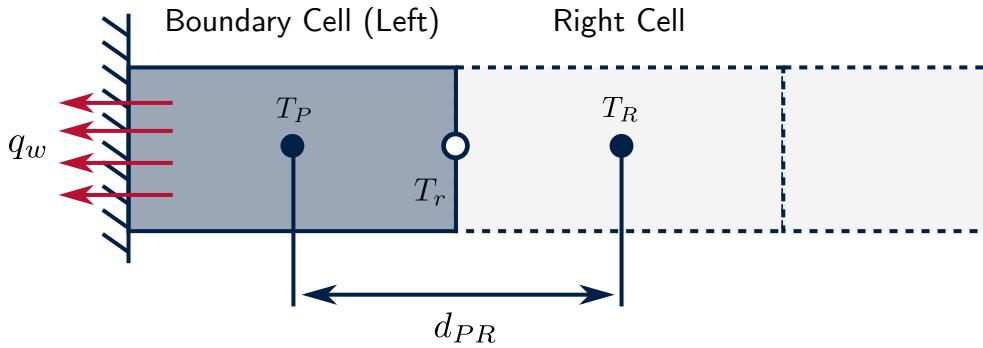
$$a_L = 0 \quad a_R = D_r A_r \quad a_p = a_L + a_R - S_p \quad (33)$$

$$S_p = -2D_l A_l \quad S_u = \bar{S}V + T_w (2D_l A_l) \quad (34)$$

It follows that the boundary temperature  $T_w$  enters the equation through the source terms  $S_u$  and  $S_p$ . The left coefficient  $a_L$  is also set to zero, as this face is not connected to an interior cell. Using summation notation, the finite volume discretisation for the boundary cell can be written:

$$\begin{aligned} a_p T_p &= \sum_N a_N T_N + S_u \\ a_p &= \left( \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \right) + \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| \quad a_N = \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \\ S_u &= \bar{S}V + T_w \left( \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| \right) \end{aligned} \quad (35)$$

When using summation notation, it should be noted that  $a_N = 0$  for the left boundary face and the summation for  $a_P$  does not include the left boundary face. The wall temperature enters the equation through the source terms  $S_u$  and  $S_p$  as before.



**Figure 6:** The left boundary cell with a temperature \$T\_P\$ at its centroid. The shared face between the boundary cell and the right cell is at a temperature \$T\_r\$ and a fixed heat flux of \$q\_w\$ is applied at the left boundary face. Positive \$q\_w\$ indicates heat passing out of the domain.

## Boundary Cell (Left) - Neumann Boundary Condition

Figure 6 shows a schematic diagram of the same boundary cell, but with a fixed heat flux (per unit area) \$q\_w\$ applied at the left wall, instead of a fixed temperature \$T\_w\$. Mathematically, the heat flux from the wall (per unit area) \$q\_w\$ is given by Fourier's law:

$$q_w = -k \nabla T \cdot \hat{n} \quad [\text{W/m}^2] \quad (36)$$

$$q_w = -k \left( \frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}, \frac{\partial T}{\partial z} \right) \cdot (n_x, n_y, n_z) \quad (37)$$

$$q_w = -k \left( \frac{\partial T}{\partial x} n_x + \frac{\partial T}{\partial y} n_y + \frac{\partial T}{\partial z} n_z \right) \quad (38)$$

where \$\hat{n}\$ is the unit normal vector **out of the cell** into the wall. The negative sign is required to ensure that the heat flows in the opposite direction to the temperature gradient (high temperature to low temperature). In 1D, Fourier's Law for the wall heat flux reduces to:

$$q_w = -k \left( \frac{\partial T}{\partial x} n_x + \frac{\partial T}{\partial y}^0 n_y + \frac{\partial T}{\partial z}^0 n_z \right) \quad (39)$$

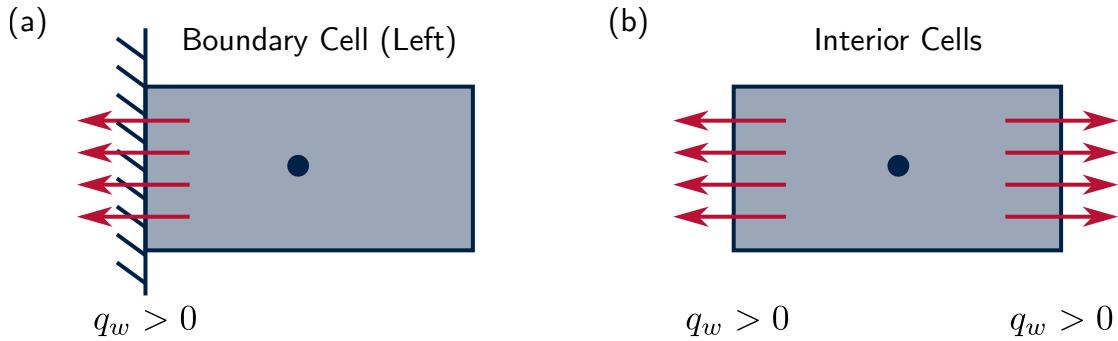
$$q_w = -k \frac{dT}{dx} n_x \quad [\text{W/m}^2] \quad (40)$$

Hence, the fixed heat flux boundary condition is a type of **Neumann** boundary condition, as the heat flux (\$q\_w\$) is being used to set the temperature gradient at the wall. To understand how heat flux boundary conditions are applied, the finite volume discretisation for a general 1D cell needs to be revisited.

$$\underbrace{\left( kA \frac{dT}{dx} n_x \right)_r}_{\text{Heat Flux Right Face}} + \underbrace{\left( kA \frac{dT}{dx} n_x \right)_l}_{\text{Heat Flux Left Face}} + \overline{S}V = 0 \quad (41)$$

The underlined terms in the above equation actually represent the heat flux **out** of the left and right faces of the cell. To understand why, consider Fourier's Law in 1D across an interior face in the mesh:

$$q = -k \frac{dT}{dx} n_x \quad [\text{W/m}^2] \quad (42)$$



**Figure 7:** Diagram to show the sign convention for positive heat flux through the (a) boundary and (b) interior faces of the mesh.

Substitute this form of Fourier's Law into the finite volume discretisation for the interior cells (equation 41):

$$\underbrace{(-q_r A_r)}_{\text{out}} + \underbrace{(-q_l A_l)}_{\text{out}} + \overline{S}V = 0 \quad (43)$$

As the unit normal vector always points out of the cells, each underlined term in the above equation represents a heat flux **out** of the cell, across its faces (except for the source term  $\overline{S}V$ ). Once again, the negative sign ensures that the heat flows in the opposite direction to the temperature gradient (high temperature to low temperature).

When considering the left boundary cell in Figure 6, a fixed heat flux  $q_w$  is applied at the left face.

$$q_w = -k_l \frac{dT}{dx} n_x \quad [\text{W/m}^2] \quad (44)$$

It should be noted that the heat flux out of the wall into the fluid is equal and opposite the heat flux out of the fluid into the wall. For consistency, the convention of positive heat flux passing **out of the fluid cell** into the wall will be taken as positive ( $q_w > 0$ ), while heat flux passing out of the wall into the fluid will be taken as negative ( $q_w < 0$ ). Figure 7 shows a schematic diagram to highlight the heat flux sign convention adopted for boundary and interior cells. Returning to Figure 6, the heat flux across the right face (interior face) of the left boundary cell can be calculated using central differencing from the previous section. Hence, the finite volume discretisation for this boundary cell becomes:

$$k_r A_r \left( \frac{T_R - T_P}{d_{LP}} \right) - q_w A_l + \overline{S}V = 0 \quad (45)$$

As the boundary heat flux per unit area ( $q_w$ ) has units of  $\text{W/m}^2$ , it is multiplied by the face area to obtain the correct units of  $\text{W}$ . To simplify this equation further, the diffusive heat flux per unit area ( $D = k/d$ ) will be introduced again.

$$D_r A_r (T_R - T_p) - q_w A_l + \overline{S}V = 0 \quad (46)$$

$$T_P (D_r A_r) = T_R (D_r A_r) - q_w A_l + \overline{S}V \quad (47)$$

For consistency with the interior cell, write in the standard form:

$$a_p T_p = a_L T_L + a_R T_R + S_u \quad (48)$$

$$T_P \underbrace{(0 + D_r A_r + 0)}_{a_p} = T_L \underbrace{(0)}_{a_L} + T_R \underbrace{(D_r A_r)}_{a_R} + \underbrace{-q_w A_l + \overline{S}V}_{S_u} \quad (49)$$

Hence, the boundary cell (left) has the following coefficients when a Neumann condition is applied:

$$a_L = 0 \quad a_R = D_r A_r \quad a_p = a_L + a_R - S_p \quad (50)$$

$$S_P = 0 \quad S_u = \bar{S}V - q_w A_l \quad (51)$$

Using summation notation, the finite volume discretisation for boundary cells with Neumann boundary conditions can be written:

$$a_p T_P = \sum_N a_N T_N + S_u$$

$$a_p = \left( \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \right) \quad a_N = \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \quad S_u = \bar{S}V - q_w A_l$$

As before, the coefficient  $a_N = 0$  for boundary faces. Notice that unlike the Dirichlet boundary condition, the Neumann boundary condition ( $q_w$ ) only makes a contribution to  $S_u$  and not  $a_p$ .

## Summary of Coefficients

A summary of the finite volume coefficients is provided in the table below for interior and boundary cells. Notice that the only difference between the Dirichlet and Neumann boundary conditions is the coefficients  $S_u$  and  $S_p$ .

Dirichlet Boundary Condition					
Cell Type	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (L)	0	$D_R A_R$	$a_l + a_r - S_p$	$-2D_L A_L$	$T_w(2D_L A_L) + \bar{S}V$
Interior	$D_L A_L$	$D_R A_R$	$a_l + a_r - S_p$	0	$\bar{S}V$
Boundary (R)	$D_L A_L$	0	$a_l + a_r - S_p$	$-2D_R A_R$	$T_w(2D_R A_R) + \bar{S}V$

Neumann Boundary Condition					
Cell Type	$a_L$	$a_R$	$a_p$	$S_p$	$S_u$
Boundary (L)	0	$D_R A_R$	$a_l + a_r - S_p$	0	$-q_w A_l + \bar{S}V$
Interior	$D_L A_L$	$D_R A_R$	$a_l + a_r - S_p$	0	$\bar{S}V$
Boundary (R)	$D_L A_l$	0	$a_l + a_r - S_p$	0	$-q_w A_r + \bar{S}V$

The summation notation can also be summarised in a table. However, rather than arranging by cell type, this table is arranged by the face type on each cell. The contribution of each face to the cell total is then given (noting that we sum over the  $N$  cell faces):

## Summation Notation

Face Type	$a_P$	$a_N$	$S_u$
Interior	$\frac{k_N A_N}{ \mathbf{d}_{PN} }  \hat{\mathbf{n}} $	$\frac{k_N A_N}{ \mathbf{d}_{PN} }  \hat{\mathbf{n}} $	0
Dirichlet	$\frac{k_w A_w}{ \mathbf{d}_{LP} /2}  \hat{\mathbf{n}} $	0	$T_w \left( \frac{k_w A_w}{ \mathbf{d}_{LP} /2}  \hat{\mathbf{n}}  \right)$
Neumann	0	0	$-q_w A_w$

If the above (summation notation) table is used to assemble the equations, it should be remembered that the additional volumetric source contribution ( $\bar{S}V$ ), which is not included in the table, is still required.

## Assemble and Solve the Equations

As shown in the previous course, once the coefficients ( $a_p, a_l, a_r, S_u$ ) have been calculated, the finite volume equations can be assembled in matrix form. Start by writing an equation for each cell in the mesh. If the mesh has 5 cells, then the equations are:

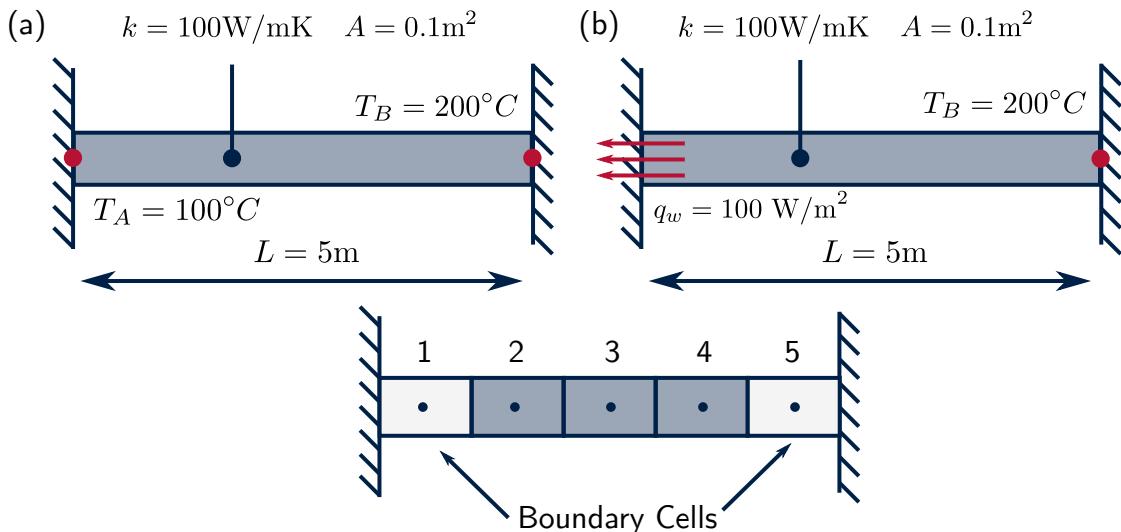
Cell 1	Boundary Cell (Left)	$a_{p1}T_1 = a_{r1}T_2 + S_{u1}$
Cell 2	Interior Cell	$a_{p2}T_2 = a_{l2}T_1 + a_{r2}T_3 + S_{u2}$
Cell 3	Interior Cell	$a_{p3}T_3 = a_{l3}T_2 + a_{r3}T_4 + S_{u3}$
Cell 4	Interior Cell	$a_{p4}T_4 = a_{l4}T_3 + a_{r4}T_5 + S_{u4}$
Cell 5	Boundary Cell (Right)	$a_{p5}T_5 = a_{l5}T_4 + S_{u5}$

where the coefficients  $a_p, a_L, a_R$  and  $S_u$  are given in the summary in the previous section. Rearrange the equations, so that all the temperatures ( $T_P, T_L$  and  $T_R$ ) are on the left hand side and the source terms ( $S_u$ ) are on the right hand side.

Cell 1	Boundary Cell (Left)	$a_{p1}T_1 - a_{r1}T_2 = S_{u1}$
Cell 2	Interior Cell	$-a_{l2}T_1 + a_{p2}T_2 - a_{r2}T_3 = S_{u2}$
Cell 3	Interior Cell	$-a_{l3}T_2 + a_{p3}T_3 - a_{r3}T_4 = S_{u3}$
Cell 4	Interior Cell	$-a_{l4}T_3 + a_{p4}T_4 - a_{r4}T_5 = S_{u4}$
Cell 5	Boundary Cell (Right)	$-a_{l5}T_4 + a_{p5}T_5 = S_{u5}$

These equations can now be written concisely in matrix form ( $\mathbf{AT} = \mathbf{B}$ ):

$$\begin{bmatrix} a_{p1} & -a_{r1} & 0 & 0 & 0 \\ -a_{l2} & a_{p2} & -a_{r2} & 0 & 0 \\ 0 & -a_{l3} & a_{p3} & -a_{r3} & 0 \\ 0 & 0 & -a_{l4} & a_{p4} & -a_{r4} \\ 0 & 0 & 0 & -a_{l5} & a_{p5} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} S_{u1} \\ S_{u2} \\ S_{u3} \\ S_{u4} \\ S_{u5} \end{bmatrix} \quad (52)$$



**Figure 8:** An example problem to demonstrate 1D heat diffusion in a bar. Two cases will be considered: (a) a fixed temperature ( $T_A$ ) of  $100^\circ\text{C}$  at the left end of the bar and (b) a fixed heat flux ( $q_w$ ) of  $100 \text{ W/m}^2$  at the left end of the bar.

Notice that the matrices have a diagonal banded structure. This is because the mesh is structured (the cells are ordered in a regular pattern) and the coefficients represent the connectivity to the neighbouring cells. The banded structure of the matrices will change slightly in later chapters, when 2D meshes are considered.

Commercial CFD solvers populate the matrices by calculating the coefficients ( $a_l$ ,  $a_p$  and  $a_r$ ) automatically for the user and then solve the matrix equations. In the next section, the entire process will be demonstrated with an example problem. A mesh will be defined, the coefficients will be calculated and then the matrices will be constructed and solved.

## Example Problem - Heat Diffusion in a Bar

To demonstrate the difference between the Neumann and Dirichlet boundary conditions an example problem will now be considered. Consider 1D steady-state diffusion of heat in a bar, as shown in Figure 8 (b). For ease of comparison, this example problem has the same geometry, mesh and material properties as the previous course. However, at the left hand end, a heat flux boundary condition will be applied (Figure 8 (b)), rather than a fixed temperature boundary condition (Figure 8 (a)), as was applied in the previous course. The bar has a length of 5m, a cross-sectional area of  $0.1 \text{ m}^2$  and a thermal conductivity of  $100 \text{ W/mK}$ . The temperature at the right end of the bar ( $T_B$ ) is  $200^\circ\text{C}$ . There is a constant heat source ( $\bar{S}$ ) of  $1000 \text{ W/m}^3$  in the bar. At the left end of the bar, a fixed heat flux ( $q_w$ ) of  $100 \text{ W/m}^2$  will be applied, rather than a fixed temperature ( $T_A$ ) of  $100^\circ\text{C}$ , as was applied in the previous course.

### Step 1: Divide the Geometry into a Mesh

For the example in Figure 8, divide the geometry into a mesh of 5 cells of equal length. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{5}{5} = 1\text{m} \quad (53)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d = 1\text{m} \quad (54)$$

## Step 2: Assign Material Properties

The thermal conductivity  $k$  and the cross-sectional area  $A$  are the same for every cell in the mesh. Hence, the parameter  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 \text{ [W/K]} \quad (55)$$

The volumetric heat source in each cell is given by:

$$\bar{SV} = \bar{S}AL_{\text{cell}} = 1000 * 0.1 * 1 = 100 \text{ [W]} \quad (56)$$

## Step 3: Calculate Matrix Coefficients

Now calculate the coefficients required to assemble the matrices. The most straightforward way is to fill in the table of coefficients:

Neumann Boundary Conditions (at the left end of the bar)					
	$a_L$	$a_R$	$S_p$	$S_u$	$a_p$
Boundary (Left)	0	10	0	90	10
Interior	10	10	0	100	20
Boundary (Right)	10	0	-20	4100	30

For comparison, the matrix coefficients for the same problem with a Dirichlet boundary condition (fixed temperature) at the left end are shown below. The differences between the two sets of matrix coefficients are highlighted in red.

Dirichlet Boundary Conditions (at the left end of the bar)					
	$a_L$	$a_R$	$S_p$	$S_u$	$a_p$
Boundary (Left)	0	10	-20	2100	30
Interior	10	10	0	100	20
Boundary (Right)	10	0	-20	4100	30

Alternatively, the summation notation summary table can be filled in (noting that an additional source  $\bar{SV} = 100\text{W}$  is required in each cell):

## Dirichlet Boundary Conditions (at the left end of the bar)

Face Type	$a_P$	$a_N$	$S_u$
Interior	10	10	0
Dirichlet (Left)	20	0	2000
Dirichlet (Right)	20	0	4000
Neumann (Left)	0	0	-10

**Step 4: Assemble the Matrices**

Assign the coefficients to their correct locations in the matrix. Note that the final matrices will be identical, regardless of the method used to calculate the coefficients (summation notation,  $D = k/d$ , or otherwise).

For the Neumann (fixed heat flux) boundary condition at the left end:

$$\begin{bmatrix} \textcolor{red}{10} & -10 & 0 & 0 & 0 \\ -10 & 20 & -10 & 0 & 0 \\ 0 & -10 & 20 & -10 & 0 \\ 0 & 0 & -10 & 20 & -10 \\ 0 & 0 & 0 & -10 & 30 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} \textcolor{red}{90} \\ 100 \\ 100 \\ 100 \\ 4100 \end{bmatrix} \quad (57)$$

and for the Dirichlet boundary condition (fixed temperature) at the left end:

$$\begin{bmatrix} \textcolor{red}{30} & -10 & 0 & 0 & 0 \\ -10 & 20 & -10 & 0 & 0 \\ 0 & -10 & 20 & -10 & 0 \\ 0 & 0 & -10 & 20 & -10 \\ 0 & 0 & 0 & -10 & 30 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} \textcolor{red}{2100} \\ 100 \\ 100 \\ 100 \\ 4100 \end{bmatrix} \quad (58)$$

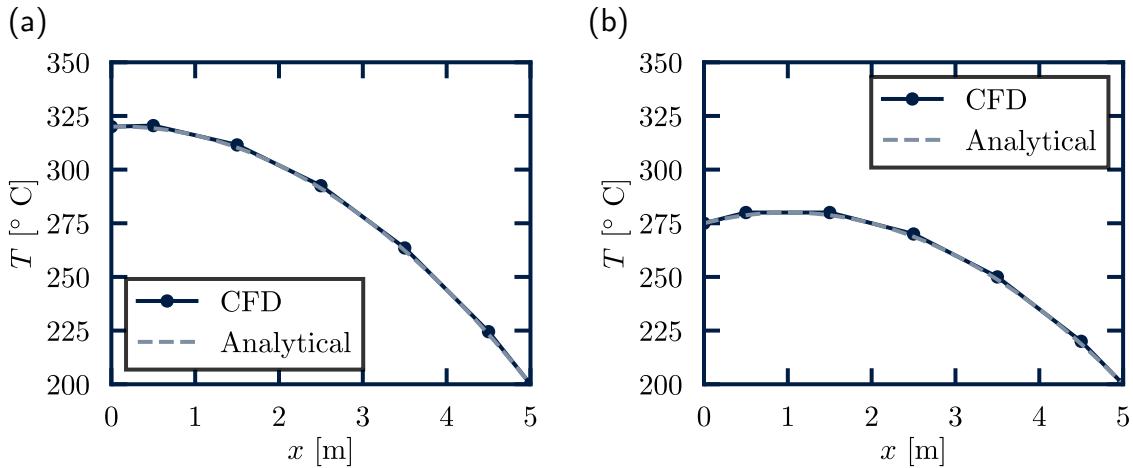
Once again, the differences between the Dirichlet and Neumann boundary conditions are highlighted in red.

**Step 5: Solve the Equations**

Now that the matrices have been assembled, the matrix equation can be solved with an appropriate *iterative method*. As with the previous course, different algorithms to solve the matrix equation  $\mathbf{AT} = \mathbf{B}$  will not be considered, as details can be found in any comprehensive text on linear algebra. Instead, the default algorithms used by Excel and Python will be used, as this is not the focus of this course.

**Run the Example Problem Yourself!**

Now, open either the Excel spreadsheets or the Python source code and solve the problem with the Neumann boundary condition (fixed heat flux) yourself. For the Dirichlet boundary condition, you can use the source code from the previous course.



**Figure 9:** Temperature variation along the 1D bar with a heat flux of (a) 100 W/m<sup>2</sup> and (b) 1000 W/m<sup>2</sup> applied at the left end.

Excel neumannBoundaryConditions.xlsx

Python neumannBoundaryConditions.py

Examine the calculation of the coefficients, the assembly of the matrices and run the code. You can even try changing the strength of the heat flux at the left hand end of the bar ( $q_w$ ), or even set an adiabatic condition (zero heat flux,  $q_w = 0$ ).

## Results

Figure 9 shows the temperature distribution in the bar with a heat flux of 100 W/m<sup>2</sup> applied at the left hand end. For comparison, the analytical solution is also shown:

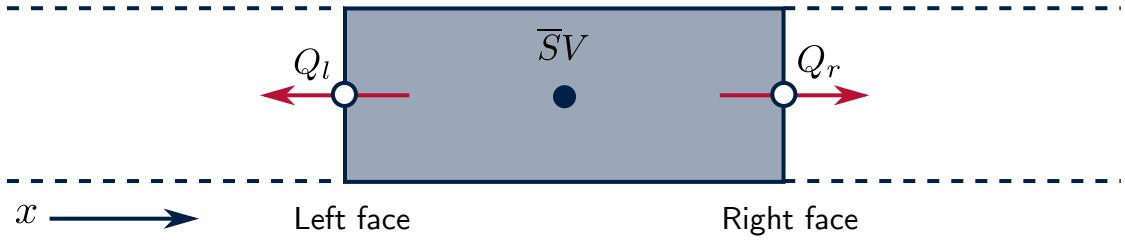
$$T = \frac{\bar{S}}{2k} (L^2 - x^2) + \frac{q_w}{k}(x - L) + T_B \quad (59)$$

By changing the heat flux at the left boundary from 100 W/m<sup>2</sup> to 1000 W/m<sup>2</sup>, Figure 9(b) shows that the temperature of the left end of the bar is not fixed. The heat flux  $q_w$  implicitly sets the temperature gradient at the left end of the bar (as the thermal conductivity  $k$  is fixed) and the temperature profile develops to match this gradient.

$$q_w = -k \frac{dT}{dx} n_x \quad (60)$$

As a result, when a heat flux boundary condition is applied in a CFD code, the boundary temperature on that surface develops as part of the solution. To calculate the boundary temperature from the CFD solution ( $T_w$ ), the temperature must be extrapolated from the temperature at the cell centroid  $T_1$ . Central differencing can be used to extrapolate the boundary temperature from the temperature at the cell centroids.

$$q_w A_L = -k_L A_L \left( \frac{T_1 - T_A}{d_{LP}/2} \right) n_x \quad D_L A_l = \frac{k_L A_L}{d_{LP}} \quad (61)$$



**Figure 10:** A diagram to show the heat flux out of the cell across the left ( $Q_l$ ) and right ( $Q_r$ ) faces of the cell.

$$T_A = T_1 + \frac{q_A A_L}{2D_L A_L n_x} \quad (62)$$

The use of central differencing to extrapolate and calculate the boundary temperature can be found in both the Excel spreadsheet and Python source code. Note that for the left boundary face, the unit normal vector ( $n_x$ ) points in the negative  $x$  direction, so  $n_x = -1$ .

## Heat Balancing

Once a CFD solution has been computed, a heat balance can be written for each of the cells in the mesh as an additional check of the solution. To understand the heat balance, start with the general finite volume discretisation for the interior cells:

$$k_l A_l \left( \frac{T_P - T_L}{d_{LP}} \right) n_x + k_r A_r \left( \frac{T_R - T_P}{d_{PR}} \right) n_x + \bar{S}V = 0 \quad (63)$$

The heat flux out of the left face of the cell ( $Q_l$ ) is:

$$Q_l = -k_l A_l \left( \frac{T_P - T_L}{d_{LP}} \right) n_x \quad [\text{W}] \quad (64)$$

and the heat flux out of the right face of the cell ( $Q_r$ ) is:

$$Q_r = -k_r A_r \left( \frac{T_R - T_P}{d_{PR}} \right) n_x \quad [\text{W}] \quad (65)$$

The negative sign is required to ensure that the heat flows in the opposite direction to the temperature gradient and the unit normal vector ensures that heat fluxes out of the cell are positive. Substitute these definitions into the general finite volume discretisation for the interior cell:

$$-Q_l - Q_r + \bar{S}V = 0 \quad (66)$$

For the left boundary cell, the distance from the boundary cell centroid to the wall is  $d_{LP}/2$ , so the heat flux balance is:

$$-2Q_l - Q_r + \bar{S}V = 0 \quad (67)$$

Likewise, for the right boundary face:

$$-Q_l - 2Q_r + \bar{S}V = 0 \quad (68)$$

Physically, these equations state that the sum of the heat fluxes into the cell (positive heat sources  $\bar{S}$  generate heat in the cell) is equal to zero. The equation will now be evaluated

explicitly, using the temperatures that were computed at the cell centroids. Remember that this is a post-processing operation and the temperatures are now known. As the matrix solvers are (generally) iterative, there will be some error in solving the finite volume equations. This error in the heat balance in each cell is given by:

$$\text{Error} = -Q_l - Q_r + \bar{S}V \quad (69)$$

The table below summarises the heat balance for each cell in the mesh:

Cell	$Q_l$ [W]	$Q_r$ [W]	$\bar{S}V$ [W]	Error
1	10	90	100	0
2	-90	190	100	0
3	-190	290	100	0
4	-290	390	100	0
5	-390	490	100	0

Total heat flux out of the bar =  $10 + 490 = 500\text{W}$   
 Total heat generated in the bar =  $100 + 100 + 100 + 100 + 100 = 500\text{W}$

The heat balance is useful as we can easily observe that the total heat flux out of the left and right faces of the bar (highlighted in red in the table), balances the total heat generated in the bar (500W). As both heat fluxes are positive, this indicates that heat is flowing out of the bar at both ends, with the majority of the heat flowing out of the right face as it is at a lower temperature (see Figure 9). As a further check, we can also see that the heat flux boundary condition has been applied correctly to the left boundary cell (cell 1):

$$Q_A = q_w A = 100 * 0.1 = 10 \text{ W} \quad (70)$$

It may be noted that for this problem, the error in the heat balance in every cell is zero. This is because a direct matrix solver has been used to solve the equations. However, real CFD codes use iterative matrix solvers and a small error will generally exist in every cell in the mesh. This error is often called the **residual** of the equation. As a vector, the root-mean-square and maximum values of the vector are often computed and reported to the user (instead of the entire vector) as a measure of the convergence of the solution.

$$\text{RMS} = \sqrt{\text{Error 1}^2 + \text{Error 2}^2 + \text{Error 3}^2 + \text{Error 4}^2 + \text{Error 5}^2} \quad (71)$$

$$\text{MAX} = \max(\text{Error 1}, \text{Error 2}, \text{Error 3}, \text{Error 4}, \text{Error 5}) \quad (72)$$

These are the values that are typically shown to the user as convergence graphs in the graphical user interface (GUI) of the CFD code.

## 2 Transport Equations in 2D

The heat diffusion equation will now be solved in 2D ( $x$  and  $y$ ) rather than just 1D ( $x$ ) using the finite volume method. This same approach can be used to integrate and solve any transport equation (momentum, turbulence, species concentration etc.) under consideration. Starting with the heat diffusion equation in vector notation:

$$0 = \nabla \cdot (k \nabla T) + S \quad (73)$$

In 2D Cartesian coordinates, the derivatives can be expanded to give:

$$0 = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + S \quad (74)$$

Following the same approach as the previous chapter, integrate the equation over a finite control volume (cell) with volume  $V$ .

$$0 = \int_V \left[ \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + S \right] dV \quad (75)$$

Once again, split the integral into diffusion and source components. This is permissible as integration and addition are commutative operations (they can be performed in any order without changing the result).

$$0 = \int_V \left[ \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) \right] dV + \int_V [S] dV \quad (76)$$

As before, the source term is averaged over the control volume, so that it can be moved outside the integral.

$$0 = \int_V \left[ \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) \right] dV + \bar{S} \int_V dV \quad (77)$$

$$0 = \int_V \left[ \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) \right] dV + \bar{S}V \quad (78)$$

The volume integral of the diffusion term can be simplified using Gauss's divergence theorem. Recall (from the previous course) that the divergence theorem for a general vector field  $\mathbf{B}$  is written as:

$$\int_V (\nabla \cdot \mathbf{B}) dV = \int_A (\mathbf{B} \cdot \hat{\mathbf{n}}) dA \quad (79)$$

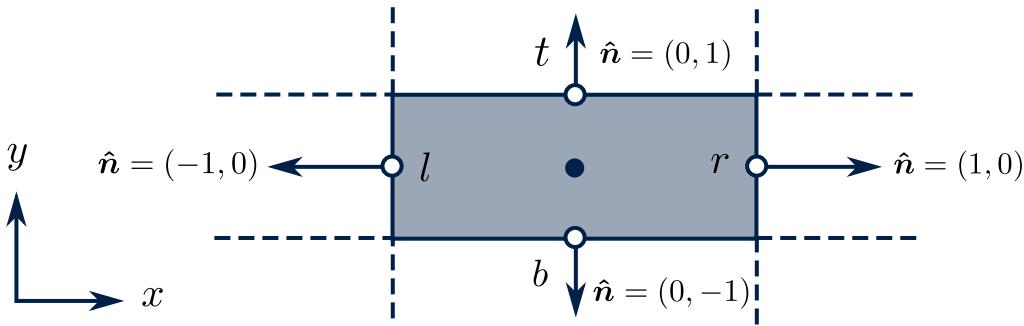
$$\int_V \left( \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \right) dV = \int_A (B_x n_x + B_y n_y + B_z n_z) dA \quad (80)$$

where  $\hat{\mathbf{n}}$  is the unit normal vector pointing out of the control volume and  $A$  is the surface area of the control volume. In 2D, the divergence theorem can be written:

$$\int_V \left( \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} \right) dV = \int_A (B_x n_x + B_y n_y) dA \quad (81)$$

For the 2D heat diffusion equation  $\mathbf{B} = k \nabla T$ . Hence  $B_x = k \partial T / \partial x$  and  $B_y = k \partial T / \partial y$ . Applying the 2D divergence theorem to the 2D heat diffusion equation leads to:

$$0 = \int_A \left[ k \frac{\partial T}{\partial x} n_x + k \frac{\partial T}{\partial y} n_y \right] dA + \bar{S}V \quad (82)$$



**Figure 11:** A diagram to show the face normal vectors on the left ( $l$ ), right ( $r$ ), top ( $t$ ) and bottom ( $b$ ) faces of the 2D cell. The cell normal vectors always point out of the cell.

The cell has a finite number of faces ( $N$ ). For a quadrilateral cell  $N = 4$ , while for a triangular cell  $N = 3$ . Hence, the surface integral over the entire surface can be replaced with an integral over each of the  $N$  faces of the cell.

$$0 = \sum_N \int_A \left[ k \frac{\partial T}{\partial x} n_x + k \frac{\partial T}{\partial y} n_y \right] dA + \bar{S}V \quad (83)$$

In the second-order finite volume method, the flow quantities all vary linearly across the face. The integral across the face can therefore be reduced to the value at the centre of the face (a constant value).

$$0 = \sum_N \left[ k \frac{\partial T}{\partial x} n_x + k \frac{\partial T}{\partial y} n_y \right] \int_A dA + \bar{S}V \quad (84)$$

$$0 = \sum_N \left[ k \frac{\partial T}{\partial x} n_x + k \frac{\partial T}{\partial y} n_y \right] A + \bar{S}V \quad (85)$$

It may be noted that this is equivalent to:

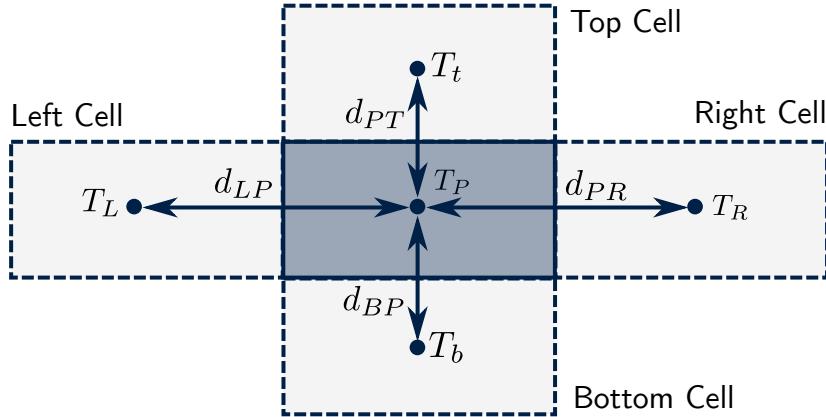
$$0 = \sum_N [kA (\nabla T \cdot \hat{n})] + \bar{S}V \quad (86)$$

To simplify the equation further, consider the 2D quadrilateral cell in Figure 11. The cell has a left face ( $l$ ), a right face ( $r$ ), a top face ( $t$ ) and a bottom face ( $b$ ). The unit normal vectors always point out of the cell. As the  $x$  direction is positive left to right and the  $y$  direction is positive bottom to top, the unit normal vectors on each face are:

Face	$n_x$	$n_y$
Left ( $l$ )	-1	0
Right ( $r$ )	1	0
Bottom ( $b$ )	0	-1
Top ( $t$ )	0	1

Hence, the finite volume discretisation can be simplified further:

$$0 = \left( kA \frac{\partial T}{\partial x} \right)_r - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial y} \right)_t - \left( kA \frac{\partial T}{\partial y} \right)_b + \bar{S}V \quad (87)$$



**Figure 12:** A diagram to show the distance between the cell centroid and the left cell ( $d_{LP}$ ), right cell ( $d_{PR}$ ), bottom cell ( $d_{BP}$ ) and top cell ( $d_{PT}$ ).

This discretisation is valid for all cells in the mesh that are the same shape and orientation as the quadrilateral cell in Figure 11 (a regular structured mesh). To simplify further, the interior and boundary cells need to be considered separately.

## Interior Cells

Starting with the general finite volume discretisation for the 2D quadrilateral cell:

$$0 = \left( kA \frac{\partial T}{\partial x} \right)_r - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial y} \right)_t - \left( kA \frac{\partial T}{\partial y} \right)_b + \bar{S}V \quad (88)$$

Central differencing will be used for all of the diffusion terms. Using the notation in Figure 12 for the distances between the cell centroids ( $d$ ):

$$0 = k_r A_r \frac{T_R - T_P}{d_{PR}} - k_l A_l \frac{T_P - T_L}{d_{LP}} + k_t A_t \frac{T_T - T_P}{d_{PT}} - k_b A_b \frac{T_P - T_B}{d_{BP}} + \bar{S}V \quad (89)$$

As before, the lowercase subscript notation ( $l, r, b, t$ ) is used to refer to the faces of the cell and the uppercase subscript notation ( $L, R, B, T, P$ ) is used to refer to the cell centroids. For simplicity, introduce the diffusive flux per unit area  $D = k/d$ , which has units of  $\text{W}/\text{m}^2\text{K}$ .

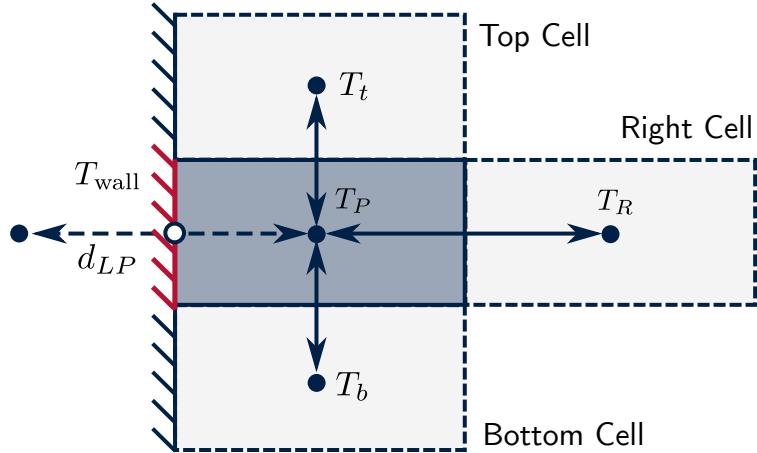
$$0 = D_r A_r (T_R - T_P) - D_l A_l (T_P - T_L) + D_t A_t (T_T - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (90)$$

Rearrange and group the terms by cell centroid temperatures ( $T_P, T_R, T_L, T_T, T_B$ ).

$$\begin{aligned} T_P [D_l A_l + D_r A_r + D_b A_b + D_t A_t] &= T_L [D_l A_l] + T_R [D_r A_r] \\ &\quad + T_B [D_b A_b] + T_T [D_t A_t] + \bar{S}V \end{aligned} \quad (91)$$

Manipulate the equation slightly (add a zero to the  $T_P$  bracket) to express in standard form.

$$\begin{aligned} T_P \underbrace{[D_l A_l + D_r A_r + D_b A_b + D_t A_t + 0]}_{a_p} &= T_L \underbrace{[D_l A_l]}_{a_L} + T_R \underbrace{[D_r A_r]}_{a_R} \\ &\quad + T_B \underbrace{[D_b A_b]}_{a_B} + T_T \underbrace{[D_t A_t]}_{a_B} + \underbrace{\bar{S}V}_{S_u} \end{aligned} \quad (92)$$



**Figure 13:** The boundary cell at the left of the domain with temperature  $T_P$  at its centroid. The wall (the left face of the cell) is at a fixed temperature  $T_{\text{wall}}$  and is shown in red.

The standard form is similar to the standard form used for the 1D diffusion equation.

$$\begin{aligned} a_p T_P &= a_L T_L + a_R T_R + a_B T_B + a_T T_T + S_u & [2D] \\ a_p T_P &= a_L T_L + a_R T_R + S_u & [1D] \end{aligned} \quad (93)$$

With the following coefficients for interior cells:

$$\begin{aligned} a_p &= a_L + a_R + a_T + a_B - S_p \\ a_L &= D_l A_l & a_R = D_r A_r & a_B = D_b A_b & a_T = D_t A_t \\ S_p &= 0 & S_u &= \bar{S}V \end{aligned} \quad (94)$$

Using summation notation (from the previous chapter), the finite volume discretisation is identical to the 1D discretisation from the previous chapter.

$$\begin{aligned} a_p T_P &= \sum_N a_N T_N + S_u \\ a_p &= \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| & a_N &= \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| & S_u &= \bar{S}V \end{aligned} \quad (95)$$

The summation notation becomes more useful as the number of faces ( $N$ ) increases. Hence, it is often used by real CFD codes (and given in user manuals) as cells in 2D and 3D meshes may have many faces and it is easier to keep track of all the terms in the finite volume discretisation.

## Boundary Cells - Dirichlet Boundary Conditions

Figure 13 shows a boundary cell on the left of the domain, which will be used to simplify the finite volume discretisation for boundary cells. Starting with the general finite volume discretisation for the 2D heat diffusion equation:

$$0 = \left( kA \frac{\partial T}{\partial x} \right)_r - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial y} \right)_t - \left( kA \frac{\partial T}{\partial y} \right)_b + \bar{S}V \quad (96)$$

The right, top and bottom faces are connected to interior cells. Hence, the heat diffusion through these faces can be simplified using the same treatment as the interior cells.

$$0 = D_r A_r (T_R - T_P) - \left( kA \frac{\partial T}{\partial x} \right)_l + D_t A_t (T_T - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (97)$$

In this chapter, fixed value (Dirichlet) boundary conditions will be applied on the left face. However, the method used in the previous chapter for Neumann boundary conditions could be applied instead if desired. The left boundary face is at a temperature  $T_{\text{wall}}$  and the distance to the wall is half the distance to the next cell centroid. Hence the heat diffusion through the left face is:

$$\left( kA \frac{\partial T}{\partial x} \right)_l = k_l A_l \frac{T_P - T_{\text{wall}}}{d_{LP}/2} = 2D_l A_l (T_p - T_{\text{wall}}) \quad (98)$$

Substitute into the finite volume discretisation:

$$0 = D_r A_r (T_R - T_P) - 2D_l A_l (T_p - T_{\text{wall}}) + D_t A_t (T_T - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (99)$$

Rearrange and group the terms by cell centroid temperatures ( $T_P, T_R, T_L, T_T, T_B$ ).

$$\begin{aligned} T_P [0 + D_r A_r + D_b A_b + D_t A_t + 2D_l A_l] &= T_L [0] + T_R [D_r A_r] \\ &\quad + T_B [D_b A_b] + T_T [D_t A_t] \\ &\quad + T_{\text{wall}} [2D_l A_l] + \bar{S}V \end{aligned} \quad (100)$$

The equation is now in standard form:

$$T_P \underbrace{[0 + D_r A_r + D_b A_b + D_t A_t + 2D_l A_l]}_{a_p} = T_L \underbrace{[0]}_{a_L} + T_R \underbrace{[D_r A_r]}_{a_R} \quad (101)$$

$$+ T_B \underbrace{[D_b A_b]}_{a_B} + T_T \underbrace{[D_t A_t]}_{a_T} \quad (102)$$

$$+ \underbrace{T_{\text{wall}} [2D_l A_l] + \bar{S}V}_{S_u} \quad (103)$$

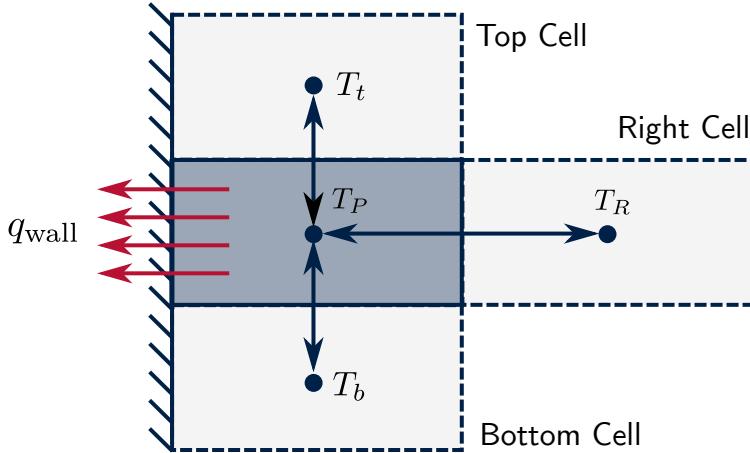
With the following coefficients:

$$\begin{aligned} a_p &= a_L + a_R + a_T + a_B - S_p \\ a_L &= 0 & a_R &= D_r A_r & a_B &= D_b A_b & a_T &= D_t A_t \\ S_p &= -2D_l A_l & S_u &= T_{\text{wall}} (2D_l A_l) + \bar{S}V \end{aligned} \quad (104)$$

Using summation notation (from the previous chapter), the finite volume discretisation is identical to the 1D discretisation from the previous chapter.

$$\begin{aligned} a_p T_P &= \sum_N a_N T_N + S_u \\ a_p &= \left( \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \right) + \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| & a_N &= \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \\ S_u &= \bar{S}V + T_{\text{wall}} \left( \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| \right) \end{aligned} \quad (105)$$

Once again it should be remembered that when using the summation notation  $a_N = 0$  for the boundary faces and the summation for  $a_p$  does not include the boundary faces.



**Figure 14:** The boundary cell at the left of the domain with temperature  $T_P$  at its centroid. A fixed heat flux  $q_{\text{wall}}$  is applied at the wall (the left face of the cell).

## Boundary Cells - Neumann Boundary Conditions

Now consider the case where a fixed heat flux (Neumann boundary condition) is applied to the left boundary face, as shown in Figure 14. Starting with the general finite volume discretisation for the 2D heat diffusion equation:

$$0 = \left( kA \frac{\partial T}{\partial x} \right)_r - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial y} \right)_t - \left( kA \frac{\partial T}{\partial y} \right)_b + \bar{S}V \quad (106)$$

The right, top and bottom faces are connected to interior cells. Hence, the heat flux through these faces can be simplified using the same treatment as the interior cells.

$$0 = D_r A_r (T_R - T_P) - \left( kA \frac{\partial T}{\partial x} \right)_l + D_t A_t (T_T - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (107)$$

With a Neumann boundary condition, the heat flux per unit area **out** of the left face of the cell is:

$$q_{\text{wall}} = -k_l \frac{\partial T}{\partial x} n_x \quad (108)$$

Note that the unit normal vector for this cell  $n_x = -1$ . Hence:

$$q_{\text{wall}} = k_l \frac{\partial T}{\partial x} \quad (109)$$

Hence, the finite volume discretisation now becomes:

$$0 = D_r A_r (T_R - T_P) - q_{\text{wall}} A_l + D_t A_t (T_T - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (110)$$

Rearrange and group the terms by cell centroid temperatures ( $T_P, T_R, T_L, T_T, T_B$ ).

$$\begin{aligned} T_P [0 + D_r A_r + D_b A_b + D_t A_t + 0] &= T_L [0] + T_R [D_r A_r] \\ &\quad + T_B [D_b A_b] + T_T [D_t A_t] \\ &\quad - q_{\text{wall}} A_l + \bar{S}V \end{aligned} \quad (111)$$

The equation is now in standard form:

$$\begin{aligned}
 T_P \underbrace{[0 + D_r A_r + D_b A_b + D_t A_t + 0]}_{a_p} = & T_L \underbrace{[0]}_{a_L} + T_R \underbrace{[D_r A_r]}_{a_R} \\
 & + T_B \underbrace{[D_b A_b]}_{a_B} + T_T \underbrace{[D_t A_t]}_{a_T} \\
 & + \underbrace{-q_{\text{wall}} A_l + \bar{S}V}_{S_u}
 \end{aligned} \tag{112}$$

With the following coefficients:

$$\begin{aligned}
 a_p &= a_L + a_R + a_T + a_B - S_p \\
 a_L = 0 \quad a_R &= D_r A_r \quad a_B = D_b A_b \quad a_T = D_t A_t \\
 S_p = 0 \quad S_u &= -q_{\text{wall}} A_l + \bar{S}V
 \end{aligned} \tag{113}$$

Using summation notation (from the previous chapter), the finite volume discretisation is identical to the 1D discretisation from the previous chapter.

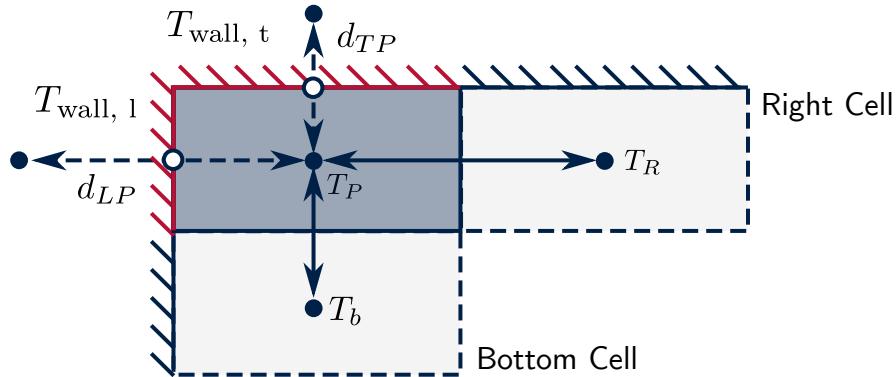
$$\begin{aligned}
 a_p T_P &= \sum_N a_N T_N + S_u \\
 a_p &= \left( \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \right) \quad a_N = \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \quad S_u = \bar{S}V - q_w A_w
 \end{aligned}$$

Once again it should be remembered that when using the summation notation  $a_N = 0$  for the boundary faces and the summation for  $a_p$  does not include the boundary faces.

## Summary of Coefficients

The tables below summarise the coefficients for each of the boundary and interior cells in the mesh when Dirichlet and Neumann boundary conditions are applied.

Dirichlet Boundary Condition (2D)						
	$a_L$	$a_R$	$a_B$	$a_T$	$S_p$	$S_u$
Interior	$D_L A_L$	$D_R A_R$	$D_B A_B$	$D_T A_T$	0	$\bar{S}V$
Boundary (L)	0	$D_R A_R$	$D_B A_B$	$D_T A_T$	$-2D_L A_L$	$T_{\text{wall}}(2D_L A_L) + \bar{S}V$
Boundary (R)	$D_L A_L$	0	$D_B A_B$	$D_T A_T$	$-2D_R A_R$	$T_{\text{wall}}(2D_R A_R) + \bar{S}V$
Boundary (B)	$D_L A_L$	$D_R A_R$	0	$D_T A_T$	$-2D_B A_B$	$T_{\text{wall}}(2D_B A_B) + \bar{S}V$
Boundary (T)	$D_L A_L$	$D_R A_R$	$D_T A_T$	0	$-2D_T A_T$	$T_{\text{wall}}(2D_T A_T) + \bar{S}V$



**Figure 15:** The boundary cell at the top-left of the domain with temperature  $T_P$  at its centroid. The top and left walls are at fixed temperatures of  $T_{\text{wall}, l}$  and  $T_{\text{wall}, t}$

### Neumann Boundary Condition (2D)

	$a_L$	$a_R$	$a_B$	$a_T$	$S_p$	$S_u$
Interior	$D_L A_L$	$D_R A_R$	$D_B A_B$	$D_T A_T$	0	$\bar{S}V$
Boundary (L)	0	$D_R A_R$	$D_B A_B$	$D_T A_T$	0	$-q_{\text{wall}} A_l + \bar{S}V$
Boundary (R)	$D_L A_L$	0	$D_B A_B$	$D_T A_T$	0	$-q_{\text{wall}} A_r + \bar{S}V$
Boundary (B)	$D_L A_L$	$D_R A_R$	0	$D_T A_T$	0	$-q_{\text{wall}} A_B + \bar{S}V$
Boundary (T)	$D_L A_L$	$D_R A_R$	$D_T A_T$	0	0	$-q_{\text{wall}} A_T + \bar{S}V$

The summation notation can also be summarised in a table. Following the previous chapter, this table gives the contribution of each face to the cell total:

### Summation Notation

Face Type	$a_P$	$a_N$	$S_u$
Interior	$\frac{k_N A_N}{ d_{PN} }  \hat{n} $	$\frac{k_N A_N}{ d_{PN} }  \hat{n} $	0
Dirichlet	$\frac{k_w A_w}{ d_{LP} /2}  \hat{n} $	0	$T_w \left( \frac{k_w A_w}{ d_{LP} /2}  \hat{n}  \right)$
Neumann	0	0	$-q_{\text{wall}} A_w$

It should be emphasised once again that with the summation notation, the contribution of the volumetric source term ( $\bar{S}V$ ) also needs to be included in the total for  $S_u$ .

## Cells with Multiple Boundary Faces

For 2D and 3D meshes, some boundary cells may have more than 1 boundary face. Furthermore, the boundary conditions on these boundary faces may be different. Figure 15 shows an example cell with 2 boundary faces. For this cell, the finite volume discretisation is:

$$0 = \left( kA \frac{\partial T}{\partial x} \right)_r - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial y} \right)_t - \left( kA \frac{\partial T}{\partial y} \right)_b + \bar{S}V \quad (114)$$

The right and bottom faces are connected to interior cells. Hence, central differencing can be used to simplify the heat flux through these faces.

$$0 = k_r A_r \frac{T_R - T_P}{d_{PR}} - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial x} \right)_t - k_b A_b \frac{T_P - T_B}{d_{BP}} + \bar{S}V \quad (115)$$

Introduce the diffusive flux per unit  $D = k/d$ , as normal.

$$0 = D_r A_r (T_R - T_P) - \left( kA \frac{\partial T}{\partial x} \right)_l + \left( kA \frac{\partial T}{\partial x} \right)_t - D_b A_b (T_P - T_B) + \bar{S}V \quad (116)$$

If the boundary faces are at fixed temperatures of  $T_{\text{wall}, l}$  on the left and  $T_{\text{wall}, t}$  on the top, then central differencing can also be used on the boundary faces.

$$0 = D_r A_r (T_R - T_P) - 2D_l A_l (T_P - T_{\text{wall}, l}) + 2D_t A_t (T_{\text{wall}, t} - T_P) - D_b A_b (T_P - T_B) + \bar{S}V \quad (117)$$

Rearrange and collect the temperatures at the cell centroids ( $T_P, T_R, T_T$ ).

$$\begin{aligned} T_P [0 + D_r A_r + D_b A_b + 0 + 2D_l A_l + 2D_t A_t] &= T_L [0] + T_r [D_r A_r] \\ &\quad + T_B [D_b A_b] + T_P [0] + \bar{S}V \\ &\quad + T_{\text{wall}, l} (2D_l A_l) + T_{\text{wall}, t} (2D_t A_t) \end{aligned} \quad (118)$$

The equation is in standard form:

$$a_p = a_L + a_R + a_T + a_B - S_p \quad (119)$$

With the following coefficients:

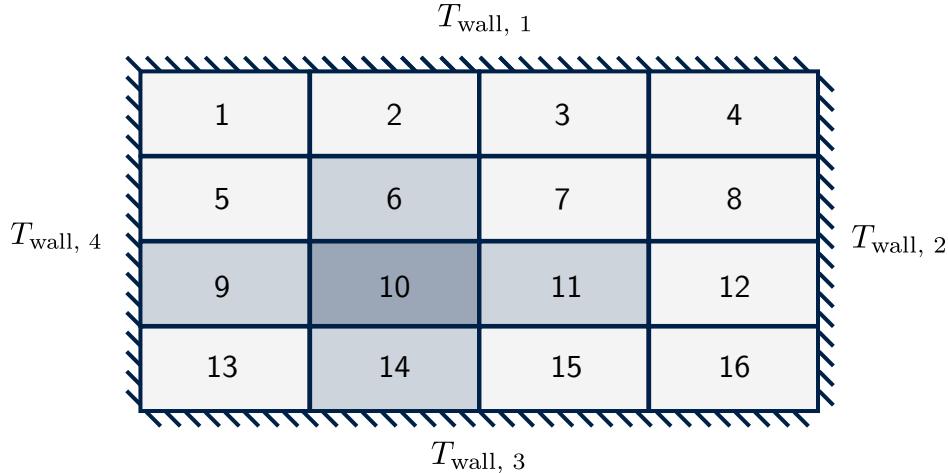
$$\begin{aligned} a_L &= 0 & a_R &= D_r A_r & a_B &= D_b A_b & a_T &= 0 \\ S_p &= -2D_l A_l - 2D_t A_t & S_u &= \bar{S}V + T_{\text{wall}, l} (2D_l A_l) + T_{\text{wall}, t} (2D_t A_t) \end{aligned} \quad (120)$$

Hence, when a cell has multiple boundary faces the contribution of those faces to  $a_p = 0$ . These faces then add their contribution to  $S_p$  and  $S_u$  instead.

When using summation notation, additional summation symbols are used in the definition of  $a_p$  and  $S_u$  to emphasise that additional contributions are added for each boundary face  $M$  that the cell has. Otherwise the equations are identical.

$$\begin{aligned} a_p T_P &= \sum_N a_N T_N + S_u \\ a_p &= \left( \sum_N \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \right) + \left( \sum_M \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| \right) & a_N &= \frac{k_N A_N}{|\mathbf{d}_{PN}|} |\hat{\mathbf{n}}| \\ S_u &= \bar{S}V + \sum_M T_{\text{wall}} \left( \frac{k_w A_w}{|\mathbf{d}_{LP}|/2} |\hat{\mathbf{n}}| \right) \end{aligned} \quad (121)$$

Cells with multiple boundary faces will be considered further in the example problem in this chapter.

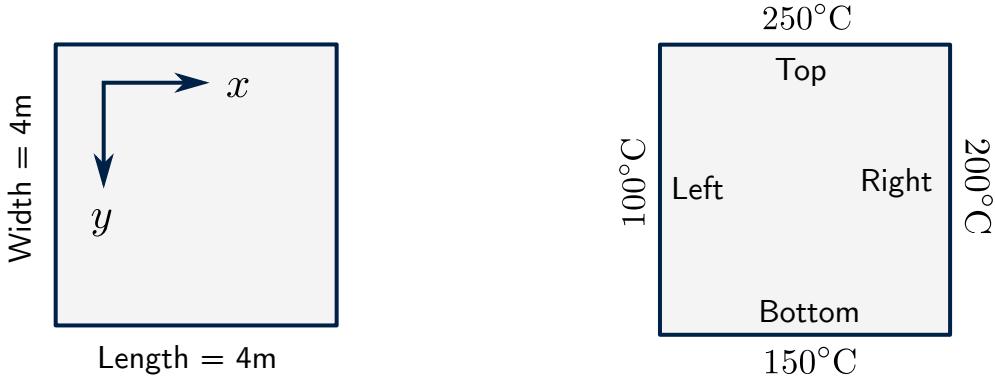


**Figure 16:** Cell numbering scheme for a 4x4 mesh. The mesh stencil around cell 10 is highlighted.

## Assembling the Matrices

When applying the finite volume method in 2D and 3D, a cell numbering scheme must be adopted to ensure that the correct cells are connected together. Figure 16 shows an example numbering scheme for a mesh with 16 cells ( $4 \times 4$ ). For each cell in the mesh, it is useful to visualise a stencil of 4 cells that surround the cell under consideration (also shown in Figure 16). This stencil allows us to identify the neighbouring cells more easily. As shown in the previous course, an algebraic equation is written for each cell in the mesh individually. For the mesh in Figure 16, the algebraic equations are:

$$\begin{aligned}
 a_{p1}T_1 &= a_{r2}T_2 + a_{b5}T_5 + S_{u1} \\
 a_{p2}T_2 &= a_{l1}T_1 + a_{r3}T_3 + a_{b6}T_6 + S_{u2} \\
 a_{p3}T_3 &= a_{l2}T_2 + a_{r4}T_4 + a_{b7}T_7 + S_{u3} \\
 a_{p4}T_4 &= a_{l3}T_3 + a_{b8}T_8 + S_{u4} \\
 a_{p5}T_5 &= a_{t1}T_1 + a_{r6}T_6 + a_{b9}T_9 + S_{u5} \\
 a_{p6}T_6 &= a_{l2}T_2 + a_{t5}T_5 + a_{r7}T_7 + a_{b10}T_{10} + S_{u6} \\
 a_{p7}T_7 &= a_{l3}T_3 + a_{t6}T_6 + a_{r8}T_8 + a_{b11}T_{11} + S_{u7} \\
 a_{p8}T_8 &= a_{l4}T_4 + a_{t7}T_7 + a_{b12}T_{12} + S_{u8} \\
 a_{p9}T_9 &= a_{r5}T_5 + a_{t10}T_{10} + a_{b13}T_{13} + S_{u9} \\
 a_{p10}T_{10} &= a_{l6}T_6 + a_{r9}T_9 + a_{t11}T_{11} + a_{b14}T_{14} + S_{u10} \\
 a_{p11}T_{11} &= a_{l7}T_7 + a_{r10}T_{10} + a_{t12}T_{12} + a_{b15}T_{15} + S_{u11} \\
 a_{p12}T_{12} &= a_{l8}T_8 + a_{t11}T_{11} + a_{b16}T_{16} + S_{u12} \\
 a_{p13}T_{13} &= a_{t9}T_9 + a_{r14}T_{14} + S_{u13} \\
 a_{p14}T_{14} &= a_{l10}T_{10} + a_{t13}T_{13} + a_{r15}T_{15} + S_{u14} \\
 a_{p15}T_{15} &= a_{l11}T_{11} + a_{t14}T_{14} + a_{r16}T_{16} + S_{u15} \\
 a_{p16}T_{16} &= a_{l12}T_{12} + a_{t15}T_{15} + S_{u16}
 \end{aligned} \tag{122}$$



**Figure 17:** Example geometry used for heat diffusion in a 2D plate.

As before, the equations are then arranged into matrix form.

$$\mathbf{A} = \begin{bmatrix} a_{p1} - a_{r2} & 0 & 0 & -a_{b5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -a_{l1} & a_{p1} - a_{r3} & 0 & 0 & -a_{b6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -a_{l2} & a_{p3} - a_{r4} & 0 & 0 & -a_{b7} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -a_{l3} & a_{p4} - a_{r5} & 0 & 0 & -a_{b8} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -a_{t1} & 0 & 0 & -a_{l4} & a_{p5} - a_{r6} & 0 & 0 & -a_{b9} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -a_{t2} & 0 & 0 & -a_{l5} & a_{p6} - a_{r7} & 0 & 0 & -a_{b10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -a_{t3} & 0 & 0 & -a_{l6} & a_{p7} - a_{r8} & 0 & 0 & -a_{b11} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -a_{t4} & 0 & 0 & -a_{l7} & a_{p8} - a_{r9} & 0 & 0 & -a_{b12} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_{t5} & 0 & 0 & -a_{l8} & a_{p9} - a_{r10} & 0 & 0 & -a_{b13} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_{t6} & 0 & 0 & -a_{l9} & a_{p10} - a_{r11} & 0 & 0 & -a_{b14} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -a_{t7} & 0 & 0 & -a_{l10} & a_{p11} - a_{r12} & 0 & 0 & -a_{b15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -a_{t8} & 0 & 0 & -a_{l11} & a_{p12} - a_{r13} & 0 & 0 & -a_{b16} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -a_{t9} & 0 & 0 & -a_{l12} & a_{p13} - a_{r14} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -a_{t10} & 0 & 0 & -a_{l13} & a_{p14} - a_{r15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -a_{t11} & 0 & 0 & -a_{l14} & a_{p15} - a_{r16} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -a_{t12} & 0 & 0 & 0 & -a_{l15} & a_{p16} \end{bmatrix}$$

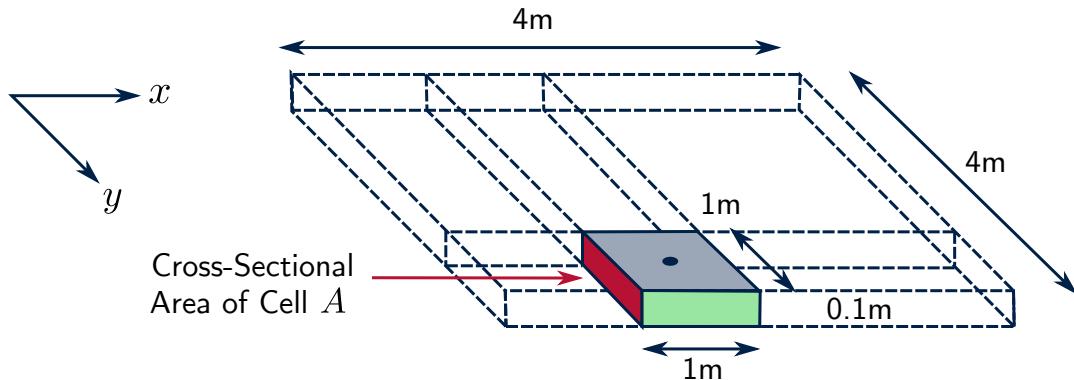
$$\mathbf{T} = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6 \ T_7 \ T_8 \ T_9 \ T_{10} \ T_{11} \ \dots \ \dots \ T_{12} \ T_{13} \ T_{14} \ T_{15} \ T_{16}]^T \quad (123)$$

$$\mathbf{B} = [S_{u1} \ S_{u2} \ S_{u3} \ S_{u4} \ S_{u5} \ S_{u6} \ S_{u7} \ S_{u8} \ S_{u9} \ S_{u10} \ S_{u11} \ \dots \ \dots \ S_{u12} \ S_{u13} \ S_{u14} \ S_{u15} \ S_{u16}]^T \quad (124)$$

In the same manner as the 1D flow case, the  $\mathbf{A}$  matrix has a diagonal banded structure. However, rather than having 2 off-diagonal bands (representing the left and right faces of the cell), the matrix now has 4 bands. The additional bands represent the connectivity to the top and bottom cells.

## Example Problem - Heat Diffusion in a 2D Plate

Consider steady-state diffusion of heat in a 2D plate, as shown in Figure 17. The plate has a length of 4m, a width of 4m, a thickness of 0.1m and a thermal conductivity of 100 W/mK.



**Figure 18:** A diagram to show the dimensions of a cell in the mesh. The cross-sectional area that is used to calculate the heat fluxes in the  $x$  direction is highlighted in red and the cross-sectional area that is used to calculate the heat fluxes in the  $y$  direction is highlighted in green.

The temperature on the sides of the plate are fixed at  $100^\circ\text{C}$ ,  $150^\circ\text{C}$ ,  $200^\circ\text{C}$  and  $250^\circ\text{C}$  on the left, bottom, right and top faces respectively. There is a constant heat source of  $1000 \text{ W/m}^3$  in the plate. The temperature field in the plate is governed by the 2D steady-state diffusion equation.

$$\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + S = 0 \quad (125)$$

### Step 1: Divide the Geometry into a Mesh

For the example in Figure 17, divide the geometry into a mesh of 16 quadrilateral cells (4 cells in each direction). These cells have equal length and equal width. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{4}{4} = 1\text{m} \quad (126)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d_{BP} = d_{PT} = d = 1\text{m} \quad (127)$$

As shown in Figure 18, the cross-sectional area of all of the cells (in the  $x$  and  $y$  directions) is:

$$A = 1.0 * 0.1 = 0.1\text{m}^2 \quad (128)$$

### Step 2: Assign Material Properties

The thermal conductivity  $k$  and the cross-sectional area  $A$  are the same for every cell in the mesh. Hence, the parameter  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 \text{ [W/K]} \quad (129)$$

The heat source in each cell is given by:

$$\bar{SV} = \bar{SL}_{\text{cell}}Wt = 1000 * 1 * 1 * 0.1 = 100 \text{ [W]} \quad (130)$$

### Step 3: Calculate Matrix Coefficients

Now calculate the coefficients required to assemble the matrices. The most straightforward way is to fill in the table of coefficients:

Cell	$a_L$	$a_R$	$a_b$	$a_t$	$S_p$	$S_u$	$a_p$
1	0	10	10	0	-40	7100	60
2	10	10	10	0	-20	5100	50
3	10	10	10	0	-20	5100	50
4	10	0	10	0	-40	9100	60
5	0	10	10	10	-20	2100	50
6	10	10	10	10	0	100	40
7	10	10	10	10	0	100	40
8	10	0	10	10	-20	4100	50
9	0	10	10	10	-20	2100	50
10	10	10	10	10	0	100	40
11	10	10	10	10	0	100	40
12	10	0	10	10	-20	4100	50
13	0	10	0	10	-40	5100	60
14	10	10	0	10	-20	3100	50
15	10	10	0	10	-20	3100	50
16	10	0	0	10	-40	7100	60

Alternatively, the summation notation summary table can be filled in (noting that an additional source  $\bar{S}V = 100W$  is required in each cell):

Face Type	$a_P$	$a_N$	$S_u$
Interior	10	10	0
Dirichlet (Left)	20	0	2000
Dirichlet (Bottom)	20	0	3000
Dirichlet (Right)	20	0	4000
Dirichlet (Top)	20	0	5000

### Step 4: Assemble the Matrices

Assign the coefficients to their correct location in the matrix.

$$\mathbf{A} = \begin{bmatrix} 60 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & 50 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -10 & 50 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -10 & 60 & 0 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & 0 & 0 & 0 & 50 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -10 & 0 & 0 & -10 & 40 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -10 & 0 & 0 & -10 & 40 & -10 & 0 & 0 & -10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -10 & 0 & 0 & -10 & 50 & 0 & 0 & 0 & -10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -10 & 0 & 0 & 0 & 50 & -10 & 0 & 0 & -10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & -10 & 40 & -10 & 0 & 0 & -10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & -10 & 40 & -10 & 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & 0 & -10 & 50 & 0 & 0 & -10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & 0 & 60 & -10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & -10 & 50 & -10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & -10 & 50 & -10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & -10 & 60 \end{bmatrix} \quad (131)$$

$$\mathbf{T} = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6 \ T_7 \ T_8 \ T_9 \ T_{10} \ T_{11} \ \dots \ \dots \ T_{12} \ T_{13} \ T_{14} \ T_{15} \ T_{16}]^T \quad (132)$$

$$\mathbf{B} = [7100 \ 5100 \ 5100 \ 9100 \ 2100 \ 100 \ 100 \ 4100 \ 2100 \ 100 \ 100 \ \dots \ \dots \ 4100 \ 5100 \ 3100 \ 3100 \ 7100]^T \quad (133)$$

### Step 5: Solve the Equations

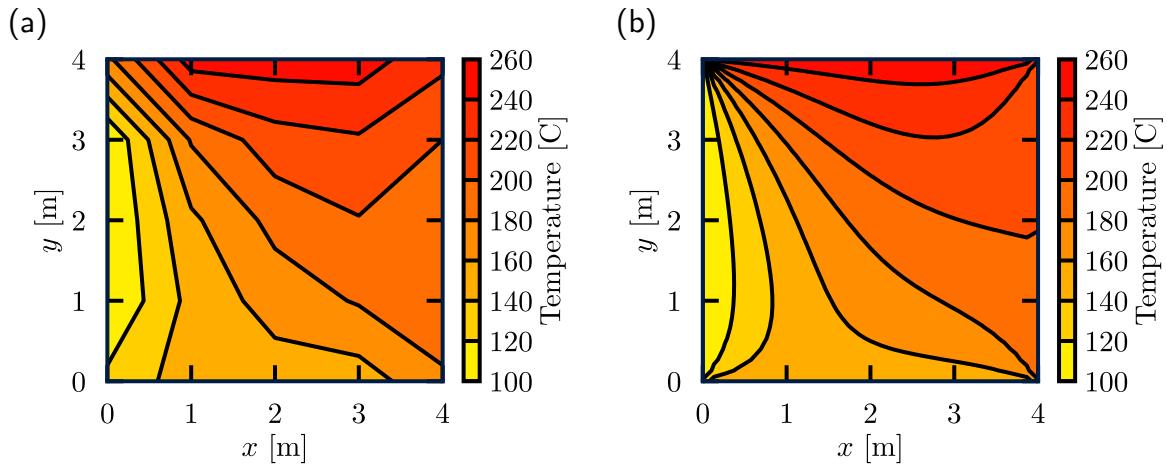
Now that the matrices have been assembled, the matrix equation can be solved with an appropriate *iterative method*. As with the previous course, different algorithms to solve the matrix equation  $\mathbf{AT} = \mathbf{B}$  will not be considered and the default algorithms used by Excel and Python will be used instead.

#### Run the Example Problem Yourself!

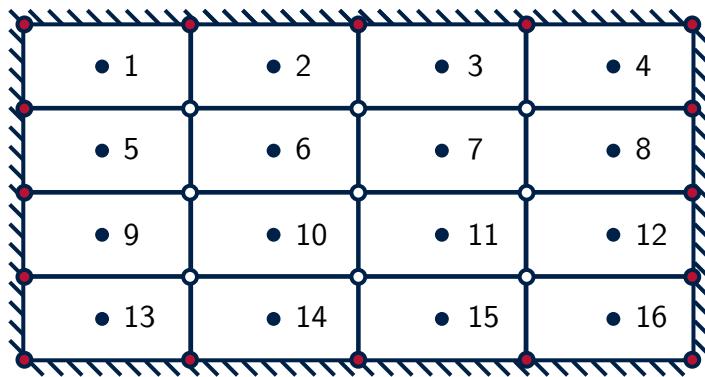
Now, open either the Excel spreadsheets or the Python source code and solve the problem yourself.

Excel `solve2DDiffusionEquation.xlsx`

Python `solve2DDiffusionEquation.py`



**Figure 19:** Temperature variation across the 2D plate with (a) a coarse mesh and (b) a fine mesh.



**Figure 20:** A diagram to show the centroids numbering scheme used in the mesh. The centroids are shown as dark blue circles, the boundary nodes as red circles and the interior nodes as white circles.

## Results

Figure 19 shows the temperature variation across the plate with (a) a coarse mesh and (b) a fine mesh. To plot the temperature variation as a contour plot, it is more convenient to plot the temperature at the mesh **nodes** rather than the mesh **centroids**. As shown in Figure 20, this is because the boundary temperatures are specified at the nodes (shown in red) and these would be missed if a contour plot was generated from the mesh centroids (shown in blue).

The CFD solution is computed at the cell centroids. Hence, the interior nodal values (shown in white in Figure 20) have to be computed by interpolating between the centroid values (shown in blue). This interpolation has already been carried out in the Excel spreadsheet and the python source code for you.

## Heat Balancing

Now that a CFD solution has been computed, a heat balance can be written for every cell in the mesh. In the same manner as the previous chapter, the sum of the heat fluxes into the

cell and the heat generated in the cell must be equal to zero for energy to be conserved.

$$0 = k_r A_r \left( \frac{T_R - T_P}{d_{PR}} \right) n_x + k_l A_l \left( \frac{T_P - T_L}{d_{LP}} \right) n_x \\ + k_t A_t \left( \frac{T_T - T_P}{d_{PT}} \right) n_y + k_b A_b \left( \frac{T_P - T_B}{d_{BP}} \right) n_y + \bar{S}V \quad (134)$$

From Fourier's Law, the heat conduction across each face is given by:

$$Q_l = -k_l A_l \left( \frac{T_P - T_L}{d_{LP}} \right) n_x \quad Q_r = -k_r A_r \left( \frac{T_R - T_P}{d_{PR}} \right) n_x \\ Q_b = -k_b A_b \left( \frac{T_P - T_B}{d_{BP}} \right) n_y \quad Q_t = -k_t A_t \left( \frac{T_T - T_P}{d_{PT}} \right) n_y \quad (135)$$

The negative sign is required to ensure that the heat flows in the opposite direction to the temperature gradient and the unit normal vectors ensure that positive heat fluxes are out of the cell. By substituting Fourier's Law (equation 135) into the finite volume discretisation (equation 134), the heat balance for each cell can be written in concise form as:

$$-Q_r - Q_l - Q_t - Q_b + \bar{S}V = 0 \quad (136)$$

$$\sum_N (-Q) + \bar{S}V = 0 \quad (137)$$

The error in the heat flux balance for 2D quadrilateral cells can therefore be written as:

$$\text{Error} = -Q_r - Q_l - Q_t - Q_b + \bar{S}V \quad (138)$$

The table below summarises the heat balance for every cell in the mesh. The heat fluxes across the boundaries of the plate are highlighted in blue.

Cell Heat Flux Balance Table

Cell	$Q_l$ [W]	$Q_r$ [W]	$Q_t$ [W]	$Q_b$ [W]	$\bar{S}V$ [W]	Error
1	<b>1575.0</b>	-427.3	<b>-1425.0</b>	377.3	100	0
2	427.3	-109.2	<b>-570.4</b>	352.3	100	0
3	109.2	74.4	<b>-351.9</b>	268.3	100	0
4	-74.4	<b>499.4</b>	<b>-500.6</b>	175.6	100	0
5	<b>820.4</b>	-452.3	-377.3	109.2	100	0
6	452.3	-193.3	-352.3	193.3	100	0
7	193.3	-18.3	-268.3	193.3	100	0
8	18.3	<b>148.1</b>	-175.6	109.2	100	0
9	<b>601.9</b>	-368.3	-109.2	-24.4	100	0
10	368.3	-193.3	-193.3	118.3	100	0
11	192.3	-102.3	-193.3	202.3	100	0
12	102.3	<b>-70.4</b>	-109.2	177.3	100	0
13	<b>650.6</b>	-225.6	24.4	<b>-349.4</b>	100	0
14	225.6	-109.2	-118.3	<b>101.9</b>	100	0
15	109.2	-127.3	-202.3	<b>320.4</b>	100	0
16	127.3	<b>-425.0</b>	-177.3	<b>575.0</b>	100	0

The cell heat flux balance table can also be used to calculate the total heat flux out of each of the faces of the plate (across all the boundaries). By summing the heat flux out of each of the boundary faces (highlighted in blue in the table above):

Heat Flux Summary Table		
Boundary	Heat Flux Out [W]	Total [W]
Left	$1575.0 + 820.4 + 601.9 + 650.6$	3647.9
Right	$499.4 + 148.1 - 70.4 - 425.0$	152.1
Bottom	$-349.4 + 101.9 + 320.4 + 575.0$	647.9
Top	$-1425 - 570.4 - 351.9 - 500.6$	-2847.9
Total		<b>1600</b>

Once again positive heat fluxes indicate heat travelling out of the plate across the boundary, while negative heat fluxes indicate heat travelling into the plate. As the coldest face, the left face of the plate ( $100^{\circ}\text{C}$ ) experiences the largest heat flux out of the plate. Conversely, the top face of the plate experiences a negative heat flux as heat is drawn into the plate by the hot boundary temperature ( $250^{\circ}\text{C}$ ). The total of these heat fluxes (1600W) balances the total heat generated in the plate (1600W), so energy is conserved. As the mesh is refined, the same total heat flux (1600W) is conserved. However, the temperature distribution in the plate and the heat flux across each of the individual cells changes as a result of the mesh refinement.

# Chapter eight The finite volume method for unsteady flows

## 8.1

### Introduction

Having finished the task of developing the finite volume method for steady flows we are now in a position to consider the more complex category of time-dependent problems. The conservation law for the transport of a scalar in an unsteady flow has the general form

$$\frac{\partial}{\partial t}(\rho\phi) + \operatorname{div}(\rho\mathbf{u}\phi) = \operatorname{div}(\Gamma \operatorname{grad} \phi) + S_\phi \quad (8.1)$$

The first term of the equation represents the rate of change term and is zero for steady flows. To predict transient problems we must retain this term in the discretisation process. The finite volume integration of equation (8.1) over a control volume (CV) must be augmented with a further integration over a finite time step  $\Delta t$ . By replacing the volume integrals of the convective and diffusive terms with surface integrals as before (see section 2.5) and changing the order of integration in the rate of change term we obtain

$$\begin{aligned} & \int_{\text{CV}} \left( \int_t^{t+\Delta t} \frac{\partial}{\partial t}(\rho\phi) dt \right) dV + \int_t^{t+\Delta t} \left( \int_A \mathbf{n} \cdot (\rho\mathbf{u}\phi) dA \right) dt \\ &= \int_t^{t+\Delta t} \left( \int_A \mathbf{n} \cdot (\Gamma \operatorname{grad} \phi) dA \right) dt + \int_t^{t+\Delta t} \int_{\text{CV}} S_\phi dV dt \end{aligned} \quad (8.2)$$

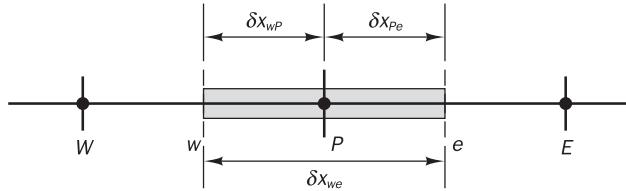
So far we have made no approximations but to make progress we need techniques for evaluating the integrals. The control volume integration is essentially the same as in steady flows and the measures explained in Chapters 4 and 5 are again adopted to ensure successful treatment of convection, diffusion and source terms. Here we focus our attention on methods necessary for the time integration. The process is illustrated below using the one-dimensional unsteady diffusion (heat transfer) equation and is later extended to multi-dimensional unsteady diffusion and convection-diffusion problems.

## 8.2

### One-dimensional unsteady heat conduction

Unsteady one-dimensional heat conduction is governed by the equation

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + S \quad (8.3)$$

**Figure 8.1**

In addition to the usual variables we have  $c$ , the specific heat of the material (J/kg.K).

Consider the one-dimensional control volume in Figure 8.1. Integration of equation (8.3) over the control volume and over a time interval from  $t$  to  $t + \Delta t$  gives

$$\int_t^{t+\Delta t} \int_{CV} \rho c \frac{\partial T}{\partial t} dV dt = \int_t^{t+\Delta t} \int_{CV} \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) dV dt + \int_t^{t+\Delta t} \int_{CV} S dV dt \quad (8.4)$$

This may be written as

$$\begin{aligned} \int_w^e \left[ \int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t} dt \right] dV &= \int_t^{t+\Delta t} \left[ \left( kA \frac{\partial T}{\partial x} \right)_e - \left( kA \frac{\partial T}{\partial x} \right)_w \right] dt \\ &\quad + \int_t^{t+\Delta t} \bar{S} \Delta V dt \end{aligned} \quad (8.5)$$

In equation (8.5),  $A$  is the face area of the control volume,  $\Delta V$  is its volume, which is equal to  $A\Delta x$ , where  $\Delta x = \delta x_{we}$  is the width of the control volume, and  $\bar{S}$  is the average source strength. If the temperature at a node is assumed to prevail over the whole control volume, the left hand side can be written as

$$\int_{CV} \left[ \int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t} dt \right] dV = \rho c (T_p - T_p^o) \Delta V \quad (8.6)$$

In equation (8.6) superscript ‘ $o$ ’ to refers to temperatures at time  $t$ ; temperatures at time level  $t + \Delta t$  are not superscripted. The same result as (8.6) would be obtained by substituting  $(T_p - T_p^o)/\Delta t$  for  $\partial T/\partial t$ , so this term has been discretised using a first-order (backward) differencing scheme. Higher-order schemes, which may be used to discretise this term, will be discussed briefly later in this chapter. If we apply central differencing to the diffusion terms on the right hand side equation (8.5) may be written as

$$\begin{aligned} \rho c (T_p - T_p^o) \Delta V &= \int_t^{t+\Delta t} \left[ \left( k_e A \frac{T_E - T_p}{\delta x_{PE}} \right) - \left( k_w A \frac{T_p - T_W}{\delta x_{WP}} \right) \right] dt \\ &\quad + \int_t^{t+\Delta t} \bar{S} \Delta V dt \end{aligned} \quad (8.7)$$

To evaluate the right hand side of this equation we need to make an assumption about the variation of  $T_P$ ,  $T_E$  and  $T_W$  with time. We could use temperatures at time  $t$  or at time  $t + \Delta t$  to calculate the time integral or, alternatively, a combination of temperatures at time  $t$  and  $t + \Delta t$ . We may generalise the approach by means of a weighting parameter  $\theta$  between 0 and 1 and write the integral  $I_T$  of temperature  $T_P$  with respect to time as

$$I_T = \int_t^{t+\Delta t} T_P dt = [\theta T_P + (1 - \theta) T_P^\theta] \Delta t \quad (8.8)$$

Hence

$\theta$	0	1/2	1
$I_T$	$T_P^\theta \Delta t$	$\frac{1}{2}(T_P + T_P^\theta) \Delta t$	$T_P \Delta t$

We have highlighted the following values of integral  $I_T$ : if  $\theta = 0$  the temperature at (old) time level  $t$  is used; if  $\theta = 1$  the temperature at new time level  $t + \Delta t$  is used; and finally if  $\theta = 1/2$ , the temperatures at  $t$  and  $t + \Delta t$  are equally weighted.

Using formula (8.8) for  $T_W$  and  $T_E$  in equation (8.7), and dividing by  $A\Delta t$  throughout, we have

$$\begin{aligned} \rho c \left( \frac{T_P - T_P^\theta}{\Delta t} \right) \Delta x &= \theta \left[ \frac{k_e(T_E - T_P)}{\delta x_{PE}} - \frac{k_w(T_P - T_W)}{\delta x_{WP}} \right] \\ &\quad + (1 - \theta) \left[ \frac{k_e(T_E^\theta - T_P^\theta)}{\delta x_{PE}} - \frac{k_w(T_P^\theta - T_W^\theta)}{\delta x_{WP}} \right] + \bar{S} \Delta x \end{aligned} \quad (8.9)$$

which may be rearranged to give

$$\begin{aligned} &\left[ \rho c \frac{\Delta x}{\Delta t} + \theta \left( \frac{k_e}{\delta x_{PE}} + \frac{k_w}{\delta x_{WP}} \right) \right] T_P \\ &= \frac{k_e}{\delta x_{PE}} [\theta T_E + (1 - \theta) T_E^\theta] + \frac{k_w}{\delta x_{WP}} [\theta T_W + (1 - \theta) T_W^\theta] \\ &\quad + \left[ \rho c \frac{\Delta x}{\Delta t} - (1 - \theta) \frac{k_e}{\delta x_{PE}} - (1 - \theta) \frac{k_w}{\delta x_{WP}} \right] T_P^\theta + \bar{S} \Delta x \end{aligned} \quad (8.10)$$

Now we identify the coefficients of  $T_W$  and  $T_E$  as  $a_W$  and  $a_E$  and write equation (8.10) in the familiar standard form:

$$a_P T_P = a_W [\theta T_W + (1 - \theta) T_W^\theta] + a_E [\theta T_E + (1 - \theta) T_E^\theta] + [a_P^\theta - (1 - \theta)a_W - (1 - \theta)a_E] T_P^\theta + b$$

(8.11)

where

$$a_P = \theta(a_W + a_E) + a_P^\theta$$

and

$$a_p^o = \rho c \frac{\Delta x}{\Delta t}$$

with

$a_W$	$a_E$	$b$
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$	$\bar{S}\Delta x$

The exact form of the final discretised equation depends on the value of  $\theta$ . When  $\theta$  is zero, we only use temperatures  $T_p^o$ ,  $T_W^o$  and  $T_E^o$  at the old time level  $t$  on the right hand side of equation (8.11) to evaluate  $T_p$  at the new time and the resulting scheme is called **explicit**. When  $0 < \theta \leq 1$  temperatures at the new time level are used on both sides of the equation and the resulting schemes are called **implicit**. The extreme case of  $\theta = 1$  is termed **fully implicit** and the case corresponding to  $\theta = 1/2$  is called the **Crank–Nicolson** scheme (Crank and Nicolson, 1947).

### 8.2.1 Explicit scheme

In the explicit scheme the source term is linearised as  $b = S_u + S_p T_p^o$ . Now the substitution of  $\theta = 0$  into (8.11) gives the **explicit discretisation** of the unsteady conductive heat transfer equation

$$a_p T_p = a_W T_W^o + a_E T_E^o + [a_p^o - (a_W + a_E - S_p)] T_p^o + S_u \quad (8.12)$$

where

$$a_p = a_p^o$$

and

$$a_p^o = \rho c \frac{\Delta x}{\Delta t}$$

$a_W$	$a_E$
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$

The right hand side of equation (8.12) only contains values at the old time step so the left hand side can be calculated by forward marching in time. The scheme is based on backward differencing and its Taylor series truncation error accuracy is first-order with respect to time. As explained in Chapter 5, all coefficients need to be positive in the discretised equation. The coefficient of  $T_p^o$  may be viewed as the neighbour coefficient connecting the values at the old time level to those at the new time level. For this coefficient to be positive we must have  $a_p^o - a_W - a_E > 0$ . For constant  $k$  and uniform grid spacing,  $\delta x_{PE} = \delta x_{WP} = \Delta x$ , this condition may be written as

$$\rho c \frac{\Delta x}{\Delta t} > \frac{2k}{\Delta x} \quad (8.13a)$$

or

$$\Delta t < \rho c \frac{(\Delta x)^2}{2k} \quad (8.13b)$$

This inequality sets a stringent maximum limit to the time step size and represents a serious limitation for the explicit scheme. It becomes very expensive to improve spatial accuracy because the maximum possible time step needs to be reduced as the square of  $\Delta x$ . Consequently, this method is not recommended for general transient problems. Explicit schemes with greater formal accuracy than the above one have been designed. Examples are the Richardson and DuFort–Frankel methods, which use temperatures at more than two time levels. These methods also have fewer stability restrictions than the ordinary explicit method. Details of such schemes can be found in Abbot and Basco (1990), Anderson *et al* (1984) and Fletcher (1991). Nevertheless, provided that the time step size is chosen with care, the explicit scheme described above is efficient for simple conduction calculations. This will be illustrated through an example in section 8.3.

### 8.2.2 Crank–Nicolson scheme

The Crank–Nicolson method results from setting  $\theta = 1/2$  in equation (8.11). The source term is linearised as  $b = S_u + \frac{1}{2}S_p T_p + \frac{1}{2}S_p T_p^o$ . Now the discretised unsteady heat conduction equation is

$$a_p T_p = a_E \left[ \frac{T_E + T_E^o}{2} \right] + a_W \left[ \frac{T_W + T_W^o}{2} \right] + \left[ a_p^o - \frac{a_E}{2} - \frac{a_W}{2} \right] T_p^o + S_u + \frac{1}{2} S_p T_p^o \quad (8.14)$$

where

$$a_p = \frac{1}{2}(a_W + a_E) + a_p^o - \frac{1}{2}S_p$$

and

$$a_p^o = \rho c \frac{\Delta x}{\Delta t}$$

$a_W$	$a_E$
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$

Since more than one unknown value of  $T$  at the new time level is present in equation (8.14) the method is implicit, and simultaneous equations for all node points need to be solved at each time step. Although schemes with  $\frac{1}{2} \leq \theta \leq 1$ , including the Crank–Nicolson scheme, are unconditionally stable for all values of the time step (Fletcher, 1991), it is more important to ensure that all coefficients are positive for physically realistic and bounded results. This is the case if the coefficient of  $T_p^\theta$  satisfies the following condition:

$$a_p^\theta > \left[ \frac{a_E + a_W}{2} \right]$$

which leads to

$$\Delta t < \rho c \frac{\Delta x^2}{k} \quad (8.15)$$

This time step limitation is only slightly less restrictive than (8.13) associated with the explicit method. The Crank–Nicolson method is based on central differencing and hence it is second-order accurate in time. With sufficiently small time steps it is possible to achieve considerably greater accuracy than with the explicit method. The overall accuracy of a computation depends also on the spatial differencing practice so the Crank–Nicolson scheme is normally used in conjunction with spatial central differencing.

### 8.2.3 The fully implicit scheme

When the value of  $\theta$  is set equal to 1 we obtain the fully implicit scheme. Now the source term is linearised as  $b = S_u + S_p T_p$ . The discretised equation is

$$a_p T_p = a_W T_W + a_E T_E + a_p^\theta T_p^\theta + S_u \quad (8.16)$$

where

$$a_p = a_p^\theta + a_W + a_E - S_p$$

and

$$a_p^\theta = \rho c \frac{\Delta x}{\Delta t}$$

with

$a_W$	$a_E$
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$

Both sides of the equation contain temperatures at the new time step, and a system of algebraic equations must be solved at each time level (see Example 8.2). The time marching procedure starts with a given initial field

of temperatures  $T^0$ . The system of equations (8.16) is solved after selecting time step  $\Delta t$ . Next the solution  $T$  is assigned to  $T^0$  and the procedure is repeated to progress the solution by a further time step.

It can be seen that all coefficients are positive, which makes the implicit scheme unconditionally stable for any size of time step. Since the accuracy of the scheme is only first-order in time, small time steps are needed to ensure the accuracy of results. The implicit method is recommended for general-purpose transient calculations because of its robustness and unconditional stability.

### 8.3

### Illustrative examples

#### Example 8.1

We now demonstrate the properties of the explicit and implicit discretisation schemes by means of a comparison of numerical results for a one-dimensional unsteady conduction example with analytical solutions to assess the accuracy of the methods.

A thin plate is initially at a uniform temperature of 200°C. At a certain time  $t = 0$  the temperature of the east side of the plate is suddenly reduced to 0°C. The other surface is insulated. Use the explicit finite volume method in conjunction with a suitable time step size to calculate the transient temperature distribution of the slab and compare it with the analytical solution at time (i)  $t = 40$  s, (ii)  $t = 80$  s and (iii)  $t = 120$  s. Recalculate the numerical solution using a time step size equal to the limit given by (8.13) for  $t = 40$  s and compare the results with the analytical solution. The data are: plate thickness  $L = 2$  cm, thermal conductivity  $k = 10$  W/m.K and  $\rho c = 10 \times 10^6$  J/m<sup>3</sup>.K.

#### Solution

The one-dimensional transient heat conduction equation is

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) \quad (8.17)$$

and the initial conditions are

$$T = 200 \text{ at } t = 0$$

and the boundary conditions are

$$\frac{\partial T}{\partial x} = 0 \text{ at } x = 0, t > 0$$

$$T = 0 \text{ at } x = L, t > 0$$

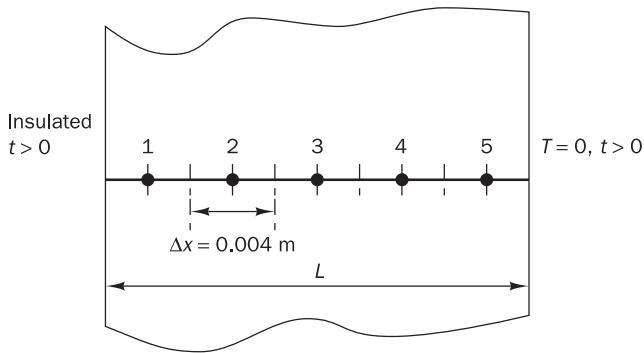
The analytical solution is given in Özışık (1985) as

$$\frac{T(x, t)}{200} = \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{2n-1} \exp(-\alpha \lambda_n^2 t) \cos(\lambda_n x) \quad (8.18)$$

$$\text{where } \lambda_n = \frac{(2n-1)\pi}{2L} \text{ and } \alpha = k/\rho c$$

The numerical solution with the explicit method is generated by dividing the domain width  $L$  into five equal control volumes with  $\Delta x = 0.004$  m. The resulting one-dimensional grid is shown in Figure 8.2.

**Figure 8.2** Geometry for Example 8.1



The discretised form of governing equation (8.17) for an internal control volume using the explicit method is given by (8.12). Control volumes 1 and 5 adjoin boundaries, so the links are cut in the direction of the boundary and the boundary fluxes are included in the source terms. At the control volume 1, the west boundary is insulated: hence the flux across that boundary is zero. We modify equation (8.9) where the physics can be most easily discerned. The discretised equation at node 1 becomes

$$\rho c \frac{(T_p - T_p^\circ)}{\Delta t} \Delta x = \left[ \frac{k}{\Delta x} (T_E^\circ - T_p^\circ) \right] - 0 \quad (8.19)$$

For time  $t > 0$ , the temperature of the east boundary of control volume 5 is constant (say  $T_B$ ). The discretised equation at node 5 becomes

$$\rho c \frac{(T_p - T_p^\circ)}{\Delta t} \Delta x = \left[ \frac{k}{\Delta x / 2} (T_B - T_p^\circ) \right] - \left[ \frac{k}{\Delta x} (T_p^\circ - T_W^\circ) \right] \quad (8.20)$$

All discretised equations can now be written in standard form:

$$a_p T_p = a_W T_W^\circ + a_E T_E^\circ + [a_p^\circ - (a_W + a_E)] T_p^\circ + S_u \quad (8.21)$$

where

$$a_p = a_p^\circ = \rho c \frac{\Delta x}{\Delta t}$$

and

Node	$a_W$	$a_E$	$S_u$
1	0	$k / \Delta x$	0
2, 3, 4	$k / \Delta x$	$k / \Delta x$	0
5	$k / \Delta x$	0	$\frac{2k}{\Delta x} (T_B - T_p^\circ)$

The time step for the explicit method is subject to the condition that

$$\Delta t < \frac{\rho c (\Delta x)^2}{2k}$$

$$\Delta t < \frac{10 \times 10^6 (0.004)^2}{2 \times 10}$$

$$\Delta t < 8 \text{ s}$$

Let us select  $\Delta t = 2$  s. Substituting numerical values we have

$$\frac{k}{\Delta x} = \frac{10}{0.004} = 2500$$

$$\rho c \frac{\Delta x}{\Delta t} = 10 \times 10^6 \times \frac{0.004}{2} = 20000$$

After substitution of numerical values and some simplification the discretisation equations for the various nodes are

$$\begin{aligned} \text{Node 1: } & 200T_P = 25T_E^0 + 175T_P^0 \\ \text{Nodes 2–4: } & 200T_P = 25T_W^0 + 25T_E^0 + 150T_P^0 \\ \text{Node 5: } & 200T_P = 25T_W^0 + 125T_P^0 \end{aligned} \quad (8.22)$$

Starting with the initial condition where all the nodes are at a temperature of 200°C, the solution at each time step is obtained using equations (8.22). Although the calculations are not complicated, their number is large and they are most effectively carried out by a computer program. Table 8.1 gives a sample of the calculations for the first two time steps.

**Table 8.1** Specimen calculations for the explicit method

Time	Node 1	Node 2	Node 3	Node 4	Node 5
$t = 0$ s	$T_1^0 = 200$	$T_2^0 = 200$	$T_3^0 = 200$	$T_4^0 = 200$	$T_5^0 = 200$
	$200T_1^1 = 25 \times 200 + 175 \times 200$	$200T_2^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_3^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_4^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_5^1 = 25 \times 200 + 125 \times 200$
$t = 2$ s	$T_1^1 = 200$	$T_2^1 = 200$	$T_3^1 = 200$	$T_4^1 = 200$	$T_5^1 = 150$
	$200T_1^2 = 25 \times 200 + 175 \times 200$	$200T_2^2 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_3^2 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_4^2 = 25 \times 200 + 25 \times 150 + 150 \times 200$	$200T_5^2 = 25 \times 200 + 125 \times 150$
$t = 4$ s	$T_1^2 = 200$	$T_2^2 = 200$	$T_3^2 = 200$	$T_4^2 = 193.75$	$T_5^2 = 118.75$

Note: Subscripts denote the node number, superscripts denote the time step

Table 8.2 shows the results for 10 consecutive time steps and Table 8.3 shows the numerical and analytical results at times 40, 80 and 120 s. As can be seen from the error analysis, the results are in good agreement with the analytical solution. Figure 8.3 shows the comparison in a graphical form.

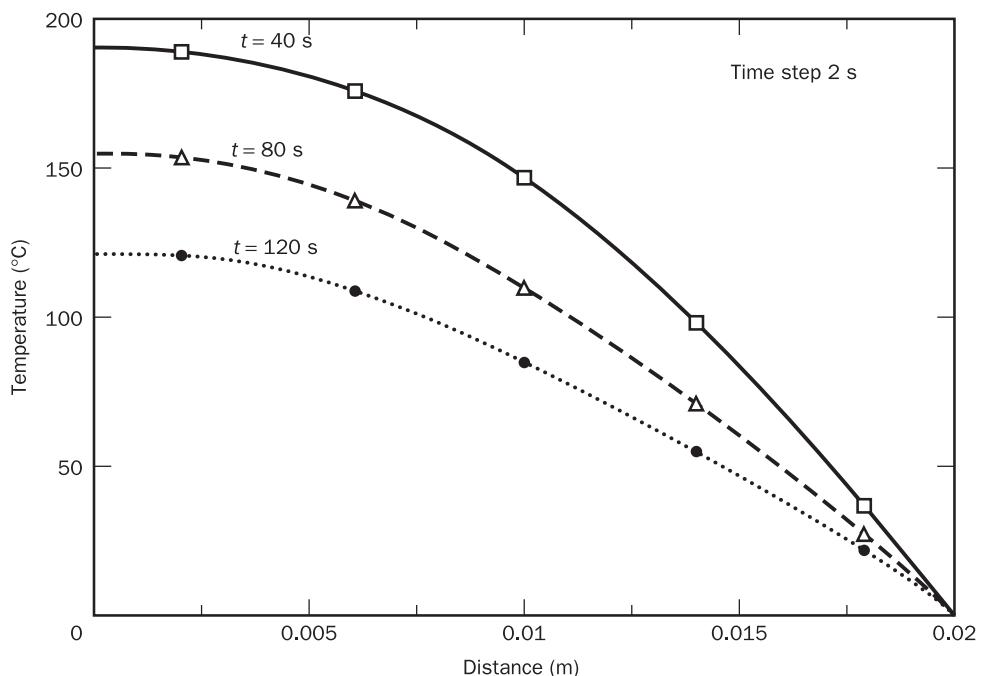
Figure 8.4 shows the solution for time  $t = 40$  s with a time step of 8 s. The previous result with a step size of 2 s and the exact solution are also shown for comparison. We conclude that a time step equal to the limiting value of 8 s gives a very inaccurate and unrealistic numerical solution that oscillates about the exact solution.

**Table 8.2** Results for Example 8.1 (explicit method)

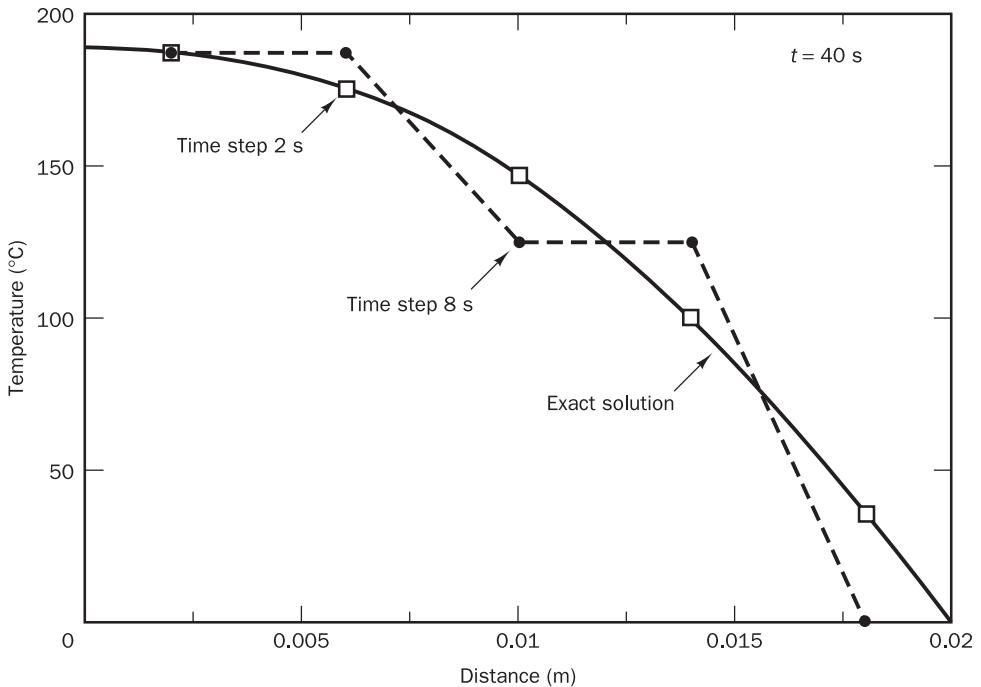
Time step	Time (s)	Node number						
		1	2	3	4	5		
		x = 0.0	x = 0.002	x = 0.006	x = 0.01	x = 0.014	x = 0.016	x = 0.018
0	0	200	200	200	200	200	200	200
1	2	200	200	200	200	200	150	0
2	4	200	200	200	200	193.75	118.75	0
3	6	200	200	200	199.21	185.16	98.43	0
4	8	200	200	199.9	197.55	176.07	84.66	0
5	10	199.98	199.98	199.62	195.16	167.33	74.92	0
6	12	199.94	199.94	199.11	192.24	159.26	67.74	0
7	14	199.83	199.83	198.35	188.98	151.94	62.24	0
8	16	199.65	199.65	197.36	185.52	145.36	57.89	0
9	18	199.37	199.37	196.17	181.98	139.45	54.35	0
10	20	198.97	198.97	194.79	178.44	134.12	51.40	0

**Table 8.3**

Node	Time = 40 s			Time = 80 s			Time = 120 s		
	Numerical	Analytical	% error	Numerical	Analytical	% error	Numerical	Analytical	% error
1	188.64	188.39	-0.13	153.33	152.65	-0.43	120.53	119.87	-0.55
2	176.41	175.76	-0.36	139.05	138.36	-0.50	108.82	108.21	-0.56
3	148.29	147.13	-0.79	111.29	110.63	-0.59	86.47	85.96	-0.58
4	100.76	99.50	-1.26	72.06	71.56	-0.69	55.58	55.25	-0.60
5	35.94	35.38	-1.57	24.96	24.77	-0.75	19.16	19.05	-0.59

**Figure 8.3** Comparison of numerical and analytical solutions at different times

**Figure 8.4** Comparison of results obtained using different time step values



### Example 8.2

Solve the problem of Example 8.1 again using the fully implicit method and compare the explicit and implicit method solutions for a time step of 8 s.

#### Solution

Let us use the same grid arrangement as in Figure 8.2. The fully implicit method describes events at internal control volumes 2, 3 and 4 by means of discretised equation (8.16). Boundary control volumes 1 and 5 again need special treatment. Upon incorporating the boundary conditions into equation (8.9) we get for node 1

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[ \frac{k}{\Delta x} (T_E - T_P) \right] - 0 \quad (8.23)$$

and for node 5

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[ \frac{k}{\Delta x / 2} (T_B - T_P) \right] - \left[ \frac{k}{\Delta x} (T_P - T_W) \right] \quad (8.24)$$

The discretised equations are written in standard form:

$$a_P T_P = a_W T_W + a_E T_E + a_P^o T_P^o + S_u \quad (8.25)$$

where

$$a_P = a_W + a_E + a_P^o - S_P$$

and

$$a_P^o = \rho c \frac{\Delta x}{\Delta t}$$

and

<i>Node</i>	$a_W$	$a_E$	$S_P$	$S_u$
1	0	$k/\Delta x$	0	0
2, 3, 4	$k/\Delta x$	$k/\Delta x$	0	0
5	$k/\Delta x$	0	$-\frac{2k}{\Delta x}$	$\frac{2k}{\Delta x}T_B$

Although the implicit method permits large values for the time step  $\Delta t$ , we will use reasonably small time steps of 2 s to ensure good accuracy. The grid spacing and other data are as before so again we have

$$\frac{k}{\Delta x} = \frac{10}{0.004} = 2500$$

$$\rho c \frac{\Delta x}{\Delta t} = 10 \times 10^6 \times \frac{0.004}{2} = 20000$$

After substitution of numerical values and the necessary simplification the discretised equations for the various nodes are

$$\text{Node 1: } 225T_P = 25T_E + 200T_P^\circ$$

$$\text{Nodes 2–4: } 250T_P = 25T_W + 25T_E + 200T_P^\circ$$

$$\text{Node 5: } 275T_P = 25T_W + 200T_P^\circ + 50T_B$$

Noting that  $T_B = 0$ , the set of equations to be solved at each time step is

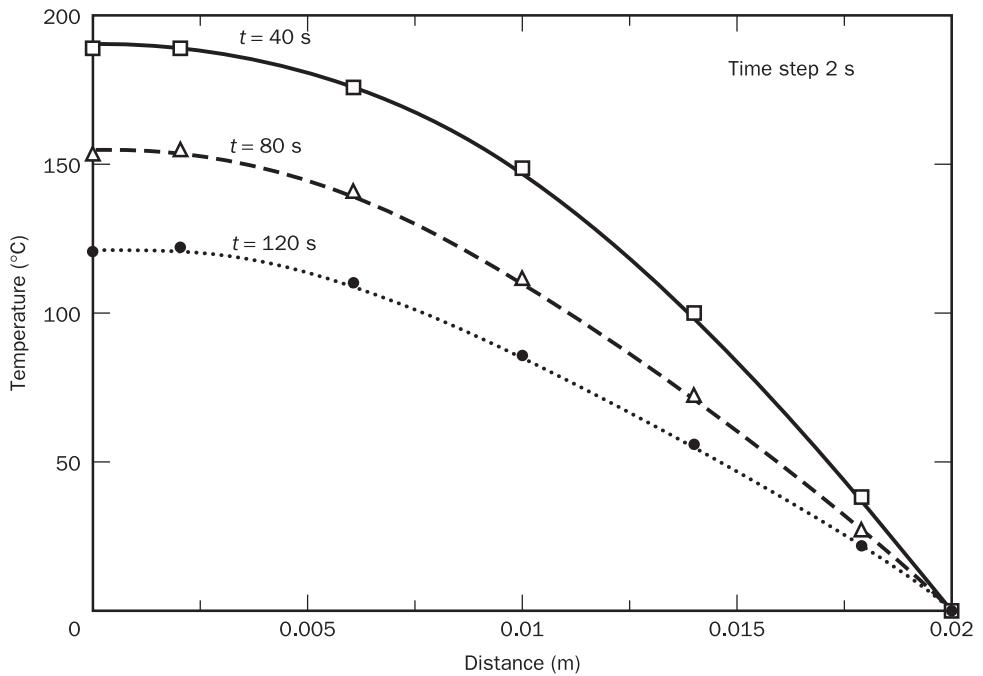
$$\begin{bmatrix} 225 & -25 & 0 & 0 & 0 \\ -25 & 250 & -25 & 0 & 0 \\ 0 & -25 & 250 & -25 & 0 \\ 0 & 0 & -25 & 250 & -25 \\ 0 & 0 & 0 & -25 & 275 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 200T_1^\circ \\ 200T_2^\circ \\ 200T_3^\circ \\ 200T_4^\circ \\ 200T_5^\circ \end{bmatrix} \quad (8.26)$$

The matrix form emphasises that the equations for each point contain unknown neighbouring temperatures. The explicit scheme involves a straightforward evaluation of a single algebraic equation to find each new nodal temperature, but the fully implicit method requires the (more expensive) solution of system (8.26) at each time level. The values of temperature at the previous time level are used to calculate the right hand side. Table 8.4 and Figure 8.5 show that the numerical results again compare favourably with the analytical solution.

**Table 8.4**

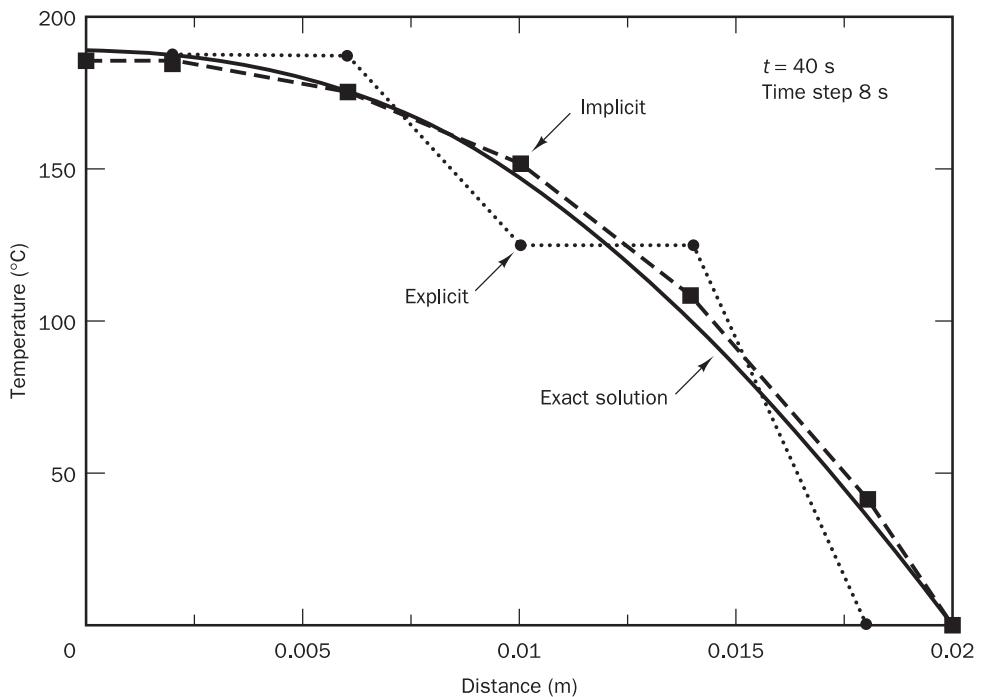
<i>Node</i>	<i>Time</i> = 40 s			<i>Time</i> = 80 s			<i>Time</i> = 120 s		
	<i>Numerical</i>	<i>Analytical</i>	% error	<i>Numerical</i>	<i>Analytical</i>	% error	<i>Numerical</i>	<i>Analytical</i>	% error
1	187.38	188.38	0.51	153.72	152.65	-0.70	121.52	119.87	-1.42
2	176.28	175.76	-0.29	139.79	138.36	-1.03	109.78	108.21	-1.24
3	150.04	147.13	-1.97	112.38	110.63	-1.57	87.33	85.96	-1.59
4	103.69	99.50	-4.20	73.09	71.56	-2.13	56.20	55.25	-1.71
5	37.51	35.38	-6.02	25.38	24.77	-2.46	19.39	19.05	-1.78

**Figure 8.5** Comparison of numerical results with the analytical solution (implicit method)



In Figure 8.6 we give the solution at  $t = 40$  s obtained using the implicit and explicit method with a time step of 8 s along with the analytical solution. Whereas the explicit method gives unrealistic oscillations at this step size, the implicit method gives results that are in reasonable agreement with the exact solution. This clearly illustrates a key advantage of the implicit method, which tolerates much larger time steps. However, we stress that good solution accuracy can, of course, only be achieved with small time steps.

**Figure 8.6** Comparison of implicit and explicit solutions for  $\Delta t = 8$  s

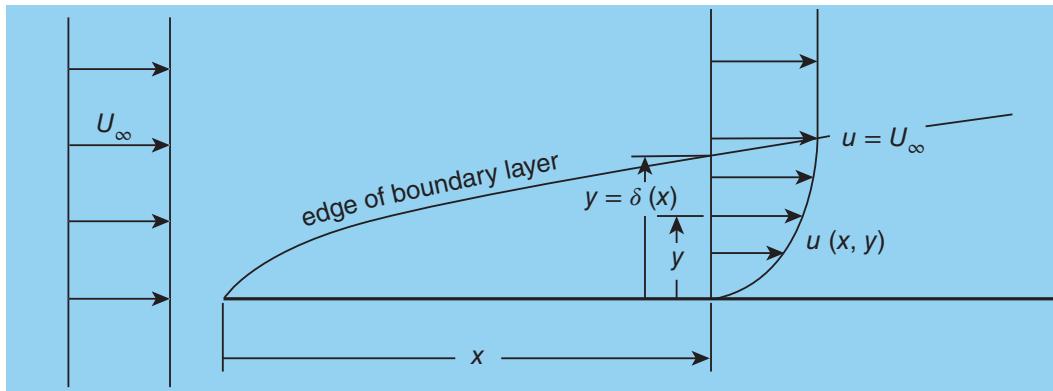


This chapter is concerned with laminar boundary layers, by which we mean the near-surface region for developing laminar flow of a viscous fluid over a solid surface<sup>141</sup>. We introduce the simplifying assumptions for thin boundary layers, which reduce the Navier-Stokes equations to the so-called **boundary-layer equations**. In the case of a **zero-pressure-gradient (flat-plate) boundary layer**, the velocity profiles exhibit self-similarity, which allows the partial differential boundary-layer equations to be reduced to a single ordinary differential equation known as **Blasius' equation**. It is shown that so-called **wedge-flow** boundary layers, where  $U_\infty \propto x^m$ , also exhibit self-similarity and lead to the **Falkner-Skan equation**. Because exact solution of any of these equations is possible only numerically, a number of approximate procedures, involving an integrated form of the boundary-layer equations known as **von Kármán's momentum-integral equation**, have been developed which allow useful information to be obtained with relatively little effort. These procedures include methods in which a simple form, such as a polynomial function, is assumed for the velocity profile. When substituted into the momentum-integral equation, **Pohlhausen's quartic velocity profile** leads to an ordinary differential equation, incorporating a pressure-gradient parameter, which can be used to calculate the development of any laminar boundary layer. Great simplification without major loss of accuracy results when the full differential equation is replaced by a linear correlation based upon the similarity solutions and other exact calculations.

## 17.1 Introductory remarks

In Chapter 16 we analysed a variety of internal laminar flows which were strongly influenced by fluid viscosity and the no-slip boundary condition, but where there were no changes in velocity in either the streamwise or the azimuthal direction. In the case of flow through a long duct this meant that beyond a certain location there were no velocity changes in the axial direction, or  $x$ -direction, and such flows were said to be **fully developed**. This chapter is also concerned with flows strongly affected by viscosity and the no-slip boundary condition but which develop in the streamwise direction, or  $x$ -direction, starting from a uniform approach velocity,  $U_\infty$ . The situation for a uniform flow approaching a stationary thin flat plate aligned with the approach flow is shown schematically in Figure 17.1 (only the flow above the surface is shown). For convenience, a Cartesian-coordinate system has been adopted and it is assumed

<sup>141</sup> A boundary layer can also develop over a porous surface or a liquid surface, but such flows will not be considered in this book.



**Figure 17.1** Viscous flow over an aligned flat plate

that there is no flow in the  $z$ -direction, a restriction that will apply throughout this and the following chapter. Such flows are said to be **two dimensional**. We shall also restrict ourselves to the study of steady, constant-property, flows in the absence of body forces.

A number of features are included in Figure 17.1 which will become increasingly familiar as this chapter develops

- at the surface ( $y = 0$ ), as a consequence of the **no-slip condition**, the streamwise velocity component  $u = u_S = 0$
- changes in the streamwise velocity component  $u$  occur both in the streamwise direction, or  $x$ -direction, and in the normal direction, or  $y$ -direction, i.e.  $u = u(x, y)$
- with increasing normal distance  $y$  from the surface, the streamwise velocity component  $u$  tends asymptotically to the approach velocity  $U_\infty$
- there is a relatively thin viscosity-affected region  $y < \delta$ , where  $u < U_\infty$ , referred to as the boundary layer, which ‘grows’ in thickness with streamwise distance  $x$ , i.e.  $\delta = \delta(x)$
- for values of the normal distance  $y$  greater than  $\delta$  it is assumed that  $u = U_\infty$
- although it cannot be defined precisely, the length  $\delta$  is referred to as the **boundary-layer thickness**
- the region  $y > \delta$  where  $u = U_\infty$  is referred to as the **free stream**<sup>142</sup> in which it is assumed that all velocity gradients with respect to  $y$  are zero, i.e.  $\partial^n u / \partial y^n = 0$ ,  $n = 1, 2, 3 \dots$
- within the boundary layer (i.e.  $y < \delta$ ) the normal component of velocity  $v \ll u$  (for flow over a solid surface,  $v = v_S = 0$  at  $y = 0$ )
- consistent with the previous statement is that streamlines within the boundary layer are almost parallel to the solid surface
- there is negligible change in static pressure  $p$  across the boundary layer, i.e.  $\partial p / \partial y \approx 0$

Some of the characteristics of flat-plate-boundary-layer flow we have just listed are identical with, or closely related to, those of fully-developed flow. With minor modification, these characteristics are typical of all boundary layers. The concept of a boundary layer, the crucial idea that explained many features of real (i.e. viscous) flows, was introduced by Ludwig Prandtl in 1904.

<sup>142</sup> The term **mainstream** is also used.

## 17.2 Two-dimensional laminar boundary-layer equations

Our starting point is the dimensional form of the continuity and Navier-Stokes equations in rectangular-Cartesian coordinates, derived in Section 15.1, assuming steady, constant-property, two-dimensional flow with zero body forces:

continuity

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (17.1)$$

*x*-component

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (17.2)$$

*y*-component

$$u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right). \quad (17.3)$$

In spite of considerable simplification compared with the full Navier-Stokes equations, these equations still represent a major challenge, even to numerical solution. We now introduce the key **boundary-layer approximations** which further simplify the equations to be solved

- there is negligible change in the static pressure  $p$  across the boundary layer, i.e.  $\partial p / \partial y = 0$
- streamwise gradients of  $u$  are much less than gradients with respect to the normal distance  $y$ , in particular  $\partial^2 u / \partial x^2 \ll \partial^2 u / \partial y^2$

The continuity equation is unchanged, but the static pressure  $p$  is now dependent only upon  $x$ , and equation (17.2) reduces to

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{dp}{dx} + \nu \frac{\partial^2 u}{\partial y^2}. \quad (17.4)$$

The essential difficulty of the non-linearity of the terms on the left-hand side of the partial differential equation (17.4) remains. These terms are identically zero for the fully-developed flows of Chapter 16. However, as we shall see in Sections 17.3 and 17.4, under some circumstances equations (17.1) and (17.4) can be combined and reduced to a single ordinary differential equation which can be solved numerically with relatively little effort<sup>143</sup>. As we shall also see, in Sections 17.5 and 17.6, a great deal of insight can be obtained from quite simple approximate solution methods.

For  $y > \delta$ ,  $\partial u / \partial y \rightarrow 0$ ,  $\partial^2 u / \partial y^2 \rightarrow 0$ , and  $u \rightarrow U_\infty(x)$ , the free-stream velocity, so that equation (17.4) reduces to

$$U_\infty \frac{dU_\infty}{dx} = -\frac{1}{\rho} \frac{dp}{dx}, \quad (17.5)$$

<sup>143</sup> Shortly after their formulation in the early 20<sup>th</sup> century such numerical calculations were carried out by hand, requiring considerable effort.

i.e. the differential form of **Bernoulli's equation**. It is usually the case in practice that either  $p(x)$  or  $U_\infty(x)$  is specified. If  $dp/dx$  is zero, the flow is usually described as a **flat-plate boundary layer** even though the plate is not necessarily flat<sup>144</sup>. The term **zero-pressure-gradient boundary layer** is also used. If  $dp/dx < 0$ , the outer flow accelerates, and as we shall see the surface shear stress  $\tau_S$  increases: the **pressure gradient** is said to be **favourable**. The opposite is true for  $dp/dx > 0$ , termed an **adverse pressure gradient**, and **flow separation**, where the surface shear stress  $\tau_S \rightarrow 0$ , is possible.

The question should be asked 'under what conditions are the boundary-layer approximations valid?' To answer this we introduce the following non-dimensional variables:  $x^* = x/L$ ,  $y^* = y\sqrt{Re}/L$ ,  $u^* = u/U_0$ ,  $v^* = v\sqrt{Re}/U_0$ , and  $p^* = p/\rho U_0^2$ , with  $L$  being a length scale characteristic of the streamwise flow direction, such as the length of the surface over which the boundary layer develops, and  $U_0$  being a velocity scale, such as the velocity far upstream of the region influenced by viscosity. The reason for incorporating a **Reynolds number**,  $Re = U_0 L / \nu$ , in the definitions of  $v^*$  and  $y^*$  will become apparent shortly. In non-dimensional form, equations (17.1) to (17.3) may be written as:

continuity

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \quad (17.6)$$

$x$ -component

$$u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = - \frac{\partial p^*}{\partial x^*} + \frac{\partial^2 u^*}{\partial y^{*2}} + \frac{1}{Re} \frac{\partial^2 u^*}{\partial x^{*2}} \quad (17.7)$$

$y$ -component

$$\frac{1}{Re} \left( u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} \right) = - \frac{\partial p^*}{\partial y^*} + \frac{1}{Re^2} \frac{\partial^2 v^*}{\partial y^{*2}} + \frac{1}{Re} \frac{\partial^2 v^*}{\partial x^{*2}} \quad (17.8)$$

with boundary conditions:  $y^* = 0$ ,  $u^* = 0$ ,  $v^* = 0$ , and  $y^* \rightarrow \infty$ ,  $u^* \rightarrow U_\infty^*(x)$ . We see that, if  $Re \rightarrow \infty$ , equation (17.8) reduces to

$$0 = - \frac{\partial p^*}{\partial y^*}, \quad (17.9)$$

and equation (17.7) to

$$u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = - \frac{\partial p^*}{\partial x^*} + \frac{\partial^2 u^*}{\partial y^{*2}}, \quad (17.10)$$

while equation (17.6) is unchanged.

If we now revert to dimensional variables we again arrive at equations (17.1) and (17.4), with equation (17.9) justifying the change from  $\partial p/\partial x$  to  $dp/dx$ . The answer to our question, evidently, is that we are concerned with flows for which the Reynolds number is high. Typically, this means  $Re > 10^3$  but it turns out that as  $Re$  approaches  $10^6$  laminar boundary layers become unstable and eventually turbulent, just as for pipe and channel flows<sup>145</sup>.

<sup>144</sup> The boundary-layer equations can be extended to strongly curved surfaces, but that is beyond the scope of this book.

<sup>145</sup> Note that, whereas for duct flows it was appropriate to define the Reynolds number in terms of the hydraulic diameter, here we are using a streamwise length scale.

It is also informative to make estimates of the orders of magnitude of each of the velocity-gradient terms in equations (17.1) to (17.3). We shall again select  $U_0$  and  $L$  as the velocity and length scales, respectively, for the streamwise direction, but  $V$  and  $\delta$  for the transverse direction, with the assumption that  $\delta \ll L$ . So far as the continuity equation is concerned, we have

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (17.1)$$

with orders of magnitude<sup>146</sup>

$$\frac{\partial u}{\partial x} = O\left(\frac{U_0}{L}\right), \quad \frac{\partial v}{\partial y} = O\left(\frac{V}{\delta}\right).$$

Since there are only two terms in the continuity equation, they must be not only of the same order of, but equal in, magnitude<sup>147</sup> so that

$$V = \frac{U_0 \delta}{L},$$

a result we shall use to replace  $V$  from now on.

For the  $x$ -direction we have

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (17.2)$$

with orders of magnitude

$$u \frac{\partial u}{\partial x} = O\left(\frac{U_0^2}{L}\right), \quad v \frac{\partial u}{\partial y} = O\left(\frac{V U_0}{\delta}\right) = O\left(\frac{U_0^2}{L}\right),$$

$$\nu \frac{\partial^2 u}{\partial x^2} = O\left(\frac{\nu U_0}{L^2}\right), \quad \nu \frac{\partial^2 u}{\partial y^2} = O\left(\frac{\nu U_0}{\delta^2}\right),$$

from which we can conclude that, since  $\delta \ll L$ , it must be that  $\partial^2 u / \partial x^2 \ll \partial^2 u / \partial y^2$ , just as we postulated earlier in this section, and we can neglect  $\partial^2 u / \partial x^2$ . Assuming all remaining velocity terms are of equal magnitude, we conclude that

$$\frac{\nu U_0}{\delta^2} = \frac{U_0^2}{L} \quad \text{or} \quad \frac{\delta}{L} = \sqrt{\frac{\nu}{U_0 L}} = \frac{1}{\sqrt{Re}},$$

i.e. the assumption that  $\delta \ll L$  is consistent with saying  $Re \gg 1$ . Note too that our conclusion that  $\delta/L = 1/\sqrt{Re}$  reveals why it was appropriate to define  $y^*$  and  $v^*$  in terms of  $\sqrt{Re}$ .

For the  $y$ -direction we have

$$u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (17.3)$$

<sup>146</sup>  $O(X)$  is used to mean of order of magnitude  $X$ .

<sup>147</sup> Note that signs are not relevant to orders of magnitude estimates.

with orders of magnitude

$$u \frac{\partial v}{\partial x} = O\left(\frac{U_0 V}{L}\right) = O\left(\frac{U_0^2 \delta}{L^2}\right), \quad v \frac{\partial v}{\partial y} = O\left(\frac{V^2}{\delta}\right) = O\left(\frac{U_0^2 \delta}{L^2}\right),$$

$$v \frac{\partial^2 v}{\partial x^2} = O\left(\frac{v V}{L^2}\right) = O\left(\frac{v U_0 \delta}{L^3}\right), \quad v \frac{\partial^2 v}{\partial y^2} = O\left(\frac{v V}{\delta^2}\right) = O\left(\frac{v U_0}{\delta L}\right).$$

Clearly, the first three velocity terms are negligible compared with the fourth term and can be neglected from now on.

So far we have avoided any statement about the pressure gradients in the  $x$ - and  $y$ -directions. From equation (17.2) for the  $x$ -direction it must be that

$$\frac{1}{\rho} \frac{\partial p}{\partial x} = O\left(\frac{U_0^2}{L}\right)$$

while from equation (17.3), for the  $y$ -direction,

$$\frac{1}{\rho} \frac{\partial p}{\partial y} = O\left(\frac{v U_0}{\delta L}\right) = O\left(\frac{1}{\sqrt{Re}} \frac{U_0^2}{L}\right)$$

and we conclude that  $\partial p / \partial y \ll \partial p / \partial x$ , again consistent with our original postulate.

Equations (17.1) and (17.4) constitute what are referred to as the constant-property, two-dimensional boundary-layer equations with surface boundary conditions  $y = 0, u = u_S = 0$ , for a stationary surface, and  $y = 0, v = v_S = 0$ , for an impermeable surface. If the surface has velocity  $u_S$  in the  $x$ -direction, and **mass transfer** through a porous surface with velocity  $v_S$  in the  $y$ -direction, then the surface boundary conditions become  $y = 0, u = u_S, v = v_S$ . **Transpiration cooling** is the term used when cooler fluid is **blown** through a porous surface into hotter boundary-layer fluid, i.e.  $v_S > 0$ . **Boundary-layer control**, for example to prevent or delay **boundary-layer separation**, can be achieved by **suction**, when  $v_S < 0$ . For  $y \rightarrow \infty$ , the boundary conditions are  $u \rightarrow U_\infty$ , and  $\partial^n u / \partial y^n \rightarrow 0$ .

Since the boundary-layer approximations lead to  $\partial u / \partial y = O(U_0 / \delta)$ , and  $\partial u / \partial x = O(U_0 / L)$ , it should be apparent that a consequence is that the shear stress  $\tau$  at any location  $x$  within a boundary layer is given by

$$\tau = \mu \frac{\partial u}{\partial y}. \tag{17.11}$$

A number of commercial software packages are available which can be used for the numerical integration of the laminar boundary-layer equations, and such calculations are now regarded as routine. The purpose of this chapter is to give the reader some insight into the properties and behaviour of laminar boundary layers, based upon exact solutions of the equations for a flat-plate boundary layer and for boundary layers where the free-stream velocity is proportional to  $x^m$ , or from more general approximate solutions. The accuracy of such solutions will be adequate for many engineering applications and often provide useful confirmation that a numerical solution has not been compromised by the input of faulty data or an error in the computer program.

For two- and three-dimensional problems, the TDMA must be applied iteratively in a line-by-line fashion, but the spread of boundary information into the calculation domain can be slow. In CFD calculations the convergence rate depends on the sweep direction, with sweeping from upstream to downstream along the flow direction producing faster convergence than sweeping against the flow or parallel to the flow direction. Convergence problems can be alleviated by alternating the sweep directions, which is particularly useful in complex three-dimensional recirculating flows, where the dominant flow direction is not known in advance. When overall stability considerations require coupling between the values over the whole calculation domain the TDMA can be unsatisfactory for the solution of discretised equations.

Higher-order schemes for the discretisation process will link each discretisation equation to nodes other than the immediate neighbours. Here, the TDMA can only be applied by incorporating several neighbouring contributions in the source term. This may be undesirable in terms of stability, can impair the effectiveness of the higher-order scheme, and may hinder the implicit nature of the scheme if it is applied in an unsteady flow (see Chapter 8). In the specific case where the system of equations to be solved has the form of a penta-diagonal matrix, as may be the case in QUICK and other higher-order discretisation schemes, there is an alternative solution: a generalised version of the TDMA, known as the penta-diagonal matrix algorithm, is available. Basically a sequence of operations is carried out on the original matrix to reduce it to upper triangular form, and back-substitution is performed to obtain the solution. Details of the method can be found in Fletcher (1991). The method is, however, not nearly as economical as the TDMA.

## 7.6

### Point-iterative methods

Point-iterative techniques are introduced by means of a simple example. Consider a set of three equations and three unknowns:

$$\begin{aligned} 2x_1 + x_2 + x_3 &= 7 \\ -x_1 + 3x_2 - x_3 &= 2 \\ x_1 - x_2 + 2x_3 &= 5 \end{aligned} \quad (7.14)$$

In iterative methods we rearrange the first equation to place  $x_1$  on the left hand side, the second equation to get  $x_2$  on the left hand side, and so on. This yields

$$\begin{aligned} x_1 &= (7 - x_2 - x_3)/2 \\ x_2 &= (2 + x_1 + x_3)/3 \\ x_3 &= (5 - x_1 + x_2)/2 \end{aligned} \quad (7.15)$$

We see that unknowns  $x_1$ ,  $x_2$  and  $x_3$  appear on both sides of (7.15). The system of equations can be solved iteratively by substituting a set of guessed initial values for  $x_1$ ,  $x_2$  and  $x_3$  on the right hand side. This allows us to calculate new values of the unknowns on the left hand side of (7.15). The next move is to substitute the new values back into the right hand side and evaluate the unknowns on the left hand side again, which are, if the procedure converges, closer to the true solution of the system of equations. This process is repeated until there is no more change in the solution.

One condition for the iteration process to be convergent is that the matrix must be diagonally dominant (see discussion on boundedness in

section 5.4.2). When general systems of equations are solved it is sometimes necessary to rearrange the equations, but the finite volume method yields diagonally dominant systems as part of the discretisation process, so this aspect does not require special attention.

The Jacobi and Gauss–Seidel methods apply slightly different substitutions on the right hand side. Below we describe the main features both methods.

### 7.6.1 Jacobi iteration method

In the Jacobi method, the values  $x_1^{(k)}, x_2^{(k)}$  etc. on the left hand side at iteration  $(k)$  – indicated here by the bracketed superscript – are evaluated by substituting in the right hand side the last known values  $x_1^{(k-1)}, x_2^{(k-1)}$  etc., which were obtained at iteration  $(k - 1)$ . In the above example, let us use  $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$  as the initial guess. Substitution of these values in the right hand side of (7.15) gives

$$x_1^{(1)} = 3.500 \quad x_2^{(1)} = 0.667 \quad x_3^{(1)} = 2.500$$

For the second iteration we substitute these values in the right hand side of (7.15). If we repeat the process we obtain the results given in Table 7.10.

**Table 7.10** Solution of system of equations (7.14) with Jacobi method

Iteration number	0	1	2	3	4	5	...	17
$x_1$	0	3.5000	1.9167	1.6250	1.2292	1.1563	...	1.0000
$x_2$	0	0.6667	2.6667	1.6667	2.1667	1.9167	...	2.0000
$x_3$	0	2.5000	1.0833	2.8750	2.5208	2.9688	...	3.0000

After 17 iterations we obtain  $x_1 = 1.0000$ ,  $x_2 = 2.0000$ ,  $x_3 = 3.0000$  and detect no further change in the solution with increase of the iteration count. Substitution of these values into the original system (7.14) shows that this result is accurate to all 4 decimal places given in the answer.

To generalise the procedure we consider a system of  $n$  equations and  $n$  unknowns in matrix form,  $\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$ , or in a form where the coefficients of matrix  $\mathbf{A}$  can be seen explicitly:

$$\sum_{j=1}^n a_{ij} x_j = b_i \quad (7.16)$$

In all iterative methods the system is rearranged to place the contribution due to  $x_i$  on the left hand side of the  $i$ th equation and the other terms on the right hand side:

$$a_{ii} x_i = b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j \quad (i = 1, 2, \dots, n) \quad (7.17)$$

We divide both sides by coefficient  $a_{ii}$  and indicate that, in the Jacobi method, we evaluate the left hand side at iteration  $(k)$  using values on the right hand side of  $x_j$  at the end of the previous iteration  $(k - 1)$ :

$$x_i^{(k)} = \sum_{j=1, j \neq i}^n \left( \frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}} \quad (i = 1, 2, \dots, n) \quad (7.18)$$

Equation (7.18) is the **iteration equation for the Jacobi method** in the form used for actual calculations. In matrix form this equation can be written as follows:

$$\mathbf{x}^{(k)} = \mathbf{T} \cdot \mathbf{x}^{(k-1)} + \mathbf{c} \quad (7.19a)$$

where  $\mathbf{T}$  = iteration matrix  
and  $\mathbf{c}$  = constant vector

The coefficients  $T_{ij}$  of the **iteration matrix** are as follows:

$$T_{ij} = \begin{cases} -\frac{a_{ij}}{a_{ii}} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \quad (7.19b)$$

and the elements of the constant vector are

$$c_i = \frac{b_i}{a_{ii}} \quad (7.19c)$$

### 7.6.2 Gauss–Seidel iteration method

We begin our discussion of the Gauss–Seidel method by reconsidering equation (7.15). In the Jacobi method the right hand side is evaluated using the results of the previous iteration level or from the initial guess. If all the right hand sides could be evaluated simultaneously there would be no further discussion, but in most computing machines the calculations are performed sequentially. Hence, at the first iteration we start the sequence of calculations by using the initial guesses  $x_2^{(0)} = 0$  and  $x_3^{(0)} = 0$  to obtain

$$x_1^{(1)} = (7 - x_2^{(0)} - x_3^{(0)})/2 = (7 - 0 - 0)/2 = 3.5$$

Next we evaluate the second equation,  $x_2 = (2 + x_1 + x_3)/3$ . We notice that it contains  $x_1$  and  $x_3$  on the right hand side. The Jacobi method uses  $x_1^{(0)} = 0$  and  $x_3^{(0)} = 0$  from the initial guesses, but we note that in a sequential calculation we have just obtained an updated value of  $x_1$ , namely  $x_1^{(1)} = 3.5$ . The Gauss–Seidel method proceeds by making direct use of this recently available value and calculates

$$x_2^{(1)} = (2 + x_1^{(1)} + x_3^{(0)})/3 = (2 + 3.5 + 0)/3 = 1.8333$$

To evaluate the third equation,  $x_3 = (5 - x_1 + x_2)/2$ , the Gauss–Seidel method continues to use the most up-to-date values on the right hand side that are available, i.e.  $x_1^{(1)} = 3.5$  and  $x_2^{(1)} = 1.8333$ :

$$x_3^{(1)} = (5 - x_1^{(1)} + x_2^{(1)})/2 = (5 - 3.5 + 1.8333)/2 = 1.6667$$

The second and subsequent iterations follow the same pattern. The results are shown in Table 7.11.

**Table 7.11** Solution of system of equations (7.14) with Gauss–Seidel method

<i>Iteration number</i>	0	1	2	3	4	5	...	13
$x_1$	0	3.5000	1.7500	1.3333	1.1181	1.0475	...	1.0000
$x_2$	0	1.8333	1.8056	1.9537	1.9761	1.9922	...	2.0000
$x_3$	0	1.6667	2.5278	2.8102	2.9290	2.9724	...	3.0000

The final result is obtained after 13 iterations. Ralston and Rabinowitz (1978) note that the Gauss–Seidel method is preferable to the Jacobi method, because it converges faster.

We can easily generalise the above example and state the **iteration equation for the Gauss–Seidel method**:

$$\boxed{x_i^{(k)} = \sum_{j=1}^{i-1} \left( \frac{-a_{ij}}{a_{ii}} \right) x_j^{(k)} + \sum_{j=i+1}^n \left( \frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}}} \quad (i = 1, 2, \dots, n)$$
(7.20)

In matrix form we have

$$\mathbf{x}^{(k)} = \mathbf{T}_1 \mathbf{x}^{(k)} + \mathbf{T}_2 \mathbf{x}^{(k-1)} + \mathbf{c} \quad (7.21a)$$

The coefficients of matrices  $\mathbf{T}_1$  and  $\mathbf{T}_2$  are as follows:

$$T_{1ij} = \begin{cases} -\frac{a_{ij}}{a_{ii}} & \text{if } i > j \\ \frac{a_{ii}}{a_{ii}} & \text{if } i \leq j \\ 0 & \text{if } i \leq j \end{cases} \quad (7.21b)$$

$$T_{2ij} = \begin{cases} 0 & \text{if } i \geq j \\ -\frac{a_{ij}}{a_{ii}} & \text{if } i < j \end{cases} \quad (7.21c)$$

and the elements of the constant vector are as before:

$$c_i = \frac{b_i}{a_{ii}} \quad (7.21d)$$

### 7.6.3 Relaxation methods

The convergence rate of the Jacobi and Gauss–Seidel methods depends on the properties of the iteration matrix. It has been found that these can be improved by the introduction of a so-called relaxation parameter  $\alpha$ . Consider the iteration equation (7.18) for the Jacobi method. It is easy to see that it can also be written as

$$x_i^{(k)} = x_i^{(k-1)} + \sum_{j=1}^n \left( \frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}} \quad (i = 1, 2, \dots, n) \quad (7.22)$$

This chapter begins with a brief description of the qualitative character of turbulent flow. This is followed by the decomposition of a turbulent flow into a mean and fluctuating parts, and the derivation of the Reynolds-averaged Navier-Stokes (RANS) equations, the turbulent kinetic-energy equation, and the equation for the transport of the so-called Reynolds stress, which arises from the time averaging. These equations represent the foundation on which turbulence modelling is based. Reduced forms of the RANS equations for two-dimensional boundary-layer and Couette flows are then derived. The Law of the Wall is shown to result from dimensional considerations applied to plane Couette flow. Three separate zones are identified: the viscous sublayer, the fully-turbulent log-law region, and the buffer layer which separates the two. The Law of the Wake for the outer region of a turbulent boundary layer is then presented. A brief discussion follows concerned with the wide spectrum of length, time, and velocity scales, which are an important aspect of any turbulent flow. The log law is then applied to the analysis of turbulent flow through a smooth-walled pipe, followed by consideration of the effect of surface roughness. The calculation of pressure loss in piping systems is presented, largely based on empirical loss coefficients for such components as elbows, tee junctions, and area changes, together with the frictional pressure drop in long sections of pipe. Both the log-law and power-law velocity distributions are used as a basis for the analysis of a flat-plate turbulent boundary layer, and the results compared with empirical skin-friction formulae. The flow over a circular cylinder in crossflow is discussed based upon the experimentally based curve for the drag coefficient  $C_D$  versus Reynolds number,  $Re$ . Values of  $C_D$  are given for cylinders of various cross section in crossflow and also for a number of three-dimensional objects.

## 18.1 Transitional and turbulent flow

For any given **shear flow**<sup>166</sup>, as the Reynolds number is progressively increased, some regions of the flowfield exhibit an unsteady (i.e. time-dependent) but still relatively orderly, essentially laminar, state, while other zones increasingly show irregular, chaotic, fluctuations in velocity. The latter is called **turbulent flow**, while the intermittent mix of quasi-laminar and turbulent flow is termed **transitional flow**. The **intervallency factor**  $\gamma$  at a fixed point in a flow is the fraction of time the flow there is turbulent, so that  $\gamma$  ranges from 0 to 1. As a consequence of

<sup>166</sup> Any flow in which viscous or turbulent shear stresses play a key role is termed a shear flow. Examples include duct flow, boundary-layer flow, free and wall jets, and wakes.

the velocity differences and overall scales involved, most flows of industrial interest fall into the turbulent-flow category.

It is generally accepted that any flow of a Newtonian fluid, whether laminar, transitional, or turbulent, obeys the **Navier-Stokes equations**. Although turbulent flow is inherently unsteady, a time average can be taken leading to a set of equations for the time-averaged motion. As we show in Section 18.2, these equations closely resemble the laminar-flow equations of Chapter 17 but include additional terms, the so-called **Reynolds stresses**, which arise as a consequence of correlations between the fluctuations in the three velocity components,  $u'$ ,  $v'$ , and  $w'$ . Accounting for these correlations is the central problem in the analysis of time-averaged turbulent flows.

In principle the Navier-Stokes equations for the fluctuating motion can be converted to finite-difference, finite-volume, or finite-element form (a process called **discretisation**) and solved numerically, an approach known as **direct numerical simulation (DNS)**. In the case of a turbulent boundary layer, such a simulation has to account for all length scales, ranging from the Kolmogorov length scale<sup>167</sup>  $l_k$  to the boundary-layer thickness  $\delta$  itself. Since a fluid volume of length  $L$ , width  $W$ , and thickness  $\delta$  contains  $WL\delta/l_k^3$  cells of side length  $l_k$ , this is the number of cells which would have to be considered in a complete simulation. With typical values  $L = W = 1$  m,  $\delta = 50$  mm, and  $l_k = 50$   $\mu\text{m}$ , we find that this corresponds to  $10^{15}$  (or 1000 trillion) cells, which exceeds the capacity of present-day computers and it is unlikely that DNS will be used for routine engineering calculations in the foreseeable future. Computing power, measured in flop/s, has increased exponentially over the last six decades, roughly according to  $\text{flop/s} \approx e^{0.5\Delta y}$ , where  $\Delta y$  is the number of years from 1938<sup>168</sup> at which we have set  $\text{flop/s} = 1$ . The fastest computer as of June 2015 is reported to have achieved close to 34 petaflops, i.e.  $10^{15}$  flop/s.

## 18.2 Reynolds decomposition, Reynolds averaging, and Reynolds stresses

At any point within a turbulent flow, the three orthogonal components of the velocity,  $u$ ,  $v$ , and  $w$ , together with the static pressure  $p$ , fluctuate apparently randomly with time. In general, it would be necessary to include the density in this list, but we shall limit consideration to constant-density flows. These fluctuations are subject to a number of constraints and so cannot be completely random: the continuity equation 15.7 for the flow of a constant-density fluid shows that  $u$ ,  $v$ , and  $w$  are not independent, while  $p$  is related to  $u$ ,  $v$ , and  $w$  through the Navier-Stokes equations 15.29 to 15.31.

Since DNS for practical engineering calculations is at best a prospect for the distant future, it is probable that for the foreseeable future the method of analysing turbulent flow, suggested by Osborne Reynolds in 1895, will be based upon separating all flow quantities into a mean (or time-averaged) part and a fluctuating (time-dependent) part, the latter having an average value

<sup>167</sup> The Kolmogorov length scale, together with other turbulence scales, is discussed in Section 18.4.

<sup>168</sup> The year 1938 is a consequence of extrapolating to zero a linear fit to a graph of computing power (flop/s) versus calendar year on log-linear coordinates.

of zero. This is known as **Reynolds decomposition**. If we take the  $x$ -component of velocity  $u$ , to illustrate this idea, we have

$$u = \bar{u} + u' \quad (18.1)$$

where  $\bar{u}$  is the time average<sup>169</sup> of the fluctuating velocity  $u$  and  $u'$  is the fluctuating part. The time average is defined by

$$\bar{u} = \frac{1}{T} \int_0^T u dt \quad (18.2)$$

where the time interval  $T$  over which the average is taken is long compared with the fluctuating time scale. The overbar<sup>170</sup> here and elsewhere signifies a time average. Substitution of  $u = \bar{u} + u'$  into equation (18.2) leads immediately to  $\bar{u}' = 0$ . As we shall see shortly, when we apply this type of averaging process to the Navier-Stokes equations, non-zero terms like  $\bar{u'^2}$  and  $\bar{u'v'}$  arise.

According to equation (15.7), for an incompressible flow the continuity equation is

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (18.3)$$

which we can now write as

$$\frac{\partial (\bar{u} + u')}{\partial x} + \frac{\partial (\bar{v} + v')}{\partial y} + \frac{\partial (\bar{w} + w')}{\partial z} = 0. \quad (18.4)$$

If we take the time average of equation (18.4), the three fluctuating terms average to zero and we have

$$\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial y} + \frac{\partial \bar{w}}{\partial z} = 0. \quad (18.5)$$

If we subtract equation (18.5) from equation (18.4) we find

$$\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0 \quad (18.6)$$

and we see that the mean and fluctuating parts of the instantaneous velocity components separately satisfy an equation of continuity.

From Subsection 15.1.6 the three components of the Navier-Stokes equations for the flow of a constant- and uniform-property fluid in the absence of body forces are:

$x$ -component

$$\rho \frac{Du}{Dt} = \rho \left[ \frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial (uv)}{\partial y} + \frac{\partial (uw)}{\partial z} \right] = -\frac{\partial p}{\partial x} + \mu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \quad (18.7)$$

<sup>169</sup> To avoid confusion, we shall assume that  $\bar{u}$  is independent of time. However, just as we can have an unsteady laminar flow, we can also have a turbulent flow for which  $\bar{u}$  varies with time at a frequency much lower than is typical of the turbulent fluctuations.

<sup>170</sup> Angle brackets are sometimes used instead of an overbar to denote a time average, i.e.  $u = \bar{u}$ .

*y*-component

$$\rho \frac{Dv}{Dt} = \rho \left[ \frac{\partial v}{\partial t} + \frac{\partial(vu)}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial(vw)}{\partial z} \right] = -\frac{\partial p}{\partial y} + \mu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right) \quad (18.8)$$

*z*-component

$$\rho \frac{Dw}{Dt} = \rho \left[ \frac{\partial w}{\partial t} + \frac{\partial(wu)}{\partial x} + \frac{\partial(wv)}{\partial y} + \frac{\partial w^2}{\partial z} \right] = -\frac{\partial p}{\partial z} + \mu \left( \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right). \quad (18.9)$$

In each of these equations we have replaced terms like  $v\partial u/\partial y$  by  $\partial(uv)/\partial y - u\partial v/\partial y$  and used the continuity equation to eliminate  $\partial v/\partial y$ <sup>171</sup>.

If we now take the time averages of these three equations we have, for a flow which is steady on average

$$\rho \left[ \overline{\frac{\partial u^2}{\partial x}} + \overline{\frac{\partial(uv)}{\partial y}} + \overline{\frac{\partial(uw)}{\partial z}} \right] = -\frac{\partial \bar{p}}{\partial x} + \mu \left( \frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} + \frac{\partial^2 \bar{u}}{\partial z^2} \right) \quad (18.10)$$

$$\rho \left[ \overline{\frac{\partial(vu)}{\partial x}} + \overline{\frac{\partial v^2}{\partial y}} + \overline{\frac{\partial(vw)}{\partial z}} \right] = -\frac{\partial \bar{p}}{\partial y} + \mu \left( \frac{\partial^2 \bar{v}}{\partial x^2} + \frac{\partial^2 \bar{v}}{\partial y^2} + \frac{\partial^2 \bar{v}}{\partial z^2} \right) \quad (18.11)$$

$$\rho \left[ \overline{\frac{\partial(wu)}{\partial x}} + \overline{\frac{\partial(wv)}{\partial y}} + \overline{\frac{\partial w^2}{\partial z}} \right] = -\frac{\partial \bar{p}}{\partial z} + \mu \left( \frac{\partial^2 \bar{w}}{\partial x^2} + \frac{\partial^2 \bar{w}}{\partial y^2} + \frac{\partial^2 \bar{w}}{\partial z^2} \right). \quad (18.12)$$

It should now be clear that the difference between the Navier-Stokes equations for a steady laminar flow and for a time-mean-steady turbulent flow arises from the non-linear advective terms on the left-hand side of each of the last three equations.

If we now introduce  $u = \bar{u} + u'$ ,  $v = \bar{v} + v'$ , and  $w = \bar{w} + w'$ , we have

$$\overline{u^2} = \bar{u}^2 + \overline{u'u'}, \overline{uv} = \bar{u}\bar{v} + \overline{u'v'}, \text{ and } \overline{uw} = \bar{u}\bar{w} + \overline{u'w'}, \text{ etc.} \quad (18.13)$$

In arriving at these identities we have made use of the fact that  $\overline{u'u} = \overline{u'\bar{u}} = 0$ ,  $\overline{v'v} = \overline{v'\bar{v}} = 0$ , etc.

Equation (18.10) may now be written as

$$\rho \left[ \frac{\partial}{\partial x} \left( \bar{u}^2 + \overline{u'u'} \right) + \frac{\partial}{\partial y} \left( \bar{u}\bar{v} + \overline{u'v'} \right) + \frac{\partial}{\partial z} \left( \bar{u}\bar{w} + \overline{u'w'} \right) \right] = -\frac{\partial \bar{p}}{\partial x} + \mu \left( \frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} + \frac{\partial^2 \bar{u}}{\partial z^2} \right)$$

which simplifies to

$$\begin{aligned} \rho \left( \bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} + \bar{w} \frac{\partial \bar{u}}{\partial z} \right) &= -\frac{\partial \bar{p}}{\partial x} + \frac{\partial}{\partial x} \left( \mu \frac{\partial \bar{u}}{\partial x} - [\rho \overline{u'u'}] \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial \bar{u}}{\partial y} - [\rho \overline{u'v'}] \right) \\ &\quad + \frac{\partial}{\partial z} \left( \mu \frac{\partial \bar{u}}{\partial z} - [\rho \overline{u'w'}] \right) \end{aligned} \quad (18.14)$$

equation (18.11) may be written as

$$\rho \left[ \frac{\partial}{\partial x} \left( \bar{v}\bar{u} + \overline{v'u'} \right) + \frac{\partial}{\partial y} \left( \bar{v}^2 + \overline{v'v'} \right) + \frac{\partial}{\partial z} \left( \bar{v}\bar{w} + \overline{v'w'} \right) \right] = -\frac{\partial \bar{p}}{\partial y} + \mu \left( \frac{\partial^2 \bar{v}}{\partial x^2} + \frac{\partial^2 \bar{v}}{\partial y^2} + \frac{\partial^2 \bar{v}}{\partial z^2} \right)$$

<sup>171</sup> This part of the analysis is the subject of Self-assessment problem 18.1.

which simplifies to

$$\begin{aligned} \rho \left( \bar{u} \frac{\partial \bar{v}}{\partial x} + \bar{v} \frac{\partial \bar{v}}{\partial y} + \bar{w} \frac{\partial \bar{v}}{\partial z} \right) = & - \frac{\partial \bar{p}}{\partial y} + \frac{\partial}{\partial x} \left( \mu \frac{\partial \bar{v}}{\partial x} - [\rho \bar{v}' u'] \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial \bar{v}}{\partial y} - [\rho \bar{v}' v'] \right) \\ & + \frac{\partial}{\partial z} \left( \mu \frac{\partial \bar{v}}{\partial z} - [\rho \bar{v}' w'] \right) \end{aligned} \quad (18.15)$$

and equation (18.12) may be written as

$$\rho \left[ \frac{\partial}{\partial x} (\bar{w}\bar{u} + \bar{w}'\bar{u}') + \frac{\partial}{\partial y} (\bar{w}\bar{v} + \bar{w}'\bar{v}') + \frac{\partial}{\partial z} (\bar{w}^2 + \bar{w}'\bar{w}') \right] = - \frac{\partial \bar{p}}{\partial z} + \mu \left( \frac{\partial^2 \bar{w}}{\partial x^2} + \frac{\partial^2 \bar{w}}{\partial y^2} + \frac{\partial^2 \bar{w}}{\partial z^2} \right)$$

which simplifies to

$$\begin{aligned} \rho \left( \bar{u} \frac{\partial \bar{w}}{\partial x} + \bar{v} \frac{\partial \bar{w}}{\partial y} + \bar{w} \frac{\partial \bar{w}}{\partial z} \right) = & - \frac{\partial \bar{p}}{\partial z} + \frac{\partial}{\partial x} \left( \mu \frac{\partial \bar{w}}{\partial x} - [\rho \bar{w}' u'] \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial \bar{w}}{\partial y} - [\rho \bar{w}' v'] \right) \\ & + \frac{\partial}{\partial z} \left( \mu \frac{\partial \bar{w}}{\partial z} - [\rho \bar{w}' w'] \right). \end{aligned} \quad (18.16)$$

Equations (18.14), (18.15), and (18.16) are known as the **Reynolds-averaged Navier-Stokes** (or **RANS**) **equations**. It is the terms in square brackets in these equations which distinguish them from their laminar-flow counterparts. In fact, if the velocity fluctuations are zero, the RANS equations reduce to those for steady, constant-property, laminar flow. From now on we shall write  $\bar{u}'\bar{u}'$ ,  $\bar{v}'\bar{v}'$ , and  $\bar{w}'\bar{w}'$  as  $\bar{u}'^2$ ,  $\bar{v}'^2$ , and  $\bar{w}'^2$ , respectively. These additional terms, usually shifted to appear on the right-hand sides of the time-averaged Navier-Stokes equations, by comparison with the viscous normal and shear stresses, may be interpreted physically as stresses:  $\rho \bar{u}'^2$ ,  $\rho \bar{v}'^2$ , and  $\rho \bar{w}'^2$  as pressure-like normal stresses, and  $\rho \bar{u}'\bar{v}'$ ,  $\rho \bar{v}'\bar{w}'$ , and  $\rho \bar{w}'\bar{u}'$  as shear stresses. The six stresses are known as the **Reynolds stresses**<sup>172</sup> or **apparent stresses**. Each of the shear stresses arises from the correlation of two orthogonal components of the velocity fluctuation at a given point, a non-zero value of the correlation indicating that the two components are not independent. If the correlation is negative, then the two components are opposite in sign over most of the averaging period.

### 18.3 Turbulent-kinetic-energy equation and Reynolds-stress equation

At any point in a turbulent flow, half the sum of the three normal stresses represents the **specific turbulent kinetic energy**<sup>173</sup>  $\bar{k}$ , of the fluctuating velocity components

$$\bar{k} = \frac{1}{2} \left( \bar{u}'^2 + \bar{v}'^2 + \bar{w}'^2 \right). \quad (18.17)$$

Since it is often the case that there is a principal flow direction, for example the  $x$ -direction, it is common to take the value of  $\bar{u}'^2$  as a measure of the turbulence intensity. Other measures

<sup>172</sup> The quantities  $\bar{u}'^2$ ,  $\bar{v}'^2$ ,  $\bar{w}'^2$ ,  $\bar{u}'\bar{v}'$ ,  $\bar{v}'\bar{w}'$ , and  $\bar{w}'\bar{u}'$  are also often referred to as the Reynolds stresses but, in the absence of the density  $\rho$ , the term **kinematic Reynolds stresses** is more appropriate.

<sup>173</sup> The specific turbulent kinetic energy is the turbulent kinetic energy per unit mass.

in use include  $\sqrt{\bar{u}^2}/\bar{u}$ ,  $\sqrt{\bar{u}^2}/U_\infty$ ,  $\sqrt{k}/\bar{u}$ , and  $\sqrt{k}/U_\infty$ . The normalising velocity,  $\bar{u}$  or  $U_\infty$ , is measured at the same location as  $u'$ .

In addition to the RANS equations, further exact equations can be derived from the Navier-Stokes equations. For example, for a steady, two-dimensional, constant-property, turbulent boundary layer, neglecting viscous terms other than the dissipation, we have the **turbulent-kinetic-energy equation**

$$\bar{u} \frac{\partial \bar{k}}{\partial x} + \bar{v} \frac{\partial \bar{k}}{\partial y} = - \frac{\partial}{\partial y} \left[ v' \left( \frac{1}{2} k + \frac{p'}{\rho} \right) \right] + \left( v \frac{\partial \bar{u}}{\partial y} - \bar{u}' \bar{v}' \right) \frac{\partial \bar{u}}{\partial y} - \epsilon \quad (18.18)$$

I                    II                    III                    IV

where  $\epsilon$  is the average **turbulent-kinetic-energy dissipation rate**<sup>174</sup> given by

$$\epsilon = v \left[ 2 \left( \frac{\partial u'}{\partial x} \right)^2 + 2 \left( \frac{\partial v'}{\partial y} \right)^2 + 2 \left( \frac{\partial w'}{\partial z} \right)^2 + \left( \frac{\partial u'}{\partial y} + \frac{\partial v'}{\partial x} \right)^2 + \left( \frac{\partial u'}{\partial z} + \frac{\partial w'}{\partial x} \right)^2 + \left( \frac{\partial v'}{\partial z} + \frac{\partial w'}{\partial y} \right)^2 \right]. \quad (18.19)$$

The combination  $\rho(v \partial \bar{u} / \partial y - \bar{u}' \bar{v}')$  is the combined viscous-plus-turbulent time-average shear stress  $\bar{\tau}$  and, apart from the region close to a solid boundary, it is usually the case that  $-\bar{u}' \bar{v}' \gg v \partial \bar{u} / \partial y$ . In addition to  $\epsilon$ , the turbulent kinetic-energy equation has introduced two further turbulence correlations:  $\bar{v}' \bar{k}$  and  $\bar{v}' p'$ .

The terms in equation (18.18) can be interpreted as follows

- I. Transport of  $k$  through advection by the mean flow
- II. Transport of  $k$  by velocity fluctuations (turbulent diffusion)
- III. Transport of  $k$  by pressure fluctuations
- IV. Rate of production of  $k$  by interaction of the shear stress and the mean-velocity gradient

It should be noted that dissipation of kinetic energy also occurs due to the velocity gradients of the time-mean motion (so-called **direct dissipation**).

Another equation commonly considered in two-dimensional turbulent boundary-layer analysis is the **Reynolds-stress equation**

$$\begin{aligned} \bar{u} \frac{\partial (-\bar{u}' \bar{v}')}{\partial x} + \bar{v} \frac{\partial (-\bar{u}' \bar{v}')}{\partial y} &= \bar{v}^2 \frac{\partial \bar{u}}{\partial y} - \frac{p'}{\rho} \left( \frac{\partial u'}{\partial y} + \frac{\partial v'}{\partial x} \right) + \frac{\partial}{\partial y} \left( \bar{u}' \bar{v}'^2 + \frac{\bar{p}' \bar{u}'}{\rho} \right) \\ &\quad + v \frac{\partial^2 (-\bar{u}' \bar{v}')}{\partial y^2} + 2v \frac{\partial \bar{u}'}{\partial y} \frac{\partial \bar{v}'}{\partial x} \end{aligned} \quad (18.20)$$

which has again introduced further terms.

The turbulent kinetic-energy equation, the Reynolds-stress equation, and other equations derived from or based on the RANS equations are the foundations for the methodology termed turbulence modelling, which we discuss briefly in Section 18.5.

<sup>174</sup> The symbol  $\epsilon$  represents the lower-case Greek letter epsilon. Epsilon is also represented by  $\varepsilon$ , which we use for surface-roughness height (see Section 18.9).

## 18.4 Turbulence scales

So far we have made little mention of the structure of a turbulent flow. It is observed experimentally that clumps (or packets) of fluid particles, called **eddies**, form, interact, break up, and reform throughout a turbulent flow. The length scale (i.e. size) of these eddies varies from the overall scale of the flow down to a **microscale**, much larger than the **molecular mean free path**, so that the **continuum hypothesis** still applies (already implied in assuming that the Navier-Stokes equations apply), where viscosity dominates and turbulent kinetic energy is dissipated into heat. Most of the kinetic energy of a turbulent flow is contained in the **integral length scales**, which are the largest scales in an energy spectrum, i.e. the distribution of kinetic energy according to length scale or frequency. The largest scales correspond to the lowest frequency, and vice versa. The kinetic energy in a turbulent flow passes progressively from the largest energy-bearing eddies to the smallest dissipative eddies in what is termed the **energy cascade**. It should be evident that, because there are fluctuations in velocity and a wide distribution of length scales, **mixing** within a turbulent flow is much stronger than in a laminar flow. The practical consequence is enhanced surface shear stress and, where a surface is heated or cooled, higher rates of heat transfer than in a laminar flow.

The smallest length scale at which turbulence can exist in a flow is the **Kolmogorov length scale**  $l_K$ , defined in terms of the kinematic viscosity of the fluid  $\nu$  and the rate of dissipation of turbulent kinetic energy per unit mass  $\epsilon$

$$l_K = \left( \frac{\nu^3}{\epsilon} \right)^{1/4}. \quad (18.21)$$

This combination of  $\nu$  and  $\epsilon$  is arrived at on dimensional grounds since  $[\nu] = L^2/T$ , and  $[\epsilon] = L^2/T^3$ . The **Kolmogorov time** and **velocity scales**,  $\tau_K$  and  $v_K$ , are similarly defined

$$\tau_K = \left( \frac{\nu}{\epsilon} \right)^{1/2} \quad (18.22)$$

and

$$v_K = (\nu \epsilon)^{1/4}. \quad (18.23)$$

An inevitable consequence of these definitions is that the Reynolds number  $v_K l_K / \nu = 1$ , indicating that the small-scale motion is quite viscous. In numerical simulations of turbulent flows, the smallest scale that has to be resolved is usually taken to be of the same order of magnitude as the Kolmogorov length scale.

At any point in a turbulent flow, fluctuations in velocity and pressure contain energy across a wide range of frequencies  $f$ . According to equation (18.17), the turbulent kinetic energy per unit mass is  $k$

$$\bar{k} = \frac{1}{2} \left( \overline{u'^2} + \overline{v'^2} + \overline{w'^2} \right) \quad (18.17)$$

and, if the kinetic energy in the frequency range  $f$  to  $f + \delta f$  is  $E(f)$ , then

$$\bar{k} = \int_0^\infty E(f) df. \quad (18.24)$$

Rather than frequency, it is usual here to introduce the idea of a **wavenumber**  $\kappa$ , where

$$\kappa = \frac{2\pi f}{v} = \frac{2\pi}{\lambda} \quad (18.25)$$

and  $\lambda = v/f$  is the wavelength,  $v$  being the instantaneous velocity at the point of measurement. Equation (18.24) is then written as

$$\bar{k} = \int_0^\infty E(\kappa) d\kappa \quad (18.26)$$

and the quantity  $E(k)$  is called the **energy spectral density**, or **energy spectrum function**.

The so-called **inertial subrange** corresponds with the non-dissipative **Taylor microscales**, which are intermediate between the largest and smallest scales. Within the inertial subrange Kolmogorov argued that a range of scales exists within which  $E(\kappa)$  is dependent upon the dissipation rate  $\epsilon$  and wavenumber  $\kappa$  but independent of viscosity so that

$$E(\kappa) = F(\epsilon, \kappa). \quad (18.27)$$

It follows from dimensional analysis that

$$E = C_K \epsilon^{2/3} \kappa^{-5/3} \quad (18.28)$$

where  $C_K$  is the **Kolmogorov constant**. The inertial subrange covers the wavenumber range  $1/l \ll \kappa \ll 1/l_K$ , where  $l$  is a measure of the largest scale, such as the **integral length scale**.

If it is assumed, as is reasonable, that the turbulence dissipation rate  $\epsilon$  is determined by the specific turbulent kinetic energy  $k$  and the integral length scale  $l$ , then dimensional analysis leads to

$$\epsilon \sim \frac{k^{3/2}}{l}. \quad (18.29)$$

For Couette flow, the convective terms on the left-hand side of the **turbulent kinetic-energy equation** (18.18) are zero, and  $\bar{u}$  is independent of  $x$ , so we have

$$-\frac{\partial}{\partial y} \left[ v' \left( \frac{1}{2} k + \frac{p'}{\rho} \right) \right] + \left( v \frac{\partial \bar{u}}{\partial y} - \bar{u}' v' \right) \frac{d \bar{u}}{dy} - \epsilon = 0. \quad (18.30)$$

It can be shown that the term involving  $\bar{v}' k$  and  $\bar{v}' p'$  is negligible compared with the production and dissipation terms, so that equation (18.30) reduces to

$$\epsilon = -\bar{u}' v' \frac{d \bar{u}}{dy} \quad (18.31)$$

where we have neglected the viscous shear stress compared with the Reynolds shear stress.

In Subsection 18.7.2 we show that, for the near-wall region, if  $-\rho \bar{u}' v' = \tau_s$ , then

$$\frac{\gamma}{u_\tau} \frac{d \bar{u}}{dy} = \frac{1}{k} \quad (18.50)$$

so that from equation (18.31)

$$\epsilon = \frac{u_\tau^3}{\kappa y} \quad (18.32)$$

and we have a good estimate for  $\epsilon$  in the log-law region.

From the definitions of the Kolmogorov scales (equations (18.21) to (18.23)) we then have

$$\frac{u_\tau l_k}{\nu} = (\kappa y^+)^{1/4} \quad (18.33)$$

$$\frac{u_\tau^2 \tau_K}{\nu} = (\kappa y^+)^{1/2} \quad (18.34)$$

and

$$\frac{v_K}{u_\tau} = \frac{1}{(\kappa y^+)^{1/4}}. \quad (18.35)$$

In Illustrative Example 18.2 we calculate values for the Kolmogorov scales for a specified pipe flow.

## 18.5 Turbulence modelling

Unfortunately, it is unlikely that physically exact equations will ever be established which take into account the entire range of length and time scales which we have just identified and which link the six Reynolds stresses, and correlations such as  $\overline{p' u'}$ ,  $\overline{u' v'^2}$ , etc., to  $\bar{u}$ ,  $\bar{v}$ ,  $\bar{w}$ ,  $\bar{p}$ , and their spatial gradients. Apart from research into DNS and the closely related **large-eddy simulation (LES)**, which avoids the need to establish such links, research hitherto has concentrated on a semi-empirical methodology in which the physics of turbulent flow is approximated by partial differential equations for the specific turbulent kinetic energy  $k$ , the rate of turbulent dissipation  $\epsilon$ , the Reynolds stresses  $\overline{u' v'}$ ,  $\overline{u'^2}$ ,  $\overline{v'^2}$ , etc., and various length scales with approximations for the correlations between  $u'$ ,  $v'$ ,  $w'$ ,  $p'$ , and their gradients. Devising these approximations has become known as **turbulence modelling**.

For the foreseeable future, DNS and LES are likely to remain research topics providing results used to guide the development of, and against which to test the predictions of, turbulence modelling, particularly where experimental data are unavailable. Depending upon the complexity involved and level of accuracy required, engineering applications will rely upon software based upon empirical correlations and turbulence modelling of varying levels of sophistication.

Probably the earliest example of turbulence modelling was Prandtl's **mixing-length hypothesis** based upon the idea that the large-scale random movements of fluid elements in turbulent motion are analogous to the small-scale random motion of molecules in a gas (kinetic theory). From this beginning, with time the following hierarchy of turbulence models has evolved

- zero equation model, such as the mixing-length or eddy-viscosity model
- two-equation model, in which time-averaged equations are solved for the specific turbulent-kinetic energy  $k$  and the turbulent-kinetic-energy dissipation rate  $\epsilon$
- Reynolds-stress equation model
- algebraic-stress model

A detailed presentation and discussion of turbulence modelling is beyond the scope of this text and from the foregoing the impression could be gained that little progress has been made

in devising relatively simple ways to analyse turbulent flows of practical interest. In fact much of what was learned throughout the 20<sup>th</sup> century, based upon simplifications, dimensional analysis, empiricism, and integral methods, forms the basis of current engineering practice and to a large extent is incorporated into turbulence modelling. Some of the ideas involved are discussed in the remainder of this chapter.

## 18.6 Two-dimensional turbulent boundary layers and Couette flow

Although velocity fluctuations in a turbulent flow always occur in the three orthogonal directions, there are many practical situations, just as for laminar flow, where there is no variation of time-averaged quantities in the third direction (usually taken as the  $z$ -direction). If in addition we introduce the boundary-layer approximations outlined in Chapter 17, the counterpart to equation (17.4) for a turbulent boundary layer is

$$\bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} = -\frac{1}{\rho} \frac{dp}{dx} + \frac{1}{\rho} \frac{\partial}{\partial y} \left( \mu \frac{\partial \bar{u}}{\partial y} - \rho \bar{u}' \bar{v}' \right). \quad (18.36)$$

Even though the Reynolds shear stress  $\rho \bar{u}' \bar{v}'$  is the only term remaining from the Reynolds decomposition and time averaging, except in the near-vicinity of a wall, it is usually several orders of magnitude greater than the viscous shear stress,  $\mu \partial \bar{u} / \partial y$ , and so has to be accounted for. Just as for laminar flow, we can consider fully-developed turbulent flow through a cylindrical channel, for which equation (18.36) reduces to

$$0 = -\frac{1}{\rho} \frac{dp}{dx} + \frac{1}{\rho} \frac{\partial}{\partial y} \left( \mu \frac{d\bar{u}}{dy} - \rho \bar{u}' \bar{v}' \right). \quad (18.37)$$

A further simplification occurs for **Couette flow** which, as we saw in Section 16.4, is the fully-developed, shear-driven flow between two parallel surfaces where one is moving tangentially with respect to the other. Within such an idealised flow the shear stress is constant (giving rise to the term **constant-stress layer**) so that, if the flow is turbulent, the mean (i.e. time-averaged) shear stress  $\bar{\tau}$  is given by

$$\bar{\tau} = \mu \frac{d\bar{u}}{dy} - \rho \bar{u}' \bar{v}' = \text{constant} = \bar{\tau}_S \quad (18.38)$$

where  $\bar{\tau}_S$  is the mean wall shear stress and  $y$  is the distance from the stationary surface.

The boundary conditions for equations (18.36), (18.37), and (18.38) are the same as those for laminar flow.

## 18.7 Plane turbulent Couette flow and the Law of the Wall

As we have just seen, for plane turbulent Couette flow the RANS equations reduce to

$$\bar{\tau} = \mu \frac{d\bar{u}}{dy} - \rho \bar{u}' \bar{v}' = \text{constant} = \bar{\tau}_S. \quad (18.39)$$

Equation (18.39) cannot simply be integrated to give the mean-velocity profile  $\bar{u}(y)$  because the dependence of the Reynolds shear stress  $-\overline{u'v'}$  on other flow properties is unknown.

We shall return to equation (18.39) shortly but for the time being we introduce the assumption that  $\bar{u}$  depends upon  $y$ ,  $\bar{\tau}_S$ , and the fluid properties  $\rho$  and  $\mu$ , i.e.

$$\bar{u} = f(y, \bar{\tau}_S, \rho, \mu), \quad (18.40)$$

an assumption which can reasonably be applied to the near-wall region of any turbulent shear flow over a smooth surface, i.e. to boundary layers and channel flows. Dimensional analysis leads to

$$\frac{\bar{u}}{u_\tau} = f\left(\frac{u_\tau y}{\nu}\right) \quad (18.41)$$

where  $u_\tau = \sqrt{\bar{\tau}_S/\rho}$  has the units of velocity and is termed the **friction velocity** (or **wall-friction velocity**)<sup>175</sup>. Equation (18.41), first postulated by Prandtl, is known as the **Law of the Wall**, or **universal velocity distribution**, and written as

$$u^+ = f(y^+) \quad (18.42)$$

where

$$u^+ \equiv \frac{\bar{u}}{u_\tau} \text{ and } y^+ \equiv \frac{u_\tau y}{\nu} \quad (18.43)$$

are the so-called **wall variables**<sup>176</sup>. It is seen that  $y^+$  can be regarded as a turbulence Reynolds number and that  $\nu/u_\tau$  is a **viscous length scale**. The Law of the Wall is usually regarded as comprising three parts: a **viscous sublayer**, a **buffer layer**, and a **fully-turbulent region**.

### 18.7.1 Viscous sublayer

From the boundary conditions at a solid surface it must be that  $u' = 0$  (the no-slip condition) and  $v' = 0$  (impermeable surface) so that in the immediate vicinity of a solid surface both  $u'$  and  $v'$  decrease and  $\mu d\bar{u}/dy \gg \rho \overline{u'v'}$ . Equation (18.39) thus reduces to

$$\frac{d\bar{u}}{dy} = \frac{\bar{\tau}_S}{\mu} \quad (18.44)$$

which integrates to give

$$\bar{u} = \frac{\bar{\tau}_S y}{\mu} \quad (18.45)$$

or, in wall variables,

$$u^+ = y^+. \quad (18.46)$$

The region where equation (18.46) is valid is termed the **viscous** (or **linear**) **sublayer**<sup>177</sup> and taken to have a thickness  $\delta_{SUB}$  given by

<sup>175</sup> The symbol  $u^*$  is also used to represent the friction velocity and spoken as ‘ustar’.

<sup>176</sup>  $u^+$  and  $y^+$  are spoken as ‘uplus’ and ‘yplus’, respectively.

<sup>177</sup> Although the turbulence intensity is small, the flow within the viscous sublayer is not purely laminar, and the term **laminar sublayer** is to be avoided.

$$\frac{u_\tau \delta_{SUB}}{\nu} = \delta_{SUB}^+ = 5. \quad (18.47)$$

### 18.7.2 Fully-turbulent layer and the log law

Beyond the viscous sublayer,  $y^+ > 30$ , say, it is argued that direct viscous effects on the turbulent structure and the influence of viscosity on the mean flow is negligible so that the mean-velocity gradient is dependent only upon  $y$ ,  $\rho$ , and  $\bar{\tau}_S$ , i.e.

$$\frac{d\bar{u}}{dy} = f(y, \rho, \bar{\tau}_S) \quad (18.48)$$

or, if we introduce  $u_\tau$ ,

$$\frac{d\bar{u}}{dy} = f(y, u_\tau). \quad (18.49)$$

The only dimensionally acceptable form of equation (18.49) is

$$\frac{y}{u_\tau} \frac{d\bar{u}}{dy} = \frac{1}{\kappa} \quad (18.50)$$

or

$$y^+ \frac{du^+}{dy^+} = \frac{1}{\kappa} \quad (18.51)$$

where  $\kappa$  is a constant, known as **von Kármán's constant**.

Equation (18.51) can be integrated to give

$$u^+ = \frac{1}{\kappa} \ln y^+ + B \quad (18.52)$$

where  $B$  is also a universal constant. The velocity distribution represented by equation (18.52) is known as the **log law** and has been confirmed experimentally with the values<sup>178</sup>  $\kappa = 0.4$ , and  $B = 5.5$ .

An approximate indication of the sublayer thickness,  $\delta_{SUB}$ , results from determining the value of  $y^+$  at which the sublayer profile, represented by equation (18.46), has the same value of  $u^+$  as given by the log law. The result is

$$\frac{u_\tau \delta_{SUB}}{\nu} = \delta_{SUB}^+ = 11. \quad (18.53)$$

This value is obviously much greater than  $\delta_{SUB}^+ = 5$  given by equation (18.47) and which represents the wall distance at which the velocity distribution begins to depart from  $u^+ = y^+$ .

Although the log law was arrived at primarily using dimensional arguments, it can also be deduced using primitive turbulence modelling. In equation (18.36), the momentum equation for a two-dimensional, turbulent boundary layer, the only additional term, compared with equation (17.4) for a laminar boundary layer, is the Reynolds shear stress  $\bar{\tau}_T = -\rho \bar{u}' \bar{v}'$ . A natural first step in attempting to account for  $-\rho \bar{u}' \bar{v}'$ , first made by Boussinesq in 1877

<sup>178</sup> Slightly different values are sometimes quoted:  $\kappa = 0.41$ , and  $B = 5.0$ , and there is evidence that  $\kappa$  and  $B$  are weakly dependent upon Reynolds number.

(two decades before Reynolds introduced the idea of time averaging), was to assume that this quantity behaved in an analogous way to that for molecular shear, i.e.

$$\bar{\tau}_T = -\rho \bar{u}' \bar{v}' = \mu_T \frac{\partial \bar{u}}{\partial y}, \quad (18.54)$$

which defines the quantity  $\mu_T$  known as the **eddy viscosity**.

As we stated in Section 18.5, the earliest example of turbulence modelling was probably Prandtl's suggestion that, by analogy with the small-scale random motion of molecules in a gas (kinetic theory), the large-scale random movements of fluid elements (i.e. the eddies) in turbulent motion leads to a transverse exchange of momentum. If this exchange occurs over an average distance  $l_M$ , which has become known as the **mixing length**, then

$$\mu_T = \rho l_M^2 \left| \frac{\partial \bar{u}}{\partial y} \right|. \quad (18.55)$$

One way of quantifying Prandtl's idea is to assume that the **root-mean-square values** of  $u'$  and  $v'$  are approximated by

$$\sqrt{\bar{u}'^2} \approx \sqrt{\bar{v}'^2} \approx l_M \frac{\partial \bar{u}}{\partial y} \quad (18.56)$$

so that

$$-\bar{u}' \bar{v}' \approx \sqrt{\bar{u}'^2} \sqrt{\bar{v}'^2} \approx l_M^2 \frac{\partial \bar{u}}{\partial y} \left| \frac{\partial \bar{u}}{\partial y} \right| = \nu_T \left| \frac{\partial \bar{u}}{\partial y} \right|. \quad (18.57)$$

The modulus sign has been introduced to ensure that the Reynolds shear stress and the velocity gradient have the same sign. In practice, it is found that, for turbulent flows in which the mean velocity is asymmetric about a maximum, there is a small region where  $-\bar{u}' \bar{v}'$  and  $\partial \bar{u} / \partial y$  are opposite in sign, but this is of little consequence. By analogy with  $\nu$ , the quantity  $\nu_T$  is termed the **kinematic eddy viscosity**.

In the vicinity of a solid surface over which there is turbulent flow, it is reasonable to assume that  $l_M$  is proportional to  $y$ , i.e.

$$l_M = \kappa y \quad (18.58)$$

where, at this stage,  $\kappa$  is simply a constant although equation (18.60) below shows that it can be identified as **von Kármán's constant**, introduced above.

If we substitute for  $l_M$  from equation (18.58) in equation (18.57), and assume that  $-\rho \bar{u}' \bar{v}' = \bar{\tau}_S$ , then we have

$$\kappa y \frac{\partial \bar{u}}{\partial y} = \sqrt{\frac{\bar{\tau}_S}{\rho}} = u_\tau \quad (18.59)$$

which can be rewritten as

$$y^+ \frac{du^+}{dy^+} = \frac{1}{\kappa} \quad (18.60)$$

which is identical to equation (18.51) and so again leads to the **log-law** velocity distribution, equation (18.52).

### 18.7.3 Buffer layer

There is no simple theory covering the near-wall range  $5 < y^+ < 30$ , which is intermediate between the viscous sublayer and the log-law region, often called the buffer layer. A formula suggested by Spalding (1961), which has equation (18.46) as the asymptote for  $y^+ \rightarrow 0$ , and which asymptotes to equation (18.52) at large  $y^+$  is

$$y^+ = u^+ + e^{-\kappa B} \left[ e^{\kappa u^+} - 1 - \kappa u^+ - \frac{1}{2} (\kappa u^+)^2 - \frac{1}{6} (\kappa u^+)^3 \right]. \quad (18.61)$$

The negative terms within square brackets can be regarded as correction terms, which are subtracted from the log-law equation in the form

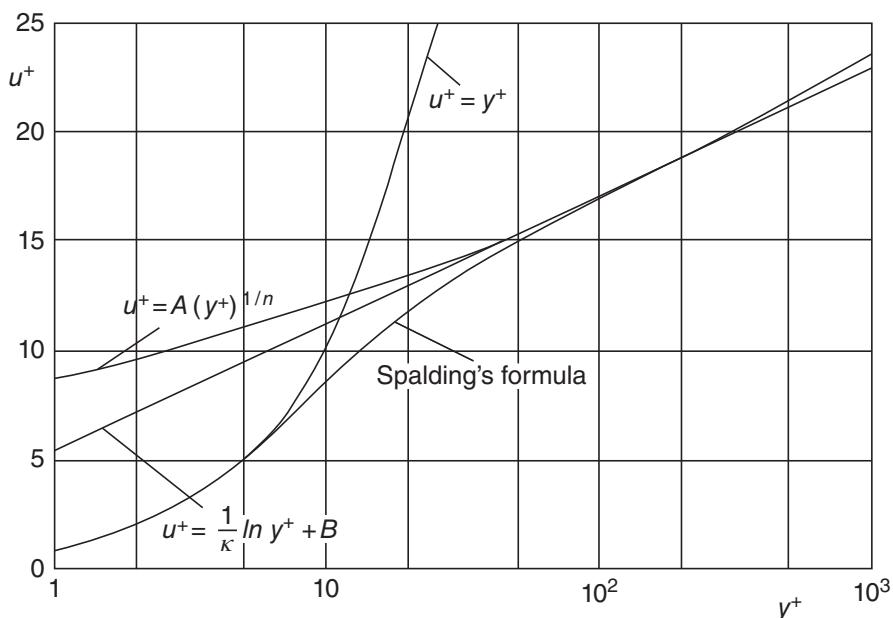
$$y^+ = e^{-\kappa B} e^{\kappa u^+} = e^{-\kappa B} \left[ 1 + \kappa u^+ + \frac{1}{2} (\kappa u^+)^2 + \frac{1}{6} (\kappa u^+)^3 + \frac{1}{24} (\kappa u^+)^4 + \dots \right]. \quad (18.62)$$

Equations (18.46), (18.52), and (18.61) are all plotted on semi-logarithmic coordinates in Figure 18.1, together with a power-law equation with  $A = 8.75$ , and  $m = 7$  (see Subsection 18.13.3). Spalding's formula has been shown to be an accurate fit to measured velocity distributions for pipe-flow data. The discrepancy between the power-law equation and the log law for  $y^+ < 100$  is exaggerated by the logarithmic scale for the abscissa.

Van Driest (1956) suggested that, to account for the viscous sublayer, equation (18.58) for the mixing length should be modified by a so-called **damping factor**

$$l_M = \kappa y \left( 1 - e^{-y^+/C} \right) \quad (18.63)$$

where the empirical constant  $C$  is usually given the value 26 if the streamwise pressure gradient is zero. If equation (18.63) is substituted in equation (18.57), numerical integration, for a constant-stress layer, results in a velocity distribution close to that corresponding



**Figure 18.1** Distributions of mean velocity for near-wall turbulent flow

with Spalding's equation (18.61) but has the advantage that, if  $C$  is appropriately modified, pressure-gradient and other effects, such as transpiration, can be incorporated.

#### 18.7.4 Outer layer and the Law of the Wake

Coles (1956) observed that, as the free stream is approached, experimentally determined velocity profiles within a turbulent boundary layer increasingly depart from the log law, particularly if there is an adverse pressure gradient, i.e. static pressure increasing with streamwise distance along the surface over which the boundary layer is developing. He showed that a good fit to the data is given by a composite in which a so-called **wake function**<sup>179</sup>  $f(\eta)$  is added to the log law

$$u^+ = \frac{1}{\kappa} \ln y^+ + B + \frac{2\Pi}{\kappa} f(\eta) \quad (18.64)$$

where the **wake-strength parameter**  $\Pi$  increases in an adverse pressure gradient. The variable  $\eta = y/\delta$ ,  $\delta$  being the boundary-layer thickness, and the wake function  $f(\eta)$  has a sigmoidal form normalised such that  $f(0) = 0$  and  $f(1) = 1$ . A simple equation which adequately represents the wake function is<sup>180</sup>

$$f(\eta) = 3\eta^2 - 2\eta^3. \quad (18.65)$$

The ratios between  $\delta$  and the integral parameters displacement thickness  $\delta^*$  (equation ((17.44)) and momentum-deficit thickness  $\theta$  (equation (17.61)) were defined in Section 17.3. These ratios, evaluated using equation (18.64) combined with equation (18.65), are given by

$$\frac{\delta^*}{\delta} = \frac{(1 + \Pi)}{\kappa} \sqrt{\frac{c_f}{2}} \quad (18.66)$$

and

$$\frac{\theta}{\delta} = \frac{\delta^*}{\delta} - \frac{F(\Pi)}{\kappa^2} \frac{c_f}{2} \quad (18.67)$$

where

$$F(\Pi) = \frac{52}{35} \Pi^2 + \frac{19}{6} \Pi + 2 \quad (18.68)$$

and  $c_f$  is the skin-friction coefficient defined by

$$c_f = \frac{\overline{\tau}_S}{\frac{1}{2} \rho U_\infty^2}. \quad (18.69)$$

The shape factor  $H$  then follows as

$$\frac{1}{H} = \frac{\theta}{\delta^*} = 1 - \frac{F(\Pi)}{(1 + \Pi) \kappa} \sqrt{\frac{c_f}{2}} \quad (18.70)$$

<sup>179</sup> Coles used the term 'wake' because the shape of the function  $f(\eta)$  resembles the velocity-defect distribution in a turbulent wake.

<sup>180</sup> Another equation for the wake function which is a good fit to the data is  $f(\eta) = \sin^2(\pi \eta/2)$  but the cubic form has the advantage that the algebra involved in determining such quantities as  $\delta^*$  and  $\theta$  is appreciably simpler.

which confirms, as was the case for a laminar boundary layer, that  $H > 1$ . What is also suggested by equation (18.70) is that, given the dependence on the skin-friction coefficient,  $H$  will be Reynolds-number dependent.

Another important result, independent of the form of the wake function, is a consequence of evaluating equation (18.64) at the edge of the boundary layer,  $y = \delta$ , where  $\bar{u} = U_\infty$

$$\sqrt{\frac{2}{c_f}} = \frac{1}{\kappa} \ln \left( \frac{U_\infty \delta}{v} \sqrt{\frac{c_f}{2}} \right) + B + \frac{2\pi}{\kappa} \quad (18.71)$$

wherein we have made use of the identities

$$U_\infty^+ = \sqrt{\frac{2}{c_f}} \quad \text{and} \quad \delta^+ = \frac{u_\tau \delta}{v} = \frac{U_\infty \delta}{v} \sqrt{\frac{c_f}{2}}. \quad (18.72)$$

For boundary layers subjected to an adverse pressure gradient, experimental measurements increasingly depart from the log law as the pressure-gradient parameter<sup>181</sup>

$$\lambda = \frac{\delta}{\bar{u}_S} \frac{dp}{dx} \quad (18.73)$$

is increased ( $\lambda > 10$  corresponds with a strong adverse pressure gradient). For weak favourable pressure gradients ( $\lambda < 0$ ) the wake strength is low so that the log-law equation, equation (18.52), applies throughout the near-wall region, i.e. for a boundary layer this means for  $y \leq \delta$ , while for pipe flow  $y \leq R$ .

### ILLUSTRATIVE EXAMPLE 18.1

Use equation (18.64) with the wake function given by equation (18.65) to show that the ratio of the displacement thickness  $\delta^*$  to the boundary-layer thickness  $\delta$  for a turbulent boundary layer is given by

$$\frac{\delta^*}{\delta} = \frac{(1 + \Pi)}{\kappa} \sqrt{\frac{c_f}{2}}$$

where  $c_f/2$  is the local skin-friction coefficient and  $\Pi$  is the wake-strength parameter.

#### Solution

The definition of the displacement thickness is

$$\delta^* = \int_0^\delta \left( 1 - \frac{\bar{u}}{U_\infty} \right) dy.$$

The mean velocity is approximated by

$$u^+ = \frac{1}{\kappa} \ln y^+ + B + \frac{2\Pi}{\kappa} f(\eta)$$

<sup>181</sup> Note that the pressure gradient  $\lambda$  defined here is different from those defined for both Poiseuille flow  $\lambda_P$ , equation (16.52), and a laminar boundary layer, equation (17.97).

where the wake function is given by

$$f(\eta) = 3\eta^2 - 2\eta^3.$$

If we substitute for  $\bar{u}$  in the definition of  $\delta^*$  we have

$$\begin{aligned} \delta^* &= \delta - \frac{\nu}{u_\tau U_\infty^+} \int_0^{\delta^+} \left( \frac{1}{\kappa} \ln y^+ + B \right) dy^+ - \frac{2\Pi\delta}{\kappa U_\infty^+} \int_0^1 (3\eta^2 - 2\eta^3) d\eta \\ &= \delta - \frac{\nu}{u_\tau U_\infty^+} \left( \frac{\delta^+}{\kappa} \ln \delta^+ - \frac{\delta^+}{\kappa} + B\delta^+ \right) - \frac{\Pi\delta}{\kappa U_\infty^+} \\ &= \frac{(\Pi + 1)\delta}{\kappa U_\infty^+} = \frac{(\Pi + 1)\delta}{\kappa} \sqrt{\frac{c_f}{2}}. \end{aligned}$$


---

## 18.8 Fully-developed turbulent flow through a smooth circular pipe

For fully-developed turbulent flow through a smooth circular pipe, velocity-profile measurements show that the wake strength  $\Pi$  is small so that equation (18.64) with  $\Pi = 0$ , i.e. the log-law equation, equation (18.52), is a good approximation to the mean-velocity distribution

$$u^+ = \frac{1}{\kappa} \ln y^+ + B.$$

We can use this equation to calculate a **bulk-average** (or **spatial average**) velocity  $\bar{V}$  for fully-developed flow through a pipe of radius  $R$  (diameter  $D$ ) through

$$\dot{Q} = \pi R^2 \bar{V} = \int_0^R \bar{u} 2\pi r dr \quad (18.74)$$

where  $\dot{Q}$  is the volumetric flowrate and  $r$  is the radial distance from the pipe centreline. It should be noted that we have neglected not only the wake component of the velocity distribution but also the contribution of the viscous sublayer. The latter approximation is increasingly valid as the pipe Reynolds number increases (see Self-assessment problem 18.2).

Since  $r = R - y$ , where  $y$  is the distance from the pipe wall, equation (18.74) leads to

$$\frac{1}{2} \bar{V} R^2 = R \int_0^R \bar{u} dy - \int_0^R \bar{u} y dy$$

which can be transformed into

$$\frac{u_\tau \bar{V} R^2}{2\nu^2} = R^+ \int_0^{R^+} u^+ dy^+ - \int_0^{R^+} u^+ y^+ dy^+. \quad (18.75)$$

If we substitute for  $u^+$  from equation (18.52), we find, after simplification,

$$\bar{V}^+ = \frac{1}{\kappa} \ln R^+ + B - \frac{3}{2\kappa} \quad (18.76)$$

where  $\bar{V}^+ = \bar{V}/u_\tau$ .

### 3 Wall Functions

In the second order finite volume method, the flow variables (temperature, velocity, pressure etc.) vary linearly across the cell. Hence, when there are steep gradients in the flow variables, many cells are required to resolve the profiles with sufficient accuracy. As shown in Figure 21, the gradients are particularly steep close to the wall. This is due to the boundary layer that develops on a solid surface in contact with the flow. In order to resolve the steep gradients near the wall accurately, many thin cells are required normal to the wall. The gradients across the cell that is immediately adjacent to the wall are particularly important, as the gradients across this cell determine the wall shear stress and the wall heat flux. In general, the wall shear stress  $\tau_w$  and the wall heat flux  $q_w$  are given by Newton's Law of Viscosity and Fourier's Law of Heat Conduction:

$$\frac{\tau_w}{\rho} = -\nu \left. \frac{\partial U}{\partial y} \right|_{y=0} \quad \frac{q_w}{\rho c_p} = -\alpha \left. \frac{\partial T}{\partial y} \right|_{y=0} \quad (139)$$

where  $\rho$  is the fluid density,  $\nu$  is the kinematic viscosity of the fluid,  $\alpha$  is the thermal diffusivity of the fluid,  $U$  is the velocity component parallel with the wall,  $T$  is the temperature and  $y$  is the direction normal to the wall. As the profile is non-linear (the gradient changes with distance from the wall), the gradient is evaluated at the wall ( $y = 0$ ). The minus sign is required because the wall shear stress acts in the opposite direction to the velocity profile and heat flux is in the opposite direction to the temperature gradient (high temperature to low temperature).

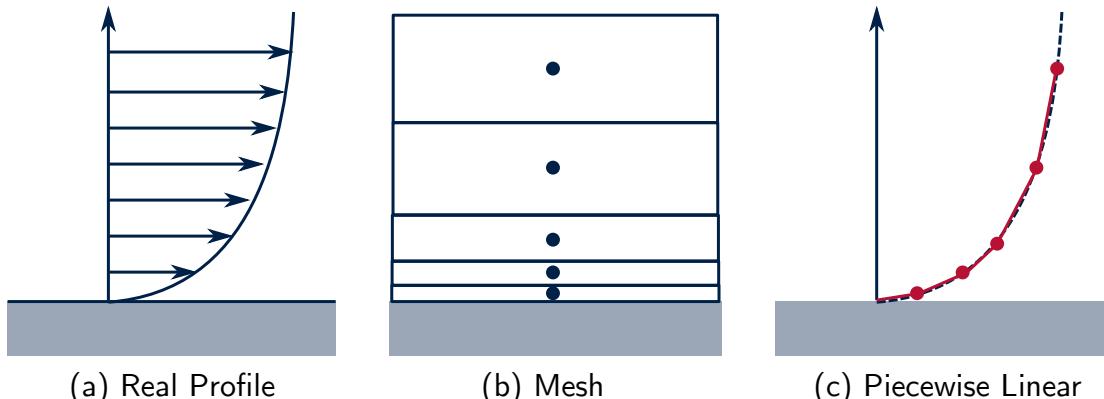
When generating a mesh for CFD simulations, a fine mesh should be used with many thin cells normal to the wall. If the mesh is sufficiently fine, then the piecewise linear variation that is computed by the CFD code has a sufficient number of nodes to accurately reproduce the real flow profile normal to the wall. However, for some flow simulations it is not possible to use many thin cells normal to the wall, as this would lead to a high cell count (and sometimes poor cell quality). For these flow scenarios, the cell adjacent to the wall is too large to accurately reproduce the flow profile. As shown in Figure 22, the velocity and temperature gradients at the wall ( $\partial U / \partial y$  and  $\partial T / \partial y$ ) will be incorrect if the cell is too large. This will lead to inaccuracies in the wall shear stress and wall heat flux (equation 139), unless the CFD code is corrected appropriately. The focus of this Chapter is the method used to correct the flow at the wall when the cells adjacent to the wall are too large.

## Background Theory

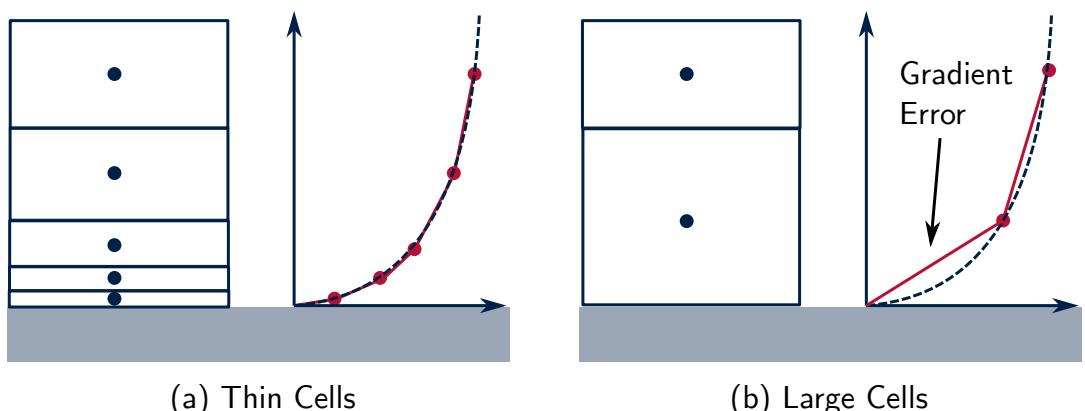
In order to compute the correct wall shear stress, the **product** of the near wall kinematic viscosity and the velocity gradient at the wall must be correct.

$$\frac{\tau_w}{\rho} = -\nu \left. \frac{\partial U}{\partial y} \right|_{y=0} = -\text{Kinematic Viscosity} \times \text{Velocity Gradient at the Wall} \quad (140)$$

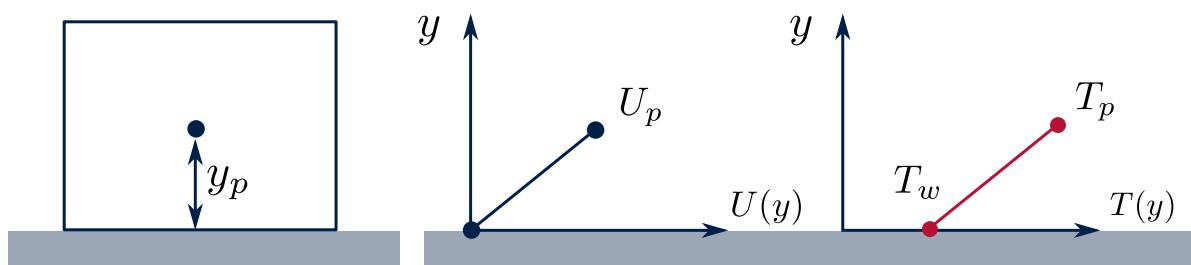
Hence, even though the gradient at the wall ( $\partial U / \partial y$ ) is incorrect, if the near wall kinematic viscosity ( $\nu$ ) is modified appropriately, then the wall shear stress will be correct. This approach of modifying the near wall kinematic viscosity and the near wall thermal diffusivity is used by the majority of modern CFD codes to correct the solution when the cells are too large.



**Figure 21:** A diagram to show the approximation of a real flow profile (velocity or temperature) by a finite volume mesh. The finite volume method uses a piecewise linear approximation of the real profile.



**Figure 22:** A diagram to show the difference in the piecewise linear approximation of the near wall velocity or temperature profile when (a) thin and (b) large cells are used. The gradient at the wall is incorrect when large cells are used.

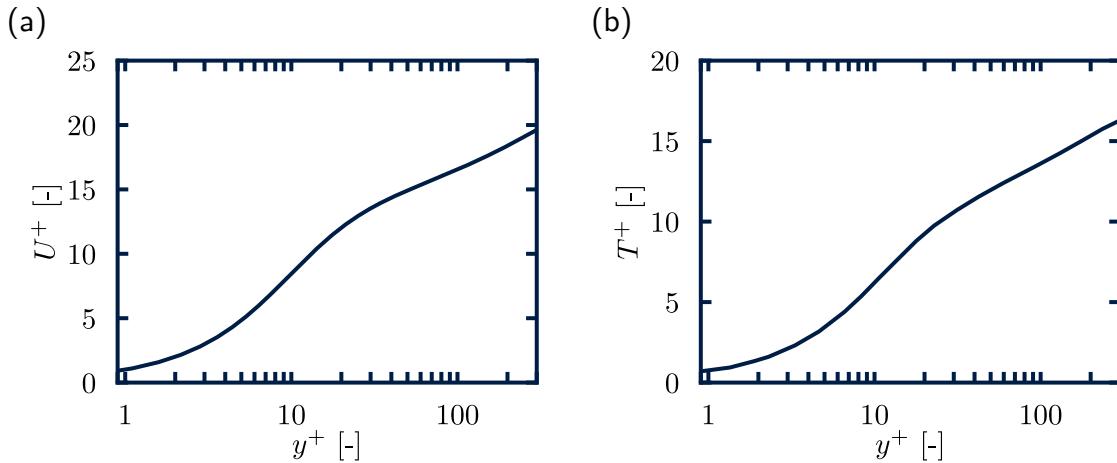


**Figure 23:** A diagram to show the linear variation of velocity and temperature across the wall adjacent cell in the mesh.

The velocity and temperature variation across the cell are **always** linear. Hence, as shown in Figure 23, the wall shear stress and wall heat flux computed by the CFD code are:

$$\tau_w = -\rho \nu_w \frac{U_p}{y_n} \quad q_w = -\rho c_p \alpha_w \left( \frac{T_p - T_w}{y_n} \right) \quad (141)$$

where  $y_p$  is the distance from the wall to the wall adjacent cell centroid,  $U_p$  is the velocity at the cell centroid,  $T_p$  is the temperature at the cell centroid and  $T_w$  is the temperature of



**Figure 24:** Experimental measurements of (a) the velocity profile and (b) the temperature profile normal to the wall in a turbulent flow of air over a flat plate.

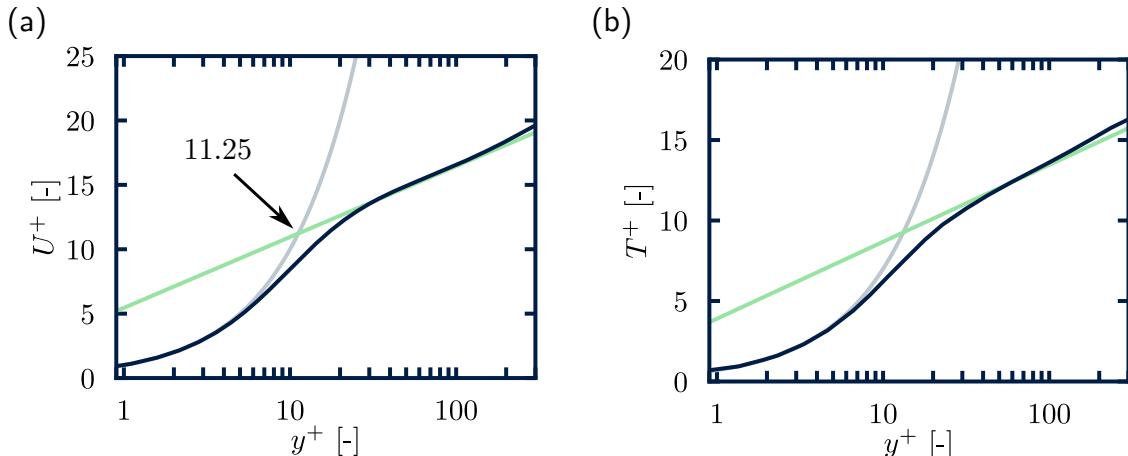
the wall. The kinematic viscosity  $\nu_w$  and the thermal diffusivity  $\alpha_w$  are now denoted with a subscript  $w$  and will be described as the **near wall** kinematic viscosity ( $\nu_w$ ) and the **near wall** thermal diffusivity ( $\alpha_w$ ). The change in notation is to emphasise that these quantities have been modified. The modification is required to ensure that the product of these quantities and the (incorrect) velocity and temperature gradients produces the correct wall shear stress and wall heat flux. The modification to the near wall kinematic viscosity and the near wall thermal diffusivity is called a **wall function**, as the modification is only carried out in the cells that are adjacent to the wall.

At this stage it should be emphasised that a wall function is a modification to  $\nu_w$  and  $\alpha_w$  in the wall adjacent cell. The wall function is not a modification to the velocity and temperature profiles (despite sometimes being called a velocity or temperature wall function in the literature) because the variation across the cell in a CFD code is always linear. The modifications to  $\alpha_w$  and  $\nu_w$  may result in a change in the velocity and temperature solution fields, but the velocity and temperature in the wall adjacent cell are not set directly by a wall function.

## Wall Functions for $\nu_w$ and $\alpha_w$

In order to propose wall functions for  $\nu_w$  and  $\alpha_w$ , the real variation of velocity and temperature between the cell centroid and the wall is required first. These are the actual velocity and temperature profiles that we are trying to model and are non-linear. Figure 24 shows the real variation of velocity and temperature normal to the wall, which were extracted from experimental measurements of fully developed turbulent flow between 2 parallel plates. Further experimental measurements found this profile to be relatively universal (independent of Reynolds number and streamwise pressure gradient) close to the wall. Hence, the profile is often called '**The Universal Law of the Wall**'. Mathematically, the experimental data in Figure 24 can be reasonably approximated by the following function:

$$U^+ = \begin{cases} y^+ & y^+ < 11.25 \\ \frac{1}{\kappa} \log(Ey^+) & y^+ > 11.25 \end{cases} \quad (142)$$



**Figure 25:** Experimental measurements of (a) the velocity profile and (b) the temperature profile normal to the wall in a turbulent flow of air. The light blue and light green lines show the mathematical functions that are fitted to the data for  $y^+ < y_L^+$  and  $y^+ > y_L^+$  respectively.

where  $\kappa = 0.4187$  and  $E = 9.793$  are empirical constants that were fitted to the data. As shown in Figure 25, these profiles are less accurate at reproducing the experimental data when  $5 < y^+ < 30$  (the buffer region). Hence, it is normally recommended to ensure that  $y^+ < 5$  or  $y^+ > 30$  for the final solution to be accurate.

The two profiles intersect where:

$$y^+ = \frac{1}{\kappa} \log(Ey^+) \quad (143)$$

$$y^+ - \frac{1}{\kappa} \log(Ey^+) = 0 \quad \rightarrow \quad f(y^+) = 0 \quad (144)$$

Substituting in the empirical coefficients  $E = 9.793$  and  $\kappa = 0.4187$  and solving for  $y^+$  with a root-finding algorithm (like the bisection method or Newton-Raphson method) results in an intersection point of  $y^+ = 11.25$ . This process of root finding will be demonstrated later in the example problem. A similar mathematical function can also be fitted to the temperature profile.

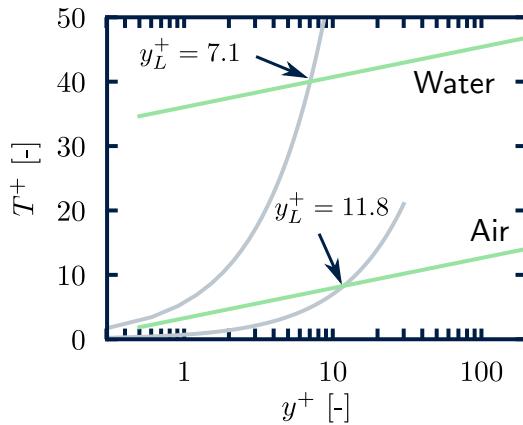
$$T^+ = \begin{cases} Pr y^+ & y^+ < y_L^+ \\ Pr_t \left( \frac{1}{\kappa} \log(Ey^+) + P \right) & y^+ > y_L^+ \end{cases} \quad (145)$$

where  $Pr = \nu/\alpha$  is the molecular Prandtl number,  $Pr_t$  is the turbulent Prandtl number (0.85) and  $P$  is an empirical function of  $Pr$  and  $Pr_t$ .

$$P = 9.24 \left[ \left( \frac{Pr}{Pr_t} \right)^{3/4} - 1 \right] \left[ 1 + 0.28e^{-0.007Pr/Pr_t} \right] \quad (146)$$

The reason for the differences between the velocity and temperature wall functions is that (unlike the velocity profile) the dimensionless temperature profile is **different for different fluids**. The difference is captured by the molecular Prandtl number  $Pr$ . For water and air for example, the molecular Prandtl numbers are:

	$Pr = \nu/\alpha$
Water	0.71
Air	5.68



**Figure 26:** A diagram to show the differences in the temperature profiles for water and air.

A value of  $Pr < 1$  indicates that thermal energy diffuses into the flow faster than momentum. This means that the thermal boundary layer is thinner than the velocity boundary layer when  $Pr < 1$ . A consequence of the temperature profile being different for different fluids is that the intersection between the 2 profiles does not occur at 11.25 (as it does for the velocity profile). The intersection point  $y_L^+$  can be determined by equating the profiles in equation 145.

$$Pr y^+ = Pr_t \left( \frac{1}{\kappa} \log(Ey^+) + P \right) \quad (147)$$

$$Pr y^+ - Pr_t \left( \frac{1}{\kappa} \log(Ey^+) + P \right) = 0 \quad \rightarrow \quad f(y^+) = 0 \quad (148)$$

Substitute in the values of  $Pr$  for the fluid of interest. Then solve the non-linear equation for  $y^+$  using a root-finding algorithm (this process will be demonstrated later in the example problem). This is the intersection point of the two curves ( $y_L^+$ ) and is different for different fluids.

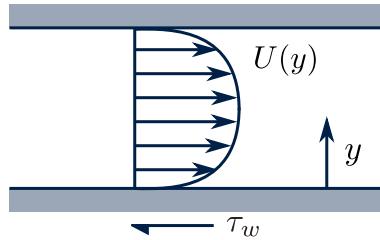
	$Pr$	$y_L^+$
Water	0.71	7.1
Air	5.68	11.8

Figure 26 compares the temperature profiles for water and air. Unlike the velocity profile (which is the same for water and air), the profiles are different and have a different intersection point. This difference in the profiles will also be reflected in the wall function for  $\alpha_w$ . Once again it should be emphasised that the mathematical functions for velocity and temperature are **not applied directly** in the CFD code. The equations are actually used to derive equations for the near wall kinematic viscosity  $\nu_w$  and the near wall thermal diffusivity  $\alpha_w$ . This derivation will be demonstrated in the next section. It is these wall functions for  $\nu_w$  and  $\alpha_w$  that are actually applied in the CFD code.

## $y^+$ and $y^*$

The velocity and temperature profiles in equations 142 and 145 are expressed in dimensionless wall units.

$$U^+ = \frac{U}{u_\tau} \quad y^+ = \frac{\rho u_\tau y}{\mu} \quad T^+ = \frac{(T_w - T_p)\rho c_p u_\tau}{q_w} \quad (149)$$



**Figure 27:** A schematic diagram of the experimental set up that was used to derive the velocity profiles normal to the wall.

The reason for using dimensionless units is so that the profiles are universal and can be applied to any flow scenario (flow over an aerofoil, flow in a pipe, flow over a flat plate etc.), as long as we are close to the wall. This is a reasonable theory, as the flow close to the wall at small scales should be universal, regardless of the freestream flow and the overall shape of the geometry.

However, to define these dimensionless groups, appropriate choices of velocity scale ( $u_\tau$ ) and length scale ( $y$ ) are required. Since the flow is close to the wall, the appropriate length scale  $y$  is the distance normal to the wall. For the velocity scale, it would not be appropriate to use the free-stream velocity ( $U_\infty$ ), as the flow is close to the wall where viscous effects are dominant and the velocity is much lower than the freestream velocity. However, the velocity at the wall itself is zero due to the no-slip condition, so this cannot be used as a velocity scale either. The original approach in the experiments was to chose a velocity scale based on the (square-root of the) wall shear stress. This velocity scale ( $u_\tau$ ) is called the friction-velocity.

$$u_\tau = \sqrt{\frac{|\tau_w|}{\rho}} \quad [\text{m/s}] \quad (150)$$

However, the problem with using a friction velocity based on the wall shear stress is that the wall shear stress is zero at separation points. This would result in a friction velocity of 0, which would create difficulties for the CFD code. An alternative is to use a friction velocity based on the turbulent kinetic energy  $k$  (which would have been more difficult to attain in the experiments).

$$u_\tau = \sqrt{C_\mu^{1/2} k} \quad C_\mu = 0.09 \quad (151)$$

A dimensionless velocity scale based on the turbulent kinetic energy is denoted as  $y^*$  rather than  $y^+$  by CFD codes.

$$U^* = \frac{U}{u_\tau} \quad y^* = \frac{\rho u_\tau y}{\mu} \quad T^* = \frac{(T_w - T_p) \rho c_p u_\tau}{q_w} \quad u_\tau = \sqrt{C_\mu^{1/2} k} \quad (152)$$

In most scenarios,  $y^+$  and  $y^*$  are almost identical and either can be used. However, the majority of modern CFD codes prefer to use  $y^*$  in their calculations.

## Deriving a Wall Function for $\nu_w$

We now have a mathematical model for the velocity profile normal to the wall:

$$U^+ = \begin{cases} y^+ & y^+ < 11.25 \\ \frac{1}{\kappa} \log(Ey^+) & y^+ > 11.25 \end{cases} \quad (153)$$

### 3 WALL FUNCTIONS

---

Replace the dimensionless variables with the actual variables ( $U^+ = U/u_\tau$  and  $y^+ = \rho u_\tau y / \mu$ ):

$$\begin{aligned}\frac{U}{u_\tau} &= \frac{\rho u_\tau y}{\mu} & y^+ < 11.25 \\ \frac{U}{u_\tau} &= \frac{1}{\kappa} \log \left( E \frac{\rho u_\tau y}{\mu} \right) & y^+ > 11.25\end{aligned}\quad (154)$$

The trick to calculating the wall shear stress  $\tau_w$  from these profiles is to write:

$$\frac{U}{u_\tau} = \frac{U u_\tau}{u_\tau^2} \quad (155)$$

Then recall that  $u_\tau = \sqrt{|\tau_w|/\rho}$  and therefore  $u_\tau^2 = -\tau_w/\rho$

$$\frac{U}{u_\tau} = \frac{U u_\tau}{u_\tau^2} = -\frac{U \rho u_\tau}{\tau_w} \quad (156)$$

The velocity profile then becomes:

$$\begin{aligned}-\frac{U \rho u_\tau}{\tau_w} &= \frac{\rho u_\tau y}{\mu} & y^+ < 11.25 \\ -\frac{U \rho u_\tau}{\tau_w} &= \frac{1}{\kappa} \log \left( E \frac{\rho u_\tau y}{\mu} \right) & y^+ > 11.25\end{aligned}\quad (157)$$

Rearrange for the wall shear stress ( $\tau_w$ ):

$$\begin{aligned}\tau_w &= -\mu \frac{U}{y} & y^+ < 11.25 \\ \tau_w &= -\frac{U \rho u_\tau}{\frac{1}{\kappa} \log \left( E \frac{\rho u_\tau y}{\mu} \right)} & y^+ > 11.25\end{aligned}\quad (158)$$

This equation gives the **real** wall shear stress that is exhibited by a turbulent velocity profile over a flat plate. In a CFD code, the velocity profile between the wall adjacent cell centroid and the wall is always linear. As shown in Figure 23, the velocity at the cell centroid is  $U_p$  and the cell is a height  $y_p$  normal to the wall. Hence, the wall shear stress **in the CFD code** is always:

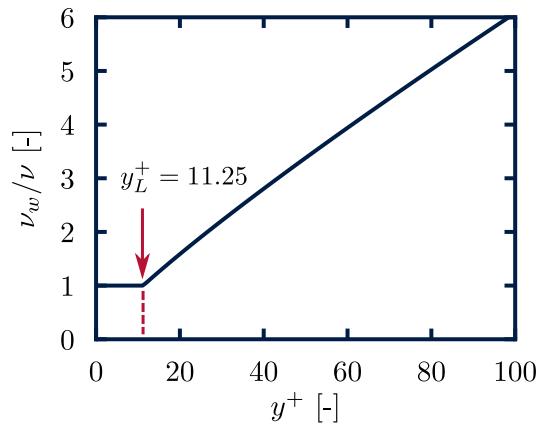
$$\tau_w = -\rho \nu \left. \frac{\partial U}{\partial y} \right|_{y=0} = -\rho \nu_w \frac{U_p}{y_p} \quad (159)$$

To ensure that the CFD code always computes the correct wall shear stress, equate equation 159 with equation 158 (noting that the velocity and wall normal distance in equation 158 are now evaluated at the cell centroid,  $U = U_p$  and  $y = y_p$ ).

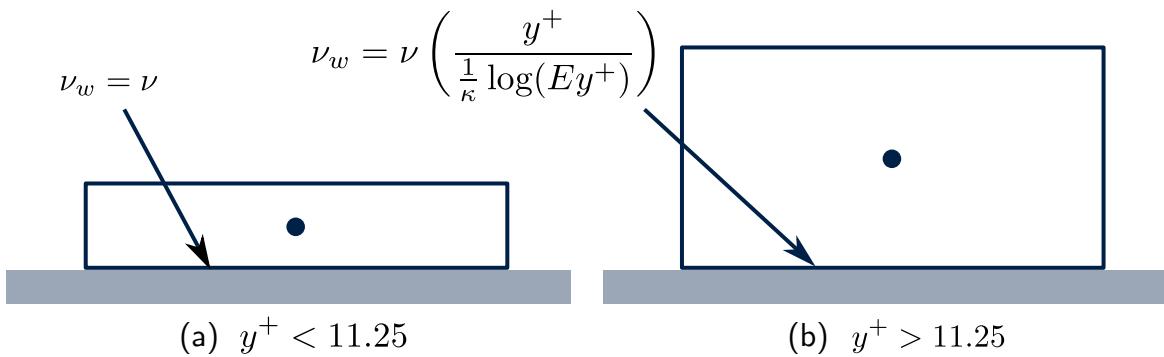
$$\begin{aligned}-\rho \nu_w \frac{U_p}{y_p} &= -\rho \nu \frac{U_p}{y_p} & y^+ < 11.25 \\ -\rho \nu_w \frac{U_p}{y_p} &= -\frac{U_p \rho u_\tau}{\frac{1}{\kappa} \log \left( E \frac{\rho u_\tau y_p}{\mu} \right)} & y^+ > 11.25\end{aligned}\quad (160)$$

Rearrange for  $\nu_w$ .

$$\begin{aligned}\nu_w &= \nu & y^+ < 11.25 \\ \nu_w &= \frac{u_\tau y_p}{\frac{1}{\kappa} \log \left( E \frac{\rho u_\tau y_p}{\mu} \right)} & y^+ > 11.25\end{aligned}\quad (161)$$



**Figure 28:** The variation of near wall kinematic viscosity  $\nu_w$  with  $y^+$



**Figure 29:** A diagram to show the difference in wall treatment when (a)  $y^+ < 11.25$  and (b)  $y^+ > 11.25$

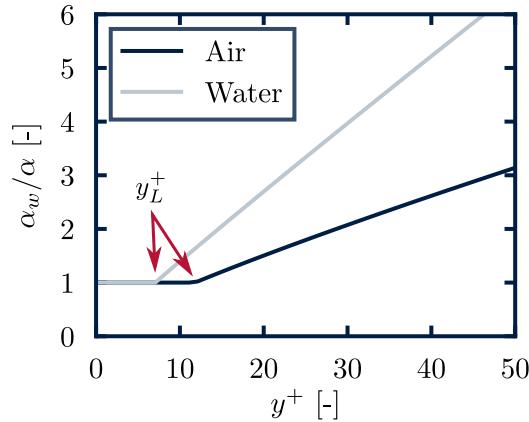
Make a final substitution ( $y^+ = y_p u_\tau / \nu$ ) to simplify the equation:

$$\nu_w = \begin{cases} \nu & y^+ < 11.25 \\ \nu \left( \frac{y^+}{\frac{1}{\kappa} \log(Ey^+)} \right) & y^+ > 11.25 \end{cases} \quad (162)$$

Equation 162 is the **wall function** for the near wall kinematic viscosity  $\nu_w$  and is plotted in Figure 28. Physically, the equation states that if the wall adjacent cell is thin enough ( $y^+ < 11.25$ ), then the near wall kinematic viscosity is set equal to the molecular viscosity of the fluid ( $\nu$ ). This will result in the correct wall shear stress as the true velocity profile is linear. However, if the cell is large ( $y^+ > 11.25$ ), then the CFD code assumes that the velocity profile between the wall adjacent cell centroid and the wall is linear. This is not correct, as the real velocity profile is non-linear. However, if the near wall kinematic viscosity is increased using equation 162, the product of the near wall kinematic viscosity and the (incorrect) velocity gradient will yield the correct wall shear stress.

$$\frac{\tau_w}{\rho} = -\text{Near Wall Kinematic Viscosity} \times \text{Velocity Gradient} \quad (163)$$

Figure 29 shows a diagram to illustrate this process. The astute reader will notice that



**Figure 30:** The variation of the near wall thermal diffusivity  $\alpha_w$  with  $y^+$  for air and water.

equation 162 can be written concisely as:

$$\nu_w = \nu * \left[ \frac{y^+}{f(y^+)} \right] \quad (164)$$

where  $f(y^+)$  is given by equation 153. Hence, if a new function for  $U^+$  is proposed (replacing equation 142), then the wall function for  $\nu_w$  can be deduced directly from the above equation.

It should also be noted that  $y^*$  can be used in place of  $y^+$  in the wall function for near wall kinematic viscosity.

$$\nu_w = \begin{cases} \nu & y^* < 11.25 \\ \nu \left( \frac{y^*}{\frac{1}{\kappa} \log(Ey^*)} \right) & y^* > 11.25 \end{cases} \quad (165)$$

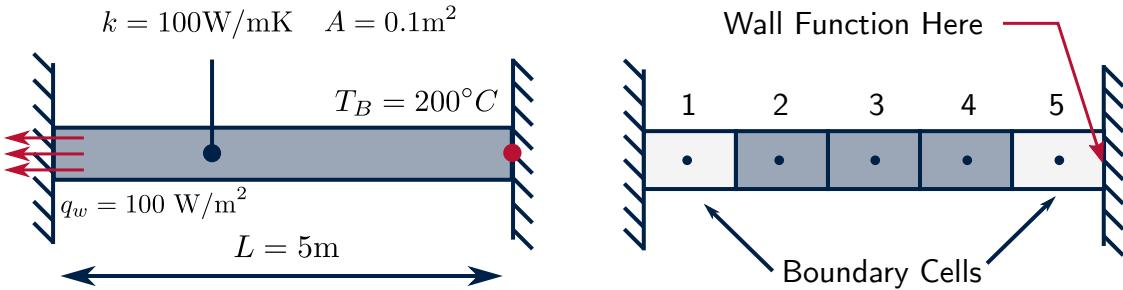
The majority of CFD codes (ANSYS Fluent, ANSYS CFX, Star CCM+) use the  $y^*$  formulation instead of  $y^+$ . OpenFOAM offers the choice of both: `nutUWallFunction` (for  $y^+$ ) and `nutkWallFunction` (for  $y^*$ ).

## Wall Function for $\alpha_w$

The wall function for the near wall thermal diffusivity ( $\alpha_w$ ) takes a similar form to  $\nu_w$ :

$$\alpha_w = \begin{cases} \alpha & y^+ < y_L^+ \\ \alpha \left[ \frac{Pr y^+}{Pr_t (\frac{1}{\kappa} \log(Ey^+) + P)} \right] & y^+ > y_L^+ \end{cases} \quad (166)$$

For brevity, the derivation will not be provided here, as it follows an identical process to the wall function for  $\nu_w$  in the previous section. The wall function for  $\alpha_w$  is shown in Figure 30 for air and water. In the same manner as  $\nu_w$ , for  $y^+ < y_L^+$ ,  $\alpha_w$  takes the value of the molecular thermal diffusivity  $\alpha$ . For  $y^+ > y_L^+$ , the thermal diffusivity is increased, to ensure that the product of the near wall thermal diffusivity and the temperature gradient gives the correct wall heat flux. Notice (in Figure 30) that air and water have different values of  $y_L^+$



**Figure 31:** An example problem to demonstrate 1D heat diffusion in a bar. A wall function is applied at the right hand end (where the temperature is fixed).

and the rate of increase of  $\alpha_w$  with  $y^+$  is greater for water than air. In general, the shape of the wall function will be different for every fluid, depending on  $Pr$ . The wall function for  $\nu_w$  however, is universal for all fluids.

Now that the wall functions for  $\nu_w$  and  $\alpha_w$  have been presented, the remainder of the chapter will demonstrate how the wall functions are incorporated into CFD codes. This demonstration will be carried through an example problem.

## Example Problem: Heat Diffusion in a Bar

Consider 1D steady-state diffusion of heat in a bar, as shown in Figure 31. The left end of the bar has a fixed heat flux ( $q_{\text{wall}}$ ) of  $100 \text{ W/m}^2$ , while the right end of the bar is at a fixed temperature of ( $T_B$ )  $200^\circ\text{C}$ . There is a constant heat source ( $\bar{S}$ ) of  $1000 \text{ W/m}^3$  applied to the bar. The bar has a density of  $8000 \text{ kg/m}^3$  and a specific heat capacity of  $500 \text{ J/kg K}$ . For ease of comparison, this example problem has the same geometry, mesh and boundary conditions as the previous chapter. However, unlike the previous chapter, a wall function for the thermal diffusivity  $\alpha_w$  will be applied at the right hand end (where the fixed temperature is applied). To evaluate the wall function, the material is assumed to have a Prandtl number ( $Pr$ ) of 0.71 and the right boundary cell has a  $y^+ = 30$ .

### Step 1: Divide the Geometry into a Mesh

For the example in Figure 31, divide the geometry into a mesh of 5 cells of equal length. The length of each cell ( $L_{\text{cell}}$ ) is given by:

$$L_{\text{cell}} = \frac{L}{N} = \frac{5}{5} = 1\text{m} \quad (167)$$

Because the cells are uniformly distributed and have equal size, the distance between cell centroids  $d$  is equivalent to the length of the cells. Hence:

$$d_{LP} = d_{PR} = d = 1\text{m} \quad (168)$$

### Step 2: Evaluate the Wall Function

Before the material properties can be assigned, the wall function needs to be evaluated. As the wall function for  $\alpha_w$  is being applied (equation 166), the first stage is to evaluate the empirical

function  $P$ . For a molecular Prandtl number of 0.71 and a turbulent Prandtl number of 0.85, the function  $P$  evaluates as:

$$P = 9.24 \left[ \left( \frac{Pr}{Pr_t} \right)^{3/4} - 1 \right] \left[ 1 + 0.28e^{-0.007Pr/Pr_t} \right] = -1.491 \quad (169)$$

Now the intersection point between the two sections of the wall function ( $y_L^+$ ) needs to be calculated. This requires the solution of the following non-linear equation for  $y^+$  (equation 148):

$$Pr y^+ - Pr_t \left( \frac{1}{\kappa} \log (Ey^+) + P \right) = 0 \quad (170)$$

This non-linear equation can be solved with any root finding algorithm (Bisection, Newton-Raphson etc.). The Newton-Raphson method gives rapid convergence and will be used here. To use the Newton-Raphson method, define:

$$f = Pr y^+ - Pr_t \left( \frac{1}{\kappa} \log (Ey^+) + P \right) \quad (171)$$

$$\frac{df}{dy^+} = Pr - \frac{Pr_t}{\kappa y^+} \quad (172)$$

The Newton-Raphson iteration proceeds by evaluating:

$$y_{i+1}^+ = y_i^+ - \frac{f}{df/dy^+} \quad (173)$$

Starting from an initial guess ( $y_0^+$ ) of 11.0, the Newton-Raphson iteration gives:

Iteration	$y_i^+$	$y_{i+1}^+$
1	11.000	11.806
2	11.806	11.796
3	11.796	11.796

Hence, the solution of the Newton Raphson procedure gives  $y_L^+ = 11.796$  when  $Pr = 0.71$ . Now that  $y_L^+$  has been evaluated,  $\alpha_w$  can be calculated (using equation 166).

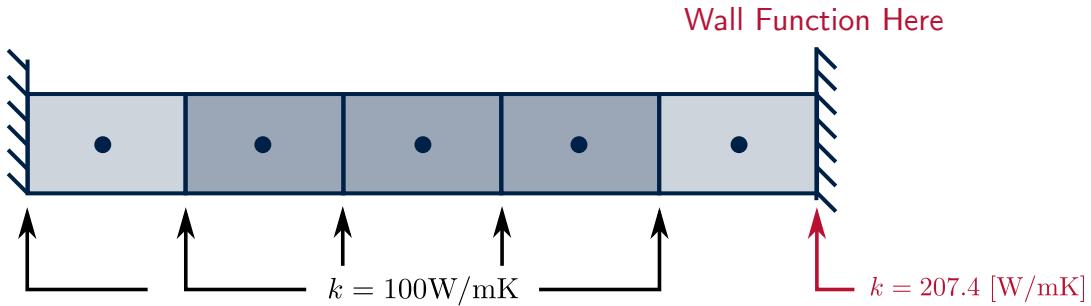
$$\alpha_w = \begin{cases} \alpha & y^+ < y_L^+ \\ \alpha \left[ \frac{Pr y^+}{Pr_t \left( \frac{1}{\kappa} \log (Ey^+) + P \right)} \right] & y^+ > y_L^+ \end{cases} \quad (174)$$

In this example,  $y^+ = 30$ . Hence,  $\alpha_w/\alpha = 2.074$  and the near wall thermal diffusivity will be approximately double the thermal diffusivity of the interior cells.

## Step 2: Assign Material Properties

The wall function is applied to the molecular thermal diffusivity  $\alpha$ . The molecular thermal diffusivity is defined as:

$$\alpha = \frac{k}{\rho c_p} \quad (175)$$



**Figure 32:** Thermal conductivity of each of the faces in the mesh when a wall function is applied to the right end of the bar.

where  $k$  is the thermal conductivity,  $\rho$  is the material density and  $c_p$  is the specific heat capacity. For the example problem considered here:

$$\alpha = \frac{k}{\rho c_p} = \frac{100}{8000 * 500} = 2.5 \times 10^{-5} [\text{m}^2/\text{s}] \quad (176)$$

This is the thermal diffusivity for all the interior cells and interior faces in the mesh. For the right boundary face, the thermal diffusivity is modified by the wall function:

$$\frac{\alpha_w}{\alpha} = 2.074 \quad \alpha_w = 5.185 \times 10^{-5} [\text{m}^2/\text{s}] \quad (177)$$

As the thermal diffusivity of the right boundary face is now  $\alpha_w$ , then the thermal conductivity of the face  $k_w$  is also modified.

$$k_w = \rho c_p \alpha_w = 8000 * 500 * 5.185 \times 10^{-5} = 207.4 [\text{W}/\text{mK}] \quad (178)$$

It follows that the thermal conductivity of all the faces in the mesh is constant (100 W/mK), except for the right boundary face, where  $k$  modified by the wall function. The variation of thermal conductivity across the mesh is shown in Figure 32. For the interior cells in the mesh, the diffusive heat flux across the cell faces  $DA$  is given by:

$$DA = \frac{kA}{d} = \frac{100 * 0.1}{1} = 10 [\text{W}/\text{K}] \quad (179)$$

For the right boundary cell, the diffusive heat flux is evaluated using the modified thermal conductivity ( $k_w$ ).

$$DA = \frac{k_w A}{d} = \frac{200 * 0.1}{1} = 20.74 [\text{W}/\text{K}] \quad (180)$$

The heat source in each cell is not affected by the wall function. It is given by:

$$\bar{SV} = \bar{S}AL_{\text{cell}} = 1000 * 0.1 * 1 = 100 [\text{W}] \quad (181)$$

### Step 3: Calculate the Matrix Coefficients

The matrix coefficients for 1D heat diffusion in a bar with fixed temperature (Dirichlet) boundary conditions at the left end and fixed heat flux (Neumann) boundary conditions at the right end are:

### 3 WALL FUNCTIONS

---

Matrix Coefficients

	$a_L$	$a_R$	$a_P$	$S_p$	$S_u$
Boundary (L)	0	$D_R A_R$	$a_l + a_r - S_p$	0	$-q_A A_L + \bar{S}V$
Interior	$D_L A_L$	$D_R A_R$	$a_l + a_r - S_p$	0	$\bar{S}V$
Boundary (R)	$D_L A_L$	0	$a_l + a_r - S_p$	$-2D_R A_R$	$T_B(2D_R A_R) + \bar{S}V$

However, as a wall function is applied to the right face of the right boundary cell, the coefficient  $D_R$  (highlighted in red in the table above) is modified by the wall function. The other coefficients remain unchanged by the wall function. Filling in the table with the calculated coefficients:

Matrix Coefficients

	$a_L$	$a_R$	$a_P$	$S_p$	$S_u$
Boundary (L)	0	10	10	0	90
Interior	10	10	20	0	100
Boundary (R)	10	0	<b>51.5</b>	<b>-41.5</b>	<b>8395.9</b>

It is clear that the wall function enters the matrix equations through the diagonal coefficient  $a_p$  and the source term  $S_u$  of the cell that contains the boundary face. The coefficients in the other cells remain unchanged.

Using summation notation, the wall function modifies the face contribution of the right boundary face through  $k_w$ . No other changes are made to the face contributions of the other faces.

Summation Notation

Face Type	$a_P$	$a_N$	$S_u$
Interior	10	10	0
Neumann (Left)	0	0	-10
Dirichlet (Right, No Wall Function)	<b>20</b>	0	<b>4000</b>
Dirichlet (Right, Wall Function)	<b>41.5</b>	0	<b>8295.9</b>

## Step 4: Assemble and Solve the Matrices

As a reminder, the matrices are populated in the following manner:

$$\begin{bmatrix} a_{p1} & -a_{r1} & 0 & 0 & 0 \\ -a_{l2} & a_{p2} & -a_{r2} & 0 & 0 \\ 0 & -a_{l3} & a_{p3} & -a_{r3} & 0 \\ 0 & 0 & -a_{l4} & a_{p4} & -a_{r4} \\ 0 & 0 & 0 & -a_{l5} & a_{p5} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} S_{u1} \\ S_{u2} \\ S_{u3} \\ S_{u4} \\ S_{u5} \end{bmatrix} \quad (182)$$

Filling in the known coefficients from the table above:

$$\begin{bmatrix} 10 & -10 & 0 & 0 & 0 \\ -10 & 20 & -10 & 0 & 0 \\ 0 & -10 & 20 & -10 & 0 \\ 0 & 0 & -10 & 20 & -10 \\ 0 & 0 & 0 & -10 & \textcolor{red}{51.5} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 90 \\ 100 \\ 100 \\ 100 \\ \textcolor{red}{8395.9} \end{bmatrix} \quad (183)$$

For the case when a wall function is not applied,  $D_R A_R = kA/d = 10 \text{ W/mK}$  and the matrices reduce back to the same matrices from the previous chapter.

$$\begin{bmatrix} 10 & -10 & 0 & 0 & 0 \\ -10 & 20 & -10 & 0 & 0 \\ 0 & -10 & 20 & -10 & 0 \\ 0 & 0 & -10 & 20 & -10 \\ 0 & 0 & 0 & -10 & \textcolor{blue}{30} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 90 \\ 100 \\ 100 \\ 100 \\ \textcolor{blue}{4100} \end{bmatrix} \quad (184)$$

Hence, the effect of the wall function is localised to the cell where the wall function is applied.

## Run the Example Problem Yourself!

Now, open either the Excel spreadsheets or the Python source code and solve the problem for yourself.

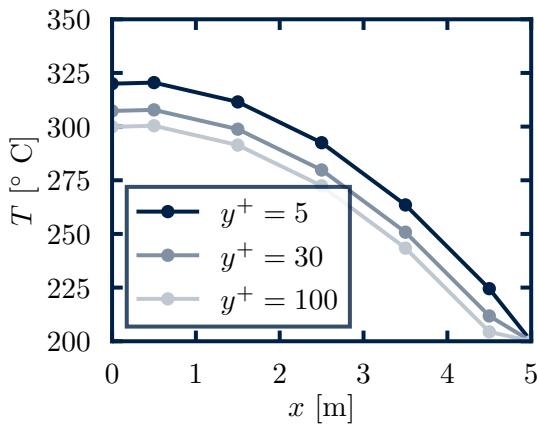
Excel `wallFunction.xlsx`

Python `wallFunction.py`

Examine the calculation of the coefficients, the assembly of the matrices and run the code. A fixed value of  $y^+ = 30$  has been set initially. Try changing the value of  $y^+$  and observe the changes in the solution. Note that in a real CFD code,  $y^+$  changes automatically in response to changes in the mesh. In this problem, we are specifying the value of  $y^+$  **directly** instead, so that we can observe the changes in solution for a **fixed mesh**. Hence, the solution is not strictly accurate, but is useful for demonstration purposes.

## Results

Figure 33 shows the temperature variation along the bar, for different values of  $y^+$  on the right boundary face. Notice that the temperature of the right boundary face is fixed at



**Figure 33:** Temperature variation along the 1D bar with different values of  $y^+$  on the right boundary face.

200°C. However, the temperature of the wall adjacent cell centroid ( $T_5$ ) is modified by the wall function. This results in a change in shape of the entire temperature profile.

The temperature of the wall adjacent cell centroid ( $T_5$ ) is not modified because the wall function changes its temperature directly. The temperature of the cell is modified because the wall function modifies the thermal conductivity of the right face of this cell from 100 W/mK to 207.4 W/mK. In order to maintain the same heat flux from the wall ( $q_w$ ), the temperature gradient ( $dT/dx$ ) is reduced to counterbalance the increase in  $k_w$ . The table below summarises the heat flux balance for every cell in the mesh for a  $y^+$  of (a) 5 and (b) 30. It is clear that the heat flux through the faces of each of the cells remains unchanged by the wall function, but the temperature profile does change (see Figure 33).

#### Heat Flux Balance

(a)  $y^+ = 5$

Cell	$Q_l$ [W]	$Q_r$ [W]	$\bar{S}V$ [W]	Error
1	<b>10</b>	90	100	0
2	-90	190	100	0
3	-190	290	100	0
4	-290	390	100	0
5	-390	<b>490</b>	100	0

Heat flux out of the bar =  $490 + 10 = 500\text{W}$

Heat generated in the bar =  $100 + 100 + 100 + 100 + 100 = 500\text{W}$

(b)  $y^+ = 30$

Cell	$Q_l$ [W]	$Q_r$ [W]	$\bar{S}V$ [W]	Error
1	<b>10</b>	90	100	0
2	-90	190	100	0
3	-190	290	100	0
4	-290	390	100	0
5	-390	<b>490</b>	100	0

Heat flux out of the bar =  $490 + 10 = 500\text{W}$

Heat generated in the bar =  $100 + 100 + 100 + 100 + 100 = 500\text{W}$

As the heat flux from the left end of the bar is fixed by the boundary condition (10W) and 500W is generated in the bar, 490W must pass out of the bar at the right end. Hence, as the wall function modifies  $\alpha_w$ , the temperature gradient at the wall changes, so that the heat flux remains the same (490W).