Gas dynamics and Heat and Mass Transfer

Numerical Solution of the Convection–Diffusion Equations

Student: Pedro López Sancha

Professor: Carlos-David Pérez Segarra

Aerospace Technology Engineering
The School of Industrial, Aerospace and Audiovisual Engineering of Terrassa
Technic University of Catalonia

August 24, 2021

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Superior d'Enginyeries Industrial, Aeroespacial i Audiovisual de Terrassa

Contents

1 Introduction

2 The convection–diffusion equations

In this section we perform a rigorous derivation of the convection–diffusion equations. To begin, Reynolds transport theorem is presented as a generalization of Leibniz integral rule and is proved. Next, we derive the distinct conservation laws. Finally we find a common structure on convection–diffusion equations which will ease their numerical study.

2.1 Reynolds transport theorem

The main result needed to derive the conservation laws of the subsequent subsections, namely, mass, momentum, energy and species conservation, is Reynolds transport theorem. This theorem is a generalization to higher dimensions of Leibniz integral rule:

Theorem 2.1 (Leibniz integral rule [1])**.** Let $U \subset \mathbb{R}$ be a closed interval and $I = [t_1, t_2]$, and let $a, b\colon I \to U$ be continuous functions with continuous derivative. Let $f\colon U \times I \to \mathbb{R}$, $(x,t) \mapsto f(x,t)$ be a continuous function such that $\frac{\partial f}{\partial t}$ is also continuous. Then for all $t \in I$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{a(t)}^{b(t)} f(x,t)\,\mathrm{d}x = \int_{a(t)}^{b(t)} \frac{\partial f}{\partial t}\,\mathrm{d}x + f(b(t),t)b'(t) - f(a(t),t)a'(t) \tag{2.1}$$

Reynolds transport theorem allows us to compute the left–hand side of (2.1) but in the more general context where f is integrated over a bounded smooth control volume $\mathcal{V}(t) \subset \mathbb{R}^m$ which depends upon time.

Theorem 2.2 (Reynolds transport theorem [2])**.** Let $U \subset \mathbb{R}^m$ be a compact set and let $\mathcal{V}(t)$ be a control volume depending on time such that it is smooth and $\mathcal{V}(t) \subset U$ for all $t \in I = [0,T]$ with $T > 0$. Let $\mathcal{S}(t) = \partial\mathcal{V}(t)$ be the boundary of $\mathcal{V}(t)$ and let $F \in \mathcal{C}^1(U \times I, \mathbb{R})$ be a scalar field. Then for all $t \in I$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \int_{\mathcal{V}(t)} \frac{\partial F}{\partial t}(\mathbf{x},t)\,\mathrm{d}\mathbf{x} + \int_{\mathcal{S}(t)} F(\mathbf{x},t)\mathbf{b}\cdot\mathbf{n}\,\mathrm{d}S \tag{2.2}$$

where $\mathbf{b}\colon \mathcal{S}(t) \to \mathbb{R}^m$ is the local velocity of the control surface.

Proof. The moving control volume $\mathcal{V}(t)$ can be seen as the image of an initial region $\mathcal{V}(0)$ by a family of smooth maps $\xi\colon U \times I \subset \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^m$, that is to say, $\mathcal{V}(t) = \xi(\mathcal{V}(0),t)$ for all $t \in I$. Moreover, by fixing one time t, the mapping $\xi(\cdot,t)\colon (0) \to (t)$ can be assumed to be a diffeomorphism. Since F is continuous, we can apply the Change of Variables Theorem taking $\mathbf{x} = \xi(\mathbf{x}_0,t)$,

$$\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \int_{\mathcal{V}(0)} F(\xi(\mathbf{x}_0,t),t)\left|\det\left(\frac{\partial\xi}{\partial\mathbf{x}_0}(\mathbf{x}_0,t)\right)\right|\mathrm{d}\mathbf{x}_0$$

where the determinant of the jacobian matrix $\det\left(\frac{\partial\xi}{\partial\mathbf{x}_0}(\mathbf{x}_0,t)\right)$ can be assumed to be positive for small enough T, hence the absolute value is dropped. Applying differentiation under the integral sign (Theorem A.1) with respect to t yields

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} &= \int_{\mathcal{V}(0)} \frac{\partial}{\partial t}\left\{F(\xi(\mathbf{x}_0,t),t)\ \det\left(\frac{\partial\xi}{\partial\mathbf{x}_0}(\mathbf{x}_0,t)\right)\right\}\mathrm{d}\mathbf{x}_0\\ &= \int_{\mathcal{V}(0)} \frac{\partial}{\partial t}\left\{F(\xi(\mathbf{x}_0,t),t)\right\}\det\left(\frac{\partial\xi}{\partial\mathbf{x}_0}(\mathbf{x}_0,t)\right)\mathrm{d}\mathbf{x}_0 + \int_{\mathcal{V}(0)} F(\xi(\mathbf{x}_0,t),t)\,\frac{\partial}{\partial t}\left\{\det\left(\frac{\partial\xi}{\partial\mathbf{x}_0}(\mathbf{x}_0,t)\right)\right\}\mathrm{d}\mathbf{x}_0\end{aligned}$$

On the one hand,

$$\frac{\partial}{\partial t}\left\{F(\xi(\mathbf{x}_0,t),t)\right\}\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}(\mathbf{x}_0,t)\right) = \left\{\frac{\partial F}{\partial t}(\xi(\mathbf{x}_0,t),t) + \boldsymbol{\nabla} F(\xi(\mathbf{x}_0,t),t)\cdot \xi_t(\mathbf{x}_0,t)\right\}\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}(\mathbf{x}_0,t)\right)$$

where $\xi_t = \frac{\partial \xi}{\partial t}$. On the other hand, using matrix calculus,

$$F(\xi(\mathbf{x}_0,t),t)\,\frac{\partial}{\partial t}\left\{\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}(\mathbf{x}_0,t)\right)\right\} = F(\xi(\mathbf{x}_0,t),t)\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}(\mathbf{x}_0,t)\right)\boldsymbol{\nabla}\cdot \xi_t(\mathbf{x}_0,t)$$

Thereby the integral is written as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} &= \int_{\mathcal{V}(0)}\left\{\frac{\partial F}{\partial t} + \boldsymbol{\nabla} F\cdot \xi_t + F\boldsymbol{\nabla}\cdot \xi_t\right\}\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}\right)\mathrm{d}\mathbf{x}_0 \\ &= \int_{\mathcal{V}(0)}\left\{\frac{\partial F}{\partial t} + \boldsymbol{\nabla}\cdot(F\xi_t)\right\}\det\left(\frac{\partial \xi}{\partial \mathbf{x}_0}\right)\mathrm{d}\mathbf{x}_0\end{aligned}$$

So as to obtain an integral over $\mathcal{V}(t)$, the previous change of variables is reverted, that is, $\mathbf{x}_0 = \xi^{-1}(\mathbf{x},t)$. In order not to complicate notation, let $\mathbf{b}(\mathbf{x},t) = \xi_t(\xi^{-1}(\mathbf{x},t),t)$, then

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \int_{\mathcal{V}(t)}\left\{\frac{\partial F}{\partial t} + \boldsymbol{\nabla}\cdot(F\mathbf{b})\right\}(\mathbf{x},t)\,\mathrm{d}\mathbf{x}$$

For a fixed $\mathbf{x}_0 \in \mathcal{V}(0)$, $\xi(\mathbf{x}_0,\cdot)$ is a function of time giving how $\mathbf{x}_0$ moves, hence $\xi_t(\mathbf{x}_0,t)$ is the instantaneous velocity of $\mathbf{x}_0$. To end, an application of divergence theorem yields the final formula

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} F(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \int_{\mathcal{V}(t)}\frac{\partial F}{\partial t}(\mathbf{x},t)\,\mathrm{d}\mathbf{x} + \int_{\mathcal{S}(t)} F(\mathbf{x},t)\mathbf{b}\cdot\mathbf{n}\,\mathrm{d}S$$

□

In an intuitive manner, Reynolds tranport theorem quantifies how a quantity over a control volume varies when the control volume itself depends on time due to transport phenomena, diffusion, etc.

2.2 Continuity equation

For the purposes of this project, where no nuclear nor relativistic effects are considered, mass is a property preserved over time. Let $\mathcal{V} \subset \mathbb{R}^m$ be a bounded control volume, which may depend on time, and let $\rho = \rho(\mathbf{x},t)$ be the mass density defined over $\mathcal{V}$ for each time $t \in I$. It can be assumed, without loss of generality, that $\mathcal{V}(t)$ is open for each $t \in I$. The mass enclosed by $\mathcal{V}$ at time t is

$$m(t) = \int_{\mathcal{V}(t)} \rho(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \int_{\mathcal{V}(t)} \rho\,\mathrm{d}\mathbf{x} \tag{2.3}$$

and as a result of the mass conservation principle

$$\frac{\mathrm{d}m(t)}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}(t)} \rho(\mathbf{x},t)\,\mathrm{d}x = 0 \tag{2.4}$$

Now applying Reynolds transport theorem to (2.4) setting $\mathbf{b} = \mathbf{v}$,

$$\int_{\mathcal{V}(t)} \frac{\partial \rho}{\partial t}\,\mathrm{d}\mathbf{x} + \int_{\mathcal{S}(t)} \rho\mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S = 0 \tag{2.5}$$

So as to transform the surface integral into a volume integral, divergence theorem must be applied:

$$\int_{\mathcal{V}(t)} \frac{\partial \rho}{\partial t} \, \mathrm{d}\mathbf{x} + \int_{\mathcal{V}(t)} \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \, \mathrm{d}\mathbf{x} = \int_{\mathcal{V}(t)} \left\{ \frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \right\} \mathrm{d}\mathbf{x} = 0 \tag{2.6}$$

Since the control volume is non–degenerate, it has non–zero m–dimensional volume for all $t \in I$, that is,

$$V(t) = |\mathcal{V}(t)| = \int_{\mathcal{V}(t)} 1 \, \mathrm{d}\mathbf{x} \neq 0 \quad \forall t \in I \tag{2.7}$$

Fix one time $t_0 \in I$ and let $\mathbf{x}_0 \in \mathcal{V}(t_0)$ be a point inside the moving control volume. As $\mathcal{V}(t)$ is open, there exists $r_0 > 0$ such that $B(\mathbf{x}_0, r_0) \subset \mathcal{V}(t_0)$, where $B(\mathbf{x}_0, r_0) = \{\mathbf{x} \in \mathbb{R}^m \mid |\mathbf{x} - \mathbf{x}_0| < r_0\}$. Since $B(\mathbf{x}_0, r_0)$ can be regarded as a control volume inside $\mathcal{V}(t_0)$, equation (2.6) also holds when the integral is computed over $B(\mathbf{x}_0, r_0)$, i.e.

$$\int_{B(\mathbf{x}_0, r_0)} \left\{ \frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \right\} \mathrm{d}\mathbf{x} = 0 \tag{2.8}$$

and it is still true if it is divided by the volume of $B(\mathbf{x}_0, r_0)$,

$$\frac{1}{|B(\mathbf{x}_0, r_0)|} \int_{B(\mathbf{x}_0, r_0)} \left\{ \frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \right\} \mathrm{d}\mathbf{x} = 0 \tag{2.9}$$

When $r_0 \to 0$, $|B(\mathbf{x}_0, r_0)| \to 0$, hence applying Lebesgue's Differentiation Theorem (Theorem A.2),

$$\lim_{r_0 \to 0} \frac{1}{|B(\mathbf{x}_0, r_0)|} \int_{B(\mathbf{x}_0, r_0)} \left\{ \frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \right\} \mathrm{d}\mathbf{x} = \left(\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \right) (\mathbf{x}_0, t_0) = 0 \tag{2.10}$$

Since this is true for each $\mathbf{x}_0 \in \mathcal{V}(t_0)$ and $t_0 \in I$ is arbitrary, the continuity equation is

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) = 0 \tag{2.11}$$

2.3 Momentum equation

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = \boldsymbol{\nabla} \cdot (\mu \boldsymbol{\nabla} \mathbf{v}) + \{ \boldsymbol{\nabla} \cdot (\tau - \mu \boldsymbol{\nabla} \mathbf{v}) - \boldsymbol{\nabla} p + \rho \mathbf{g} \} \tag{2.12}$$

2.4 Energy equation

$$\frac{\partial (\rho T)}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v} T) = \boldsymbol{\nabla} \cdot \left(\frac{\lambda}{c_v} \boldsymbol{\nabla} T \right) + \left\{ \frac{\tau \circ \boldsymbol{\nabla} \mathbf{v} - \boldsymbol{\nabla} \cdot \dot{\mathbf{q}}^R - p \boldsymbol{\nabla} \cdot \mathbf{v}}{c_v} \right\} \tag{2.13}$$

2.5 Species equation

$$\frac{\partial (\rho Y_k)}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v} Y_k) = \boldsymbol{\nabla} \cdot (\rho D_{km} \boldsymbol{\nabla} Y_k) + \{ \dot{\omega}_k \} \tag{2.14}$$

2.6 Convection–diffusion equations

$$\rho \frac{\partial \phi}{\partial t} + \rho \mathbf{v} \cdot \boldsymbol{\nabla} \phi = \boldsymbol{\nabla} \cdot (\Gamma_\phi \boldsymbol{\nabla} \phi) + \dot{s}_\phi \tag{2.15}$$

3 General numerical study

3.1 Assumptions

In order to solve the convection–diffusion equations numerically, we must make some assumptions which will simplify our study.

1. The location where the problem takes place is a closed connected set K contained in a bounded open connected set $\Omega \subset \mathbb{R}^m$, where $m = 1, 2, 3$ depends on the dimension of the problem. Both K and Ω are $\mathcal{C}^1$ or Lipschitz domains, allowing us to use vector calculus theorems.

2. The problem lasts for finite time, starting at time $t = 0$ and ending at time $t = t_{\max}$. Therefore the time interval is $I = (0, t_{\max}) \subset \mathbb{R}$.

3. The closed connected set K can be expressed as the union of finitely many closed sets $\mathcal{V}_1, \dots, \mathcal{V}_r$, that is, $K = \mathcal{V}_1 \cup \cdots \cup \mathcal{V}_r$. Moreover, these sets

The control volume centered at node P will be denoted by $\mathcal{V}_P$. Its boundary, known as the control surface, will be expressed as $\mathcal{S}_P = \partial \mathcal{V}_P$. The volume $\mathcal{V}_P$ occupies in $\mathbb{R}^m$

3.2 Spatial discretization

The type of problems addressed in this project occur in a bounded domain $\Omega \subset \mathbb{R}^m$ with $m = 2, 3$ depending on the case. In order to solve the problem numerically, a control–volume formulation is followed. This methodology discretizes the domain into nonoverlapping control volumes along with a grid of points named discretization nodes. The resulting discretized domain is named mesh or numerical grid [3].

There exist several types of grids according to the shape of control volumes and the ammount of subdivisions the domain has been partitioned into [4]:

- Structured (regular) grid:
- Block–structured grid:
- Unstructured grid:

Hereinafter, a structured regular grid approach will be followed. This formulation allows for two manners to discretize the domain, namely, cell–centered and node–centered discretizations. The former places discretization nodes over the domain and generates a control–volume centered on each node. The latter first generates the control–volumes and then places a node at the center of each one.

(a) A subfigure **(b)** A subfigure

Figure 3.1. A figure with two subfigures

3.3 Time discretization

3.4 Discretization of the continuity equation

As seen before, the continuity equation in differential form is

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) = 0 \quad (\mathbf{x}, t) \in \Omega \times I \tag{3.1}$$

Since the above relation is true on $\Omega \times I$, fixing one time $t \in I$ and integrating over a control volume $\mathcal{V}_P \subset \Omega$ gives

$$\int_{\mathcal{V}_P} \frac{\partial \rho}{\partial t} \, \mathrm{d}\mathbf{x} + \int_{\mathcal{V}_P} \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) \, \mathrm{d}\mathbf{x} = 0 \tag{3.2}$$

Let $\mathcal{S}_P = \partial \mathcal{V}_P$ be the control surface, i.e. the boundary of the control volume. Then applying the divergence theorem on the second term of equation (3.2),

$$\int_{\mathcal{V}_P} \frac{\partial \rho}{\partial t} \, \mathrm{d}\mathbf{x} + \int_{\mathcal{S}_P} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S = 0 \tag{3.3}$$

With the aim of simplifying the first term of (3.3), the average density of the control volume is defined in the following way:

$$\overline{\rho}_P = \frac{1}{V_P} \int_{\mathcal{V}_P} \rho \, \mathrm{d}\mathbf{x} \tag{3.4}$$

Introducing this relation in equation (3.3),

$$\frac{\mathrm{d}\overline{\rho}_P}{\mathrm{d}t} V_P + \int_{\mathcal{S}_P} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S = 0 \tag{3.5}$$

The mass flow term can be further simplified if a cartesian mesh is being used. In case of a 2D–mesh, the control surface can be partitioned into four different faces, namely, the east, west, north and south faces. Hence the control surface is $\mathcal{S}_P = \mathcal{S}_{Pe} \cup \mathcal{S}_{Pw} \cup \mathcal{S}_{Pn} \cup \mathcal{S}_{Ps}$ and the mass flow term can be expressed as

$$\begin{aligned} \int_{\mathcal{S}_P} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S &= \underbrace{\int_{\mathcal{S}_{Pe}} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S}_{\dot{m}_e} + \underbrace{\int_{\mathcal{S}_{Pw}} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S}_{-\dot{m}_w} + \underbrace{\int_{\mathcal{S}_{Pn}} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S}_{\dot{m}_n} + \underbrace{\int_{\mathcal{S}_{Ps}} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S}_{-\dot{m}_s} \\ &= \dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s \end{aligned} \tag{3.6}$$

Since evaluating each integral may be computationally expensive or impossible, the following approach is followed. Given a face f, the normal outer vector is constant on $\mathcal{S}_{Pf}$. Indeed, if $\mathbf{n}_f$ denotes the normal outer vector to face f, then $\mathbf{n}_e = \mathbf{i}$, $\mathbf{n}_w = -\mathbf{i}$, $\mathbf{n}_n = \mathbf{j}$ and $\mathbf{n}_s = -\mathbf{j}$. Since $\mathbf{v} = u\mathbf{i} + v\mathbf{j}$, the dot products are $\mathbf{v} \cdot \mathbf{n}_e = u$, $\mathbf{v} \cdot \mathbf{n}_w = -u$ and so on. Moreover, the integrand $(\rho \mathbf{v} \cdot \mathbf{n})_f$ can be approximated by the value each term takes at the face center, i.e.

$$(\rho \mathbf{v} \cdot \mathbf{n})_f \approx \rho_f (\mathbf{v} \cdot \mathbf{n})_f \tag{3.7}$$

Therefore the integral over $\mathcal{S}_{Pe}$ on equation (3.6) may be simplified as follows:

$$\int_{\mathcal{S}_{Pe}} \rho \mathbf{v} \cdot \mathbf{n} \, \mathrm{d}S \approx \int_{\mathcal{S}_{Pe}} (\rho \mathbf{v} \cdot \mathbf{n})_e \, \mathrm{d}S \approx \int_{\mathcal{S}_{Pe}} \rho_e (\mathbf{v} \cdot \mathbf{n})_e) \, \mathrm{d}S = \int_{\mathcal{S}_{Pe}} \rho_e u_e \, \mathrm{d}S = \rho_e u_e S_{Pe} =: \dot{m}_e \tag{3.8}$$

The same simplifications are applied to compute the remaining integral. Defining $\dot{m}_w$ and $\dot{m}_s$ as the negative integral makes the mass flow terms be positive in the positive coordinate direction. Introducing these in (3.5) yields

$$\frac{\mathrm{d}\overline{\rho}_P}{\mathrm{d}t} V_P + \dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s = 0 \tag{3.9}$$

The average density of the control volume is roughly the density at the discretization node, that is, $\overline{\rho}_P \approx \rho_P$. Integrating (3.9) over the time interval $[t^n, t^{n+1}]$ gives

$$V_P \int_{t^n}^{t^{n+1}} \frac{\mathrm{d}\rho_P}{\mathrm{d}t}\,\mathrm{d}t + \int_{t^n}^{t^{n+1}} (\dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s)\,\mathrm{d}t = 0 \tag{3.10}$$

The first term of (3.10) has a straightforward simplification applying a corollary of the fundamental theorem of calculus. Regarding the second term, numerical integration is used,

$$\begin{aligned}(\rho_P^{n+1} - \rho_P^n)V_P + \beta(\dot{m}_e^{n+1} - \dot{m}_w^{n+1} + \dot{m}_n^{n+1} - \dot{m}_s^{n+1})(t^{n+1} - t^n)\\ + (1-\beta)(\dot{m}_e^{n} - \dot{m}_w^{n} + \dot{m}_n^{n} - \dot{m}_s^{n})(t^{n+1} - t^n) = 0\end{aligned} \tag{3.11}$$

where $\beta \in \{0, \frac{1}{2}, 1\}$ depends on the chosen integration scheme. For the sake of simplicity, superindex $n+1$ shall be dropped and the time instant n will be denoted by the superindex 0. Assuming a uniform time step Δt, the resulting discretized continuity equation is

$$\frac{\rho_P - \rho_P^0}{\Delta t} V_P + \beta(\dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s) + (1-\beta)(\dot{m}_e^0 - \dot{m}_w^0 + \dot{m}_n^0 - \dot{m}_s^0) = 0 \tag{3.12}$$

Finally, when an implicit scheme is selected for the time integration,

$$\frac{\rho_P - \rho_P^0}{\Delta t} V_P + \dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s = 0 \tag{3.13}$$

If a 3D–mesh is being used, the contributions of top and bottom faces must be considered. The control surface is the union $\mathcal{S}_P = \mathcal{S}_{Pe} \cup \mathcal{S}_{Pw} \cup \mathcal{S}_{Pn} \cup \mathcal{S}_{Ps} \cup \mathcal{S}_{Pt} \cup \mathcal{S}_{Pb}$, hence equation (3.13) incorporates two new terms

$$\frac{\rho_P - \rho_P^0}{\Delta t} V_P + \dot{m}_e - \dot{m}_w + \dot{m}_n - \dot{m}_s + \dot{m}_t - \dot{m}_b = 0 \tag{3.14}$$

3.5 Discretization of the general convection–diffusion equation

The generalized convection–diffusion for a real valued function $\phi\colon \Omega \times I \subset \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ is

$$\frac{\partial(\rho\phi)}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\mathbf{v}\phi) = \boldsymbol{\nabla}\cdot(\Gamma_\phi \boldsymbol{\nabla}\phi) + \dot{s}_\phi, \quad (\mathbf{x}, t) \in \Omega \times I \tag{3.15}$$

whereas for a vector valued function $\phi = (\phi_1, \ldots, \phi_m)\colon \Omega \times I \subset \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^m$ it is written as

$$\frac{\partial(\rho\phi)}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\mathbf{v}\otimes\phi) = \boldsymbol{\nabla}\cdot(\Gamma_\phi \boldsymbol{\nabla}\phi) + \dot{s}_\phi, \quad (\mathbf{x}, t) \in \Omega \times I \tag{3.16}$$

where $\otimes$ denotes the outer product of $\mathbf{v}\colon \Omega \times I \subset \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^m$ and ϕ, which is a $m \times m$ matrix. Since the generalized convection–diffusion equation for a vector valued function actually comprises m equations, one for each component function, only the discretization for a real valued function will be studied.

Integrating (3.15) over $\mathcal{V}_P \times [t^n, t^{n+1}] \subset \Omega \times I$ and using Fubini's theorem

$$\begin{aligned}\int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \boldsymbol{\nabla}\cdot(\rho\mathbf{v}\phi)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t =\\ = \int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \boldsymbol{\nabla}\cdot(\Gamma_\phi\boldsymbol{\nabla}\phi)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \dot{s}_\phi\,\mathrm{d}\mathbf{x}\,\mathrm{d}t\end{aligned} \tag{3.17}$$

The simplification of the first term is analogous to that of the continuity equation. The average value of $\rho\phi$ on $\mathcal{V}_P$ at time t is defined by

$$(\rho\phi)_P = \frac{1}{V_P}\int_{\mathcal{V}_P} \rho\phi \,\mathrm{d}\mathbf{x} \tag{3.18}$$

although the following approximation is needed:

$$(\rho\phi)_P \approx \rho_P \phi_P \tag{3.19}$$

Then the transient term is:

$$\int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}_P} \rho\phi\,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \frac{\mathrm{d}(\rho\phi)_P}{\mathrm{d}t} V_P\,\mathrm{d}t \approx \left\{\rho_P\phi_P - \rho_P^0\phi_P^0\right\} V_P \tag{3.20}$$

Divergence theorem must be applied to simplify the convective contribution,

$$\int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \boldsymbol{\nabla}\cdot(\rho\mathbf{v}\phi)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \int_{\mathcal{S}_P} \rho\phi\mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \sum_i \int_{\mathcal{S}_{Pi}} \rho\phi\mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t \tag{3.21}$$

The value that ϕ takes on $\mathcal{S}_{Pi}$ can be approximated by its value at a representative point, for instance, the point at face center, that is to say, $\phi \approx \phi_i$. Therefore,

$$\begin{aligned}\int_{t^n}^{t^{n+1}} \sum_i \int_{\mathcal{S}_i} \rho\phi\mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t &\approx \int_{t^n}^{t^{n+1}} \sum_i \int_{\mathcal{S}_i} \rho\phi_i\mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \sum_i \dot{m}_i\phi_i\,\mathrm{d}t = \\ &= \left\{\beta\sum_i \dot{m}_i\phi_i + (1-\beta)\sum_i \dot{m}_i^0\phi_i^0\right\}\Delta t\end{aligned} \tag{3.22}$$

For the third term,

$$\int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_P} \boldsymbol{\nabla}\cdot(\Gamma_\phi\boldsymbol{\nabla}\phi)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \int_{\mathcal{S}_P} \Gamma_\phi\boldsymbol{\nabla}\phi\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \sum_i \int_{\mathcal{S}_{Pi}} \Gamma_\phi\boldsymbol{\nabla}\phi\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t \tag{3.23}$$

Since a cartesian mesh is being used, the outer normal vector to the face $\mathcal{S}_{Pi}$ is constant and points in the direction of some coordinate axis. Hence the dot product $\boldsymbol{\nabla}\phi\cdot\mathbf{n}$ in the face $\mathcal{S}_{Pi}$ equals the partial derivative with respect to x_i times ±1, depending on the direction of $\mathbf{n}$. For east, north and top faces the sign is positive, whilst for west, south and bottom faces the sign is negative. Again, Γ_ϕ will be approximated by the value at the face center, and partial derivatives will be approximated by a finite centered difference. In order to simplify the notation, subindex ϕ in the diffusion coefficient Γ_ϕ will be dropped. For short, the following magnitudes are defined

$$D_i = \frac{\Gamma_i S_i}{d_{PI}} \tag{3.24}$$

$$D_i^0 = \frac{\Gamma_i^0 S_i}{d_{PI}} \tag{3.25}$$

where i and I refer to the face letter. For a 2D–mesh, equation (3.23) results in

$$\begin{aligned}&\int_{t^n}^{t^{n+1}} \sum_i \int_{\mathcal{S}_{Pi}} \Gamma_\phi\boldsymbol{\nabla}\phi\cdot\mathbf{n}\,\mathrm{d}S\,\mathrm{d}t \approx \\ &\approx \int_{t^n}^{t^{n+1}} \Big\{D_i(\phi_E-\phi_P) - D_w(\phi_P-\phi_W) + D_n(\phi_N-\phi_P) - D_s(\phi_P-\phi_S)\Big\}\,\mathrm{d}t \approx \\ &\approx \beta\left\{D_e(\phi_E-\phi_P) - D_w(\phi_P-\phi_W) + D_n(\phi_N-\phi_P) - D_s(\phi_P-\phi_S)\right\}\Delta t + \\ &+ (1-\beta)\Big\{D_e^0(\phi_E-\phi_P) - D_w^0(\phi_P-\phi_W) + D_n^0(\phi_N-\phi_P) - D_s^0(\phi_P-\phi_S)\Big\}\Delta t\end{aligned} \tag{3.26}$$

In the case of a 3D–mesh, the contributions of top and bottom faces must be accounted for.

In order to discretize the fourth term, the mean value of the source in $\mathcal{V}_P$ at time t is by

$$\bar{\dot{s}}_\phi = \frac{1}{V_P}\int_{\mathcal{V}_P} \dot{s}_\phi \,\mathrm{d}\mathbf{x} \tag{3.27}$$

If the value of s_ϕ is known, the relation $\bar{\dot{s}}_\phi = \dot{\bar{s}}_\phi$ is true. Indeed,

$$\dot{\bar{s}}_\phi = \frac{\mathrm{d}}{\mathrm{d}t}\bar{s}_\phi = \frac{1}{V_P}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{V}_P} s_\phi \,\mathrm{d}\mathbf{x} = \frac{1}{V_P}\int_{\mathcal{V}_P} \dot{s}_\phi \,\mathrm{d}\mathbf{x} = \bar{\dot{s}}_\phi \tag{3.28}$$

In most cases, the dependence of $\dot{\bar{s}}_\phi$ on ϕ is complicated. Since the equations obtained until now are linear, the relation between the source term and the variable would ideally be linear. This linearity is imposed as follows

$$\dot{\bar{s}}_\phi = S_C^\phi + S_P^\phi \phi_P \tag{3.29}$$

where the values of S_C^ϕ and S_P^ϕ may vary with ϕ [3]. Making use of these relations, the source term integral is discretized as

$$\int_{t^n}^{t^{n+1}}\int_{\mathcal{V}_P} \dot{s}_\phi \,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t^n}^{t^{n+1}} \dot{\bar{s}}_{\phi P} V_P \Delta t = \left(S_C^\phi + S_P^\phi \phi_P\right) V_P \Delta t \tag{3.30}$$

As shall be discussed later, S_P^ϕ must be non–positive.

Finally, the discretization of the 2D generalized convection–diffusion equation is

$$\begin{aligned}
&\frac{\rho_P\phi_P - \rho_P^0\phi_P^0}{\Delta t}V_P + \\
&+\beta\Big(\dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s\Big) + (1-\beta)\left(\dot{m}_e^0\phi_e^0 - \dot{m}_w^0\phi_w^0 + \dot{m}_n^0\phi_n^0 - \dot{m}_s^0\phi_s^0\right) = \\
&= \beta\Big\{D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) - D_s(\phi_P - \phi_S)\Big\} + \\
&+ (1-\beta)\Big\{D_e^0(\phi_E^0 - \phi_P^0) - D_w^0(\phi_P^0 - \phi_W^0) + D_n^0(\phi_N^0 - \phi_P^0) - D_s^0(\phi_P^0 - \phi_S^0)\Big\} + \\
&+ \left(S_C^\phi + S_P^\phi\phi_P\right) V_P
\end{aligned} \tag{3.31}$$

In the case of a implicit integration scheme, that is, $\beta = 1$, equation (3.31) is simplified to:

$$\begin{aligned}
&\frac{\rho_P\phi_P - \rho_P^0\phi_P^0}{\Delta t}V_P + \dot{m}_e\phi_e - \dot{m}_w\phi_w + \dot{m}_n\phi_n - \dot{m}_s\phi_s = \\
&\quad = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) - D_s(\phi_P - \phi_S) + (S_C^\phi + S_P^\phi\phi_P)V_P
\end{aligned} \tag{3.32}$$

An equivalent and more useful form of the discretization equation can be found by multiplying (3.13) times ϕ_P and subtracting it from (3.32), which results in

$$\begin{aligned}
&\rho_P^0\frac{\phi_P - \phi_P^0}{\Delta t}V_P + \dot{m}_e(\phi_e - \phi_P) - \dot{m}_w(\phi_w - \phi_P) + \dot{m}_n(\phi_n - \phi_P) - \dot{m}_s(\phi_s - \phi_P) = \\
&\quad = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) - D_s(\phi_P - \phi_S) + (S_C^\phi + S_P^\phi\phi_P)V_P
\end{aligned} \tag{3.33}$$

The 3D analog of (3.33) includes the top and bottom faces contributions:

$$\begin{aligned}
\rho_P^0\frac{\phi_P - \phi_P^0}{\Delta t}V_P &+ \dot{m}_e(\phi_e - \phi_P) - \dot{m}_w(\phi_w - \phi_P) + \dot{m}_n(\phi_n - \phi_P) \\
&- \dot{m}_s(\phi_s - \phi_P) + \dot{m}_t(\phi_t - \phi_P) - \dot{m}_b(\phi_b - \phi_P) \\
&= D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) \\
&- D_s(\phi_P - \phi_S) + D_t(\phi_T - \phi_P) - D_b(\phi_P - \phi_B) + (S_C^\phi + S_P^\phi\phi_P)V_P
\end{aligned} \tag{3.34}$$

3.6 Evaluation of the convective terms

The discretized version of the generalized convection–diffusion equation requires the values of the magnitude ϕ at points different from the nodes. In this section several methods to compute ϕ at faces are given. The values of ρ and Γ will be assumed to be known at the nodal points. For the sake of simplicity, east face will be taken as reference. The generalization to the remaining faces is straightforward.

3.6.1 Upwind–Difference Scheme (UDS)

Incompressible flows and gases at low Mach number are more influenced by upstream conditions than downstream conditions. Let $(\mathbf{v}\cdot\mathbf{n})_e$ denote the value of the dot product $\mathbf{v}\cdot\mathbf{n}$ at east face $\mathcal{S}_{Pe}$. If $(\mathbf{v}\cdot\mathbf{n})_e > 0$, fluid flows from node P to node E, hence P is the upstream node and E is the downstream node. Conversely, if $(\mathbf{v}\cdot\mathbf{n})_e < 0$, nodes interchange their roles as fluid flows from node E to node P. This situation is pictured in figures 3.2 and 3.3.

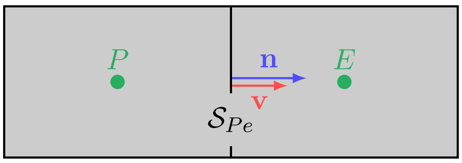


Figure 3.2. Since $(\mathbf{v}\cdot\mathbf{n})_e > 0$ fluid flows from node P (upstream node) to node E (downstream node).

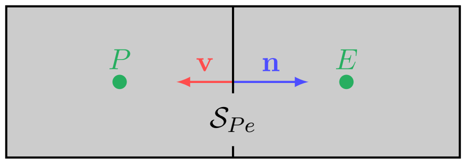


Figure 3.3. Since $(\mathbf{v}\cdot\mathbf{n})_e < 0$ fluid flows from node E (upstream node) to node P (downstream node).

If $(\mathbf{v}\cdot\mathbf{n})_e = 0$, it implies $\mathbf{v}_e$ lies in the orthogonal subspace to the vector space generated by $\mathbf{n}$. As a result, given the approximations taken, there is no fluid flow through face $\mathcal{S}_{Pe}$.

The Upwind–Difference Scheme assigns ϕ_e the value of ϕ at the upstream node, that is,

$$\phi_e = \begin{cases} \phi_P & \text{if } (\mathbf{v}\cdot\mathbf{n})_e > 0 \\ \phi_E & \text{if } (\mathbf{v}\cdot\mathbf{n})_e < 0 \end{cases} \tag{3.35}$$

The scheme is summarized in figures 3.4 and 3.5.

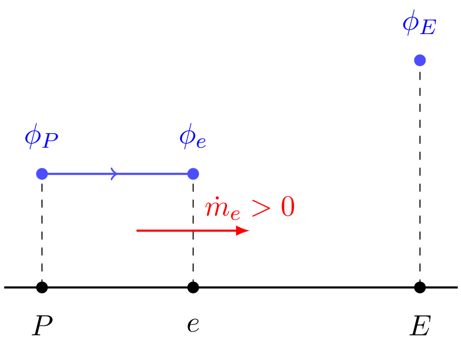


Figure 3.4. UDS when $(\mathbf{v}\cdot\mathbf{n})_e > 0$.

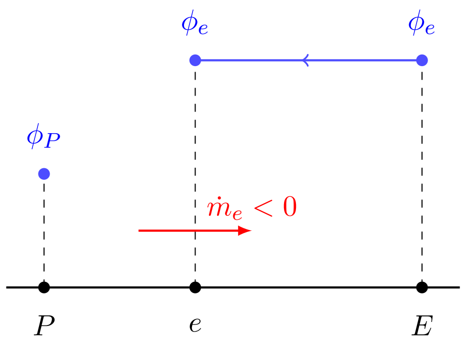


Figure 3.5. UDS when $(\mathbf{v}\cdot\mathbf{n})_e < 0$.

Equation (3.35) can be expressed in a more compact fashion as follows,

$$\dot{m}_e(\phi_e - \phi_P) = \frac{\dot{m}_e - |\dot{m}_e|}{2}(\phi_E - \phi_P) \tag{3.36}$$

since the approximation to compute $\dot{m}_e$ is related to $(\mathbf{v}\cdot\mathbf{n})_e$ through the relation $\dot{m}_e = (\mathbf{v}\cdot\mathbf{n})_e S_{Pe}$. The extension of (3.36) to the remaining faces is the following:

$$\dot{m}_w(\phi_w - \phi_P) = \frac{\dot{m}_w + |\dot{m}_w|}{2}(\phi_W - \phi_P) \tag{3.37}$$

$$\dot{m}_n(\phi_n - \phi_P) = \frac{\dot{m}_n - |\dot{m}_n|}{2}(\phi_N - \phi_P) \tag{3.38}$$

$$\dot{m}_s(\phi_s - \phi_P) = \frac{\dot{m}_s + |\dot{m}_s|}{2}(\phi_S - \phi_P) \tag{3.39}$$

UDS is a stable scheme, however it suffers from numerical diffusion. Indeed, assuming the upstream node is P, expanding ϕ about point x_P in its Taylor expansion up to 2$^{\text{nd}}$ degree and using Lagrange's remainder,

$$\phi_e = \phi_P + \left(\frac{\partial \phi}{\partial x}\right)_P d_{Pe} + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_1} \frac{d_{Pe}^2}{2} \tag{3.40}$$

it is apparent that UDS retains the first term on the left–hand side of (3.40). As a consequence, the error highest order is $(\partial_x\phi)_P d_{Pe}$, which is proportional to the distance between P and the face $\mathcal{S}_{Pe}$. This term resembles to a diffusion flux given, for instance, by Fourier's or Fick's laws of diffusion. The same result is obtained when E is the upstream node,

$$\phi_e = \phi_E - \left(\frac{\partial \phi}{\partial x}\right)_E d_{Ee} + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_2} \frac{d_{Ee}^2}{2} \tag{3.41}$$

whence it can be deduced that the error is bounded by $\max\{|(\partial_x\phi)_E d_{Pe}|, |(\partial_x\phi)_E d_{Ee}|\}$. The numerical diffusion issue is magnified in multidimensional problems, where peaks of rapid variation can be obtained, hence very fine grids are required.

3.6.2 Central–Difference Scheme (CDS)

The Central–Difference Scheme assumes a linear distribution for ϕ as illustrated in figure 3.6.

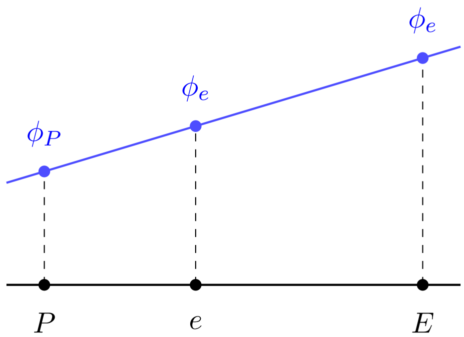


Figure 3.6. Central Difference Scheme (CDS).

Thereby ϕ_e can be obtained interpolating between ϕ_P and ϕ_E,

$$\phi_e - \phi_P = \frac{d_{Pe}}{d_{PE}}(\phi_E - \phi_P) \tag{3.42}$$

as well as the remaining faces values,

$$\phi_w - \phi_P = \frac{d_{Pw}}{d_{PW}}(\phi_W - \phi_P) \tag{3.43}$$

$$\phi_n - \phi_P = \frac{d_{Pn}}{d_{PN}}(\phi_N - \phi_P) \tag{3.44}$$

$$\phi_s - \phi_P = \frac{d_{Ps}}{d_{PS}}(\phi_S - \phi_P) \tag{3.45}$$

This yields a 2$^{\text{nd}}$ order approximation for ϕ_e if $d_{Pe} = d_{Ee}$. In effect, applying Taylor's theorem about point x_e,

$$\phi_P = \phi_e - \left(\frac{\partial \phi}{\partial x}\right)_e d_{Pe} + \frac{1}{2}\left(\frac{\partial^2 \phi}{\partial x^2}\right)_e d_{Pe}^2 + \frac{1}{6}\left(\frac{\partial^3 \phi}{\partial x^3}\right)_{\xi_1} d_{Pe}^3 \tag{3.46}$$

The 2$^{\text{nd}}$ order approximation of $(\partial_x \phi)_e$ is given by

$$\left(\frac{\partial \phi}{\partial x}\right)_e = \frac{\phi_E - \phi_P}{d_{PE}} - \left(\frac{\partial^3 \phi}{\partial x^3}\right)_{\xi_2} \frac{d_{PE}^2}{3!} = \frac{\phi_E - \phi_P}{d_{PE}} - \left(\frac{\partial^3 \phi}{\partial x^3}\right)_{\xi_2} \frac{(d_{Pe} + d_{Ee})^2}{3!} \tag{3.47}$$

Introducing (3.47) in (3.46) and imposing $d_{Pe} = d_{Ee}$,

$$\phi_e - \phi_P = \frac{d_{Pe}}{d_{PE}}(\phi_E - \phi_P) - \left(\frac{\partial^2 \phi}{\partial x^2}\right)_e \frac{d_{Pe}^2}{2} - \left\{\left(\frac{\partial^3 \phi}{\partial x^3}\right)_{\xi_1} + 4\left(\frac{\partial^3 \phi}{\partial x^3}\right)_{\xi_2}\right\} \frac{d_{Pe}^3}{6} \tag{3.48}$$

As CDS retains the first term on the left–hand side of (3.48), the highest order term of the error is $\frac{1}{2}(\partial_x^2 \phi)_e d_{Pe}^2$, proving that CDS provides a 2$^{\text{nd}}$ order approximation of ϕ_e when $d_{Pe} = d_{Ee}$. Nonetheless, this scheme is prone to stability problems producing oscillatory outputs since the approximation is of order higher than 1.

3.6.3 Exponential–Difference Scheme (EDS)

The exponential difference scheme assumes a distribution for ϕ based on the steady 2–dimensional generalized convection–diffusion equation with no source term, that is to say,

$$\frac{\mathrm{d}}{\mathrm{d}x}(\rho u \phi) = \frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma \frac{\mathrm{d}\phi}{\mathrm{d}x}\right) \tag{3.49}$$

where u is the component of $\mathbf{v}$ in the x direction. So as to ease the study, ρu and Γ are assumed to be constant. Thereby the initial value problem obtained is

$$\begin{cases} \dfrac{\mathrm{d}^2\phi}{\phi x^2} - \dfrac{\rho u}{\Gamma}\dfrac{\mathrm{d}\phi}{\mathrm{d}x} = 0 \quad \text{in } (x_P, x_E) \subset \mathbb{R} \\ \phi(x_P) = \phi_P \\ \phi(x_E) = \phi_E \end{cases} \tag{3.50}$$

Since the initial value problem (3.50) is a second order linear ODE with two boundary conditions, its solutions exists, is unique, and is given by

$$\phi(x) = \phi_P + \frac{e^{\frac{\rho u}{\Gamma}(x - x_P)} - 1}{e^{\frac{\rho u}{\Gamma} d_{PE}} - 1}(\phi_E - \phi_P) \tag{3.51}$$

Péclet's number for heat transfer is defined as the following ratio,

$$\mathrm{Pe} = \frac{\text{convection transport}}{\text{heat transport}} = \frac{\rho u L}{\lambda / c_p} \tag{3.52}$$

where L is a characteristic length of the problem. Since λ/c_p is the diffusion coefficient in equation (2.13), it can be substituted by the diffusion coefficient Γ of the generalized convection–diffusion equation, providing a new definition for Péclet's number

$$\mathrm{Pe} = \frac{\rho u L}{\Gamma} \tag{3.53}$$

Taking d_{PE} as characteristic length and evaluating (3.51) at $x = x_e$, the approximation of ϕ_e given by EDS in terms of Péclet's number is written as

$$\phi_e - \phi_P = \frac{e^{\mathrm{Pe}_e \frac{d_{Pe}}{d_{PE}}} - 1}{e^{\mathrm{Pe}_e} - 1}(\phi_E - \phi_P) \tag{3.54}$$

The extension of (3.54) to the face f is done by taking d_{PF} as characteristic length, that is, if $f = w$, then the characteristic length is d_{PW}. Thereby EDS gives the following face values:

$$\phi_w = \left(1 - \frac{e^{\mathrm{Pe}_w \frac{d_{Ww}}{d_{PW}}} - 1}{e^{\mathrm{Pe}_w} - 1}\right)\phi_W + \frac{e^{\mathrm{Pe}_w \frac{d_{Ww}}{d_{PW}}} - 1}{e^{\mathrm{Pe}_w} - 1}\phi_P \tag{3.55}$$

$$\phi_n - \phi_P = \frac{e^{\mathrm{Pe}_n \frac{d_{Pn}}{d_{PN}}} - 1}{e^{\mathrm{Pe}_n} - 1}(\phi_N - \phi_P) \tag{3.56}$$

$$\phi_s = \left(1 - \frac{e^{\mathrm{Pe}_s \frac{d_{Ss}}{d_{PS}}} - 1}{e^{\mathrm{Pe}_s} - 1}\right)\phi_S + \frac{e^{\mathrm{Pe}_s \frac{d_{Ss}}{d_{PS}}} - 1}{e^{\mathrm{Pe}_s} - 1}\phi_P \tag{3.57}$$

3.6.4 Second–order Upwind Linear Extrapolation (SUDS)

As previously mentioned, incompressible flows and fluids at low Mach number are more influenced by upstream condition than by downstream conditions. In order to account for this fact and to ease the study, a new notation is introduced. Located at the face separating two control volumes, f refers to the face, D is the downstream node, C is the first upstream node and U is the most upstream node. Some books may use U and UU instead of C and U, respectively.

The Second–order Upwind Linear Extrapolation scheme takes profit of this idea since it extrapolates ϕ_e using a straight line between the values of ϕ at nodes C and U. The two possible situations are pictured in figures 3.7 and 3.8.

On the one hand, when $(\mathbf{v} \cdot \mathbf{n})_e > 0$, the line between points (x_W, ϕ_W) and (x_P, ϕ_P) is given by

$$\phi(x) = \phi_W + \frac{\phi_P - \phi_W}{d_{PW}}(x - x_W) \tag{3.58}$$

and substituting at $x = x_e$, the formula for ϕ_e is obtained:

$$\phi_e = \phi_W + \frac{\phi_P - \phi_W}{d_{PW}}(x_e - x_W) = \phi_P + \frac{d_{Pe}}{d_{PW}}(\phi_P - \phi_W) \tag{3.59}$$

On the other hand, in the case of $(\mathbf{v} \cdot \mathbf{n})_e < 0$, the line between points (x_E, ϕ_E) and (x_{EE}, ϕ_{EE}) is

$$\phi(x) = \phi_E + \frac{\phi_{EE} - \phi_E}{d_{E,EE}}(x - x_E) \tag{3.60}$$

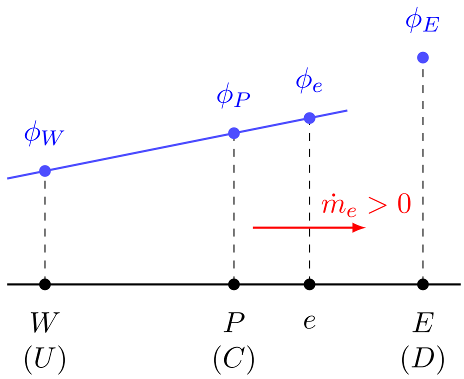


Figure 3.7. SUDS when $(\mathbf{v} \cdot \mathbf{n})_e > 0$.

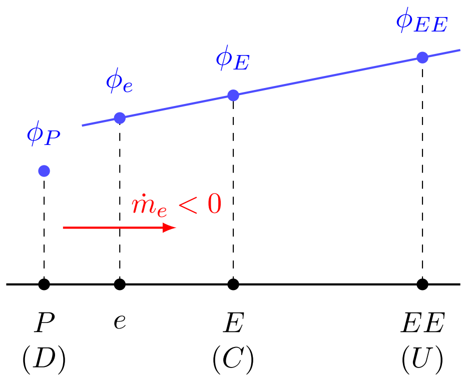


Figure 3.8. SUDS when $(\mathbf{v} \cdot \mathbf{n})_e < 0$.

and the approximation of ϕ_e is

$$\phi_e = \phi_E + \frac{\phi_{EE} - \phi_E}{d_{E,EE}}(x_e - x_E) = \phi_E + \frac{d_{Ee}}{d_{E,EE}}(\phi_E - \phi_{EE}) \tag{3.61}$$

Using the DCU notation, (3.59) and (3.61) are both rewritten in the following manner:

$$\phi_f - \phi_C = \frac{d_{Cf}}{d_{CU}}(\phi_C - \phi_U) \tag{3.62}$$

In order to prove that SUDS is a second order scheme when a locally uniform mesh is used and $(\mathbf{v} \cdot \mathbf{n})_e > 0$, consider the Taylor expansion up to 2^{nd} degree of ϕ about point x_W,

$$\phi_e = \phi_W + \left(\frac{\partial \phi}{\partial x}\right)_W d_{We} + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_1} \frac{d_{We}^2}{2} \tag{3.63}$$

The first derivative of ϕ with respect to x can be replaced by its first order approximation, namely,

$$\left(\frac{\partial \phi}{\partial x}\right)_W = \frac{\phi_P - \phi_W}{d_{PW}} - \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_2} \frac{d_{PW}}{2} \tag{3.64}$$

thereby,

$$\begin{aligned}
\phi_e &= \phi_W + \frac{d_{We}}{d_{PW}}(\phi_P - \phi_W) + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_1} \frac{d_{We}^2}{2} - \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_2} \frac{d_{We} d_{PW}}{2} \\
&= \phi_P + \frac{d_{Pe}}{d_{PW}}(\phi_P - \phi_W) + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_1} \frac{(d_{PW} + d_{Pe})^2}{2} - \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_2} \frac{(d_{PW} + d_{Pe}) d_{PW}}{2}
\end{aligned} \tag{3.65}$$

The scheme retains the two first terms on the right–hand side of (3.65), therefore the error is composed by the last two terms. The uniform mesh hypothesis implies $d_{PW} = 2d_{Pe} = L$, therefore the error term is multiplied by L^2,

$$\phi_e = \phi_P + \frac{d_{Pe}}{d_{PW}}(\phi_P - \phi_W) + \frac{3L^2}{4}\left\{3\left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_1} - \left(\frac{\partial^2 \phi}{\partial x^2}\right)_{\xi_2}\right\} \tag{3.66}$$

whence the second order of SUDS is deduced. The proof in the case of $(\mathbf{v} \cdot \mathbf{n})_e < 0$ is analogous.

3.6.5 Quadratic Upwind Interpolation for Convective Kinematics (QUICK)

A logical improvement of CDS is using a parabola to interpolate between nodal points rather than a straight line. To construct a parabola three points are needed. As aforementioned, upstream conditions have a greater influence on flow properties than downstream conditions for incompressible flows and low Mach number gases. QUICK scheme takes profit of this fact.

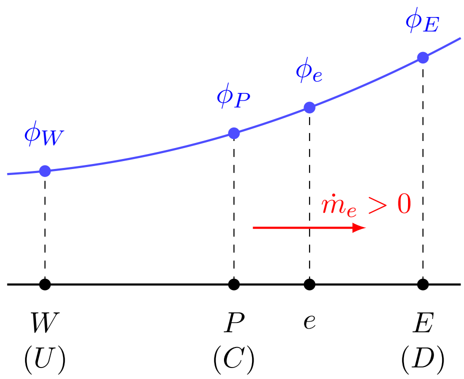


Figure 3.9. QUICK when $(\mathbf{v} \cdot \mathbf{n})_e > 0$.

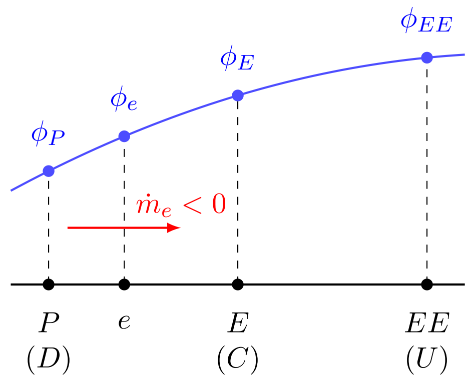


Figure 3.10. QUICK when $(\mathbf{v} \cdot \mathbf{n})_e < 0$.

Let (x_0, ϕ_0), (x_1, ϕ_1), (x_2, ϕ_2) be the points which the polynomial $p(x)$ must interpolate, that is, $p(x_0) = \phi_0$, $p(x_1) = \phi_1$ and $p(x_2) = \phi_2$, satisfying $x_0 < x_1 < x_2$. If $(\mathbf{v} \cdot \mathbf{n})_e > 0$ then $x_0 = x_W$, $x_1 = x_P$ and $x_2 = x_E$, whereas $x_0 = x_P$, $x_1 = x_E$ and $x_2 = x_{EE}$ in case of $(\mathbf{v} \cdot \mathbf{n})_e < 0$. Let $p(x)$ be the following polynomial

$$p(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1), \quad a_0, a_1, a_2 \in \mathbb{R} \tag{3.67}$$

Since the interpolating polynomial exists and is unique referencia, by imposing the interpolating condition, $p(x)$ will be the desired polynomial. The interpolating condition is,

$$\left.\begin{aligned} p(x_0) &= a_0 = \phi_0 \\ p(x_1) &= a_0 + a_1(x_1 - x_0) = \phi_1 \\ p(x_2) &= a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = \phi_2 \end{aligned}\right\} \tag{3.68}$$

which yields the following linear system:

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & x_1 - x_0 & 0 \\ 1 & x_2 - x_0 & (x_2 - x_1)(x_2 - x_0) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \end{pmatrix} \tag{3.69}$$

The determinant of the system matrix is non-zero because the abscissae are distinct, therefore the solution is given by

$$\left.\begin{aligned} a_0 &= \phi_0 \\ a_1 &= \frac{\phi_1 - \phi_0}{x_1 - x_0} \\ a_2 &= \frac{\phi_2 - \phi_0}{(x_2 - x_1)(x_2 - x_0)} - \frac{\phi_1 - \phi_0}{(x_2 - x_1)(x_1 - x_0)} \end{aligned}\right\} \tag{3.70}$$

and the polynomial is

$$p(x) = \phi_0 - \frac{(x - x_2)(x - x_0)}{(x_2 - x_1)(x_1 - x_0)}(\phi_1 - \phi_0) + \frac{(x - x_1)(x - x_0)}{(x_2 - x_1)(x_2 - x_0)}(\phi_2 - \phi_0) \tag{3.71}$$

$$\phi(x) = \phi_U - \frac{(x - x_D)(x - x_U)}{(x_D - x_C)(x_C - x_U)}(\phi_C - \phi_U) + \frac{(x - x_C)(x - x_U)}{(x_D - x_C)(x_D - x_U)}(\phi_D - \phi_U) \tag{3.72}$$

$$\phi(x) = \phi_D - \frac{(x - x_U)(x - x_D)}{(x_U - x_C)(x_C - x_D)}(\phi_C - \phi_D) + \frac{(x - x_C)(x - x_D)}{(x_U - x_C)(x_U - x_D)}(\phi_U - \phi_D) \tag{3.73}$$

$$\tag{3.74}$$

Assuming a uniform grid, i.e. $x_1 - x_0 = x_2 - x_1 = L$ and the face f located at the midpoint between nodal points, the approximation of ϕ_e given by QUICK scheme is

$$\phi_e = -\frac{1}{8}\phi_0 + \frac{6}{8}\phi_1 + \frac{3}{8}\phi_2 \tag{3.75}$$

and depending on the sign of $(\mathbf{v} \cdot \mathbf{n})_e$,

$$\phi_e = \begin{cases} -\frac{1}{8}\phi_U + \frac{6}{8}\phi_C + \frac{3}{8}\phi_D & \text{if} \quad (\mathbf{v} \cdot \mathbf{n})_e > 0 \\ -\frac{1}{8}\phi_D + \frac{6}{8}\phi_C + \frac{3}{8}\phi_U & \text{if} \quad (\mathbf{v} \cdot \mathbf{n})_e < 0 \end{cases} \tag{3.76}$$

The output (3.76) provided by QUICK scheme is second–order accurate.

3.6.6 Normalization of variables

Owing to numerical reasons, it is convenient to normalize spatial and convective variables, that is to say, define new variables which take a rather small range of values. This is accomplished using the *DCU* notation and defining

$$\hat{x} = \frac{x - x_U}{x_D - x_U}$$
$$\hat{\phi} = \frac{\phi - \phi_U}{\phi_D - \phi_U}$$

Of course, $(\hat{x}_U, \hat{\phi}_U) = (0, 0)$, $(\hat{x}_D, \hat{\phi}_D) = (1, 1)$ and $\hat{x}_C, \hat{x}_f \in [0, 1]$. However, $\hat{\phi}$ is not necessarily in $[0, 1]$ for all $x \in [0, 1]$, nor does it have to be an increasing function. These situations are represented in figures 3.11 and 3.12.

The normalized variable $\hat{\phi}_f$ can be computed directly as shown in section referencia sección posterior and, based on this, the variable at face,

$$\phi_f = \phi_U + \hat{\phi}_f(\phi_D - \phi_U) \tag{3.77}$$

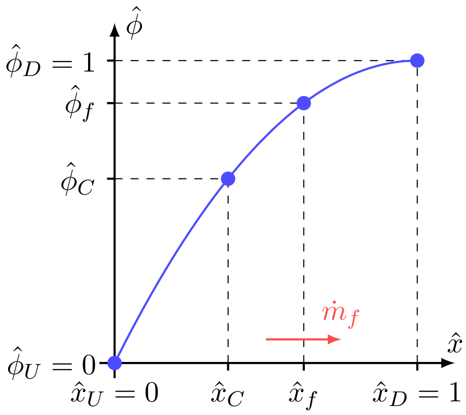


Figure 3.11. Scheme of normalized variables when $\hat{\phi}(x)$ is a strictly increasing function.

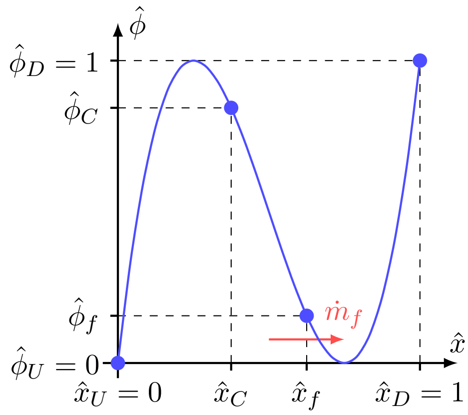


Figure 3.12. Scheme of normalized variables when $\hat{\phi}(x)$ is not a strictly increasing function.

3.6.7 Sharp and Monotonic Algorithm for Realistic Transport (SMART)

As aforementioned, schemes whose order is higher than one might be unstable, producing oscillatory outputs for the convective variables. For instance, CDS, SUDS and QUICK are not bounded schemes. The conditions for stability and accuracy are formulated in [5]:

(i) $\hat{\phi}_f$ must be a continuous function of $\hat{\phi}_C$.

(ii) If $\hat{\phi}_C = 0$, then $\hat{\phi}_f = 0$.

(iii) If $\hat{\phi}_C = 1$, then $\hat{\phi}_f = 1$.

(iv) If $0 < \hat{\phi}_f < 1$, then $\hat{\phi}_C < \hat{\phi}_f < 1$.

Conditions (i) through (iv) are represented in figure 3.13. A bounded convective scheme must output results lying within the shadowed region.

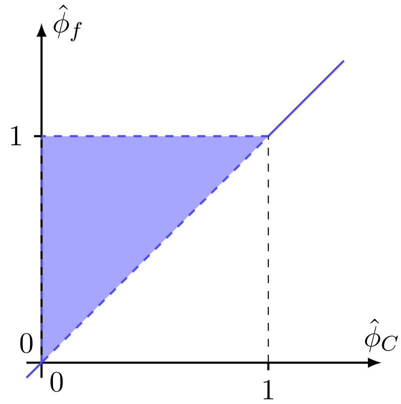


Figure 3.13. High–order bounded convection schemes conditions for stability.

The SMART scheme (Sharp and Monotonic Algorithm for Realistic Transport) is a bounded convective scheme [5], given by:

$$\hat{\phi}_f = \begin{cases} -\dfrac{\hat{x}_f(1-3\hat{x}_C+2\hat{x}_f)}{\hat{x}_C(\hat{x}_C-1)}\hat{\phi}_C & \text{if} \quad 0 < \hat{\phi}_C < \dfrac{\hat{x}_C}{3} \\ \dfrac{\hat{x}_f(\hat{x}_f-\hat{x}_C)}{1-\hat{x}_C} + \dfrac{\hat{x}_f(\hat{x}_f-1)}{\hat{x}_C(\hat{x}_C-1)}\hat{\phi}_C & \text{if} \quad \dfrac{\hat{x}_C}{3} < \hat{\phi}_C < \dfrac{\hat{x}_C(1+\hat{x}_f-\hat{x}_C)}{\hat{x}_f} \\ 1 & \text{if} \quad \dfrac{\hat{x}_C(1+\hat{x}_f-\hat{x}_C)}{\hat{x}_f} < \hat{\phi}_C < 1 \\ \hat{\phi}_C & \text{otherwise} \end{cases} \tag{3.78}$$

3.7 Final form of the generalized convection–diffusion equation

The purpose of this subsection is to obtain a discretization equation of the form

$$\mathcal{A}_P\phi_P + \sum_I \mathcal{A}_I\phi_I = \mathcal{Q}_P \tag{3.79}$$

so that it can be easily implemented to be solved numerically, starting from equation (3.32) and the studied schemes to evaluate convective properties. Among the revised schemes, some use only the surrounding nodes, whilst others involve a larger amount of nodes. As a consequence of the different treatment needed, separate subsections are devoted to each type of scheme.

3.7.1 Small molecule schemes

Small molecule schemes are those which only involve adjacent nodes to the volume faces. That is, index I in (3.79) refers to nodes E, W, N and S. As a result, these schemes can be introduced in a compact form as Patankar suggests [3]. However a different approach will be followed here.

As it can be noted from the expressions for schemes UDS (equations (3.36) to (3.39)), CDS (equations (3.42) to (3.45)) and EDS (equations (3.54) to (3.57)), once a face f is chosen, the formula for ϕ_f has the following form

$$\phi_f - \phi_P = A_f(\phi_F - \phi_P) \tag{3.80}$$

where A_f is a coefficient which depends on the face and the scheme. Particularizing for each face,

$$\phi_e - \phi_P = A_e(\phi_E - \phi_P) \tag{3.81}$$
$$\phi_w - \phi_P = A_w(\phi_W - \phi_P) \tag{3.82}$$
$$\phi_n - \phi_P = A_n(\phi_N - \phi_P) \tag{3.83}$$
$$\phi_s - \phi_P = A_s(\phi_S - \phi_P) \tag{3.84}$$

The values of A_f and B_f for each face and scheme are collected in table 3.1.

Face	UDS	CDS	EDS
East	$\dfrac{\dot{m}_e - \lvert\dot{m}_e\rvert}{2\dot{m}_e}$	$\dfrac{d_{Pe}}{d_{PE}}$	$\dfrac{e^{\mathrm{Pe}_e \frac{d_{Pe}}{d_{PE}}} - 1}{e^{\mathrm{Pe}_e} - 1}$, $\mathrm{Pe}_e = \dfrac{\rho_e u_e d_{PE}}{\Gamma_e}$
West	$\dfrac{\dot{m}_w + \lvert\dot{m}_w\rvert}{2\dot{m}_w}$	$\dfrac{d_{Pw}}{d_{PW}}$	$\mathrm{Pe}_w = \dfrac{\rho_w u_w d_{PW}}{\Gamma_w}$
North	$\dfrac{\dot{m}_n - \lvert\dot{m}_n\rvert}{2\dot{m}_n}$	$\dfrac{d_{Pn}}{d_{PN}}$	$\dfrac{e^{\mathrm{Pe}_n \frac{d_{Pn}}{d_{PN}}} - 1}{e^{\mathrm{Pe}_n} - 1}$, $\mathrm{Pe}_n = \dfrac{\rho_n v_n d_{PN}}{\Gamma_n}$
South	$\dfrac{\dot{m}_s + \lvert\dot{m}_s\rvert}{2\dot{m}_s}$	$\dfrac{d_{Ps}}{d_{PS}}$	$\mathrm{Pe}_s = \dfrac{\rho_s v_s d_{PS}}{\Gamma_s}$

Table 3.1. Coefficient A_f of equation $\phi_f - \phi_P = A_f(\phi_F - \phi_P)$ for east, west, north and south faces, and for schemes UDS, CDS and EDS.

Introducing equations (3.81) through (3.84) in (3.33), a general discretization equation comprising UDS, CDS and EDS can be obtained:

$$\begin{aligned}
&\left\{D_e - \dot{m}_e A_e + D_w + \dot{m}_w A_w + D_n - \dot{m}_n A_n + D_s + \dot{m}_s A_s + \frac{\rho_P^0 V_P}{\Delta t} - S_P^\phi V_P\right\}\phi_P \\
&= \left\{D_e - \dot{m}_e A_e\right\}\phi_E + \left\{D_w + \dot{m}_w A_w\right\}\phi_W + \left\{D_n - \dot{m}_n A_n\right\}\phi_N + \left\{D_s + \dot{m}_s A_s\right\}\phi_S \\
&+ \left\{S_C^\phi + \frac{\rho_P^0 \phi_P^0}{\Delta t}\right\} V_P
\end{aligned} \tag{3.85}$$

In order to make (3.85) more tractable, the following discretization coefficients are defined:

$$a_E = D_e - \dot{m}_e A_e \tag{3.86}$$
$$a_W = D_w + \dot{m}_w A_w \tag{3.87}$$
$$a_N = D_n - \dot{m}_n A_n \tag{3.88}$$
$$a_S = D_s + \dot{m}_s A_s \tag{3.89}$$
$$a_P = a_W + a_E + a_S + a_N + \frac{\rho_P^0 V_P}{\Delta t} - S_P^\phi V_P \tag{3.90}$$
$$b_P = S_C^\phi V_P + \frac{\rho_P^0 \phi_P^0}{\Delta t} V_P \tag{3.91}$$

Thereby the discretization equation is:

$$a_P \phi_P = a_W \phi_W + a_E \phi_E + a_S \phi_S + a_N \phi_N + b_P \tag{3.92}$$

3.7.2 Large molecule schemes

High–resolution schemes (HRS) such as SUDS, QUICK and SMART, not only use adjacent nodes to the faces but also the most upstream nodes, that is to say, involve a larger molecule. Since a larger molecule increases the memory usage and the computational effort, it is desirable to keep it as low as possible. Therefore, the aim is to obtain a discretization equation such as (3.92), where only the surrounding nodes participate, while upstream nodes are computed by different means and collected in b_P.

The first logical solution would be to use small molecule schemes, although it must be kept in mind the lower order of the approximations. The second solution would be to compute the upstream node value using the data of the previous iteration and introduce this term in the equation as a contribution to b_P. Nevertheless, this may lead to the divergence of the iterations since the terms treated explicitly may be substantial [6].

The solution is to compute the approximated terms with a higher order approximation explicitly and put them on the right–hand side of equation (3.79). Then a simpler approximation to these terms, for instance one that provides a smaller molecule, is put on the left–hand side and on the right–hand side, computing it using explicit values. Then the right–hand side is the difference between two approximations of the same value, hence is likely to be small. This technique is known as deferred correction, and is used with higher–order approximations, as well as grid non–orthogonality and correction to prevent undesirable effects in solutions [6].

Given a face f, the idea is approximate ϕ_f as

$$\phi_f^{\text{HRS}} - \phi_P = (\phi_f^{\text{UDS}} - \phi_P) + (\phi_f^{\text{HRS},*} - \phi_f^{\text{UDS},*}) \tag{3.93}$$

ϕ_f^{HRS} and ϕ_f^{UDS} are the current calculated values of ϕ using the chosen HRS and UDS, whereas $\phi_f^{\text{HRS},*}$ and $\phi_f^{\text{UDS},*}$ are the computed values in the previous iteration. As stated above, when convergence is achieved, $\phi_f^{\text{HRS}} = \phi_f^{\text{HRS},*}$ and $\phi_f^{\text{UDS}} = \phi_f^{\text{UDS},*}$ [7]. Substituting $\phi_f - \phi_P$ by $\phi_f^{\text{HRS}} - \phi_P$ in (3.31)

$$\begin{aligned} \rho_P^0 \frac{\phi_P - \phi_P^0}{\Delta t} V_P + \dot{m}_e(\phi_e^{\text{HRS}} - \phi_P) - \dot{m}_w(\phi_w^{\text{HRS}} - \phi_P) + \dot{m}_n(\phi_n^{\text{HRS}} - \phi_P) - \dot{m}_s(\phi_s^{\text{HRS}} - \phi_P) = \\ = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) - D_s(\phi_P - \phi_S) + (S_C^\phi + S_P^\phi \phi_P) V_P \end{aligned} \tag{3.94}$$

and using relation (3.93)

$$\begin{aligned}\rho_P^0 \frac{\phi_P - \phi_P^0}{\Delta t} V_P + \dot{m}_e(\phi_e^{\text{UDS}} - \phi_P) - \dot{m}_w(\phi_w^{\text{UDS}} - \phi_P) + \dot{m}_n(\phi_n^{\text{UDS}} - \phi_P) - \dot{m}_s(\phi_s^{\text{UDS}} - \phi_P) = \\ = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + D_n(\phi_N - \phi_P) - D_s(\phi_P - \phi_S) + (S_C^\phi + S_P^\phi \phi_P)V_P + \\ - \dot{m}_e(\phi_e^{\text{HRS},*} - \phi_e^{\text{UDS},*}) + \dot{m}_w(\phi_w^{\text{HRS},*} - \phi_w^{\text{UDS},*}) - \dot{m}_n(\phi_n^{\text{HRS},*} - \phi_n^{\text{UDS},*}) + \dot{m}_s(\phi_s^{\text{HRS},*} - \phi_s^{\text{UDS},*})\end{aligned} \tag{3.95}$$

Replacing the corresponding terms with expressions (3.36) through (3.39) and rearranging terms, the desired expression is found

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + b_P \tag{3.96}$$

with the following coefficients:

$$a_E = D_e - \frac{\dot{m}_e - |\dot{m}_e|}{2} = \frac{\Gamma_e S_e}{d_{PE}} - \frac{\dot{m}_e - |\dot{m}_e|}{2} \tag{3.97}$$

$$a_W = D_w + \frac{\dot{m}_w + |\dot{m}_w|}{2} = \frac{\Gamma_w S_w}{d_{PW}} + \frac{\dot{m}_w + |\dot{m}_w|}{2} \tag{3.98}$$

$$a_N = D_n - \frac{\dot{m}_n - |\dot{m}_n|}{2} = \frac{\Gamma_n S_n}{d_{PN}} - \frac{\dot{m}_n - |\dot{m}_n|}{2} \tag{3.99}$$

$$a_S = D_s + \frac{\dot{m}_s + |\dot{m}_s|}{2} = \frac{\Gamma_s S_s}{d_{PS}} + \frac{\dot{m}_s + |\dot{m}_s|}{2} \tag{3.100}$$

$$a_P = a_W + a_E + a_S + a_N + \frac{\rho_P^0 V_P}{\Delta t} - S_P^\phi V_P \tag{3.101}$$

$$\begin{aligned}b_P = \frac{\rho_P^0 \phi_P^0}{\Delta t} V_P + S_C^\phi V_P - \dot{m}_e(\phi_e^{\text{HRS},*} - \phi_e^{\text{UDS},*}) + \dot{m}_w(\phi_w^{\text{HRS},*} - \phi_w^{\text{UDS},*}) \\ - \dot{m}_n(\phi_n^{\text{HRS},*} - \phi_n^{\text{UDS},*}) + \dot{m}_s(\phi_s^{\text{HRS},*} - \phi_s^{\text{UDS},*})\end{aligned} \tag{3.102}$$

3.8 Treatment of boundary conditions

In Cauchy problems involving Partial Differential Equations (PDEs), there exist several kinds of boundary conditions which must be prescribed in order to guarantee the existence and uniqueness of solution, although in this project only two will be considered. So as to illustrate how these conditions are set, let $U \subset \mathbb{R}^m$ be a bounded open subset of $\mathbb{R}^m$. The diffusion equation, which describes the evolution in time of the density of a magnitude u such as a heat or chemical substance, is the PDE

$$u_t - \Delta u = f(\mathbf{x}, t) \quad (\mathbf{x}, t) \in U \times (0, \infty) \tag{3.103}$$

where $\Delta = \sum_{i=1}^{m} \frac{\partial^2}{\partial x_i^2}$ is Laplace's operator and f models the internal sources for magnitude u. Let $g\colon U \to \mathbb{R}$ be the initial value for u. Then the typical Cauchy problem for diffusion equation is

$$\begin{cases} u_t - \Delta u = f(\mathbf{x}, t) & \text{in } U \times (0, \infty) \\ u = g & \text{on } U \times \{t = 0\} \\ \text{Boundary conditions} & \end{cases} \tag{3.104}$$

The boundary conditions considered are:

- Dirichlet boundary condition: the value of u is prescribed on $\partial U \times (0, \infty)$, that is to say, if $d\colon \partial U \times (0,\infty) \to \mathbb{R}$, $(\mathbf{x}, t) \mapsto d(\mathbf{x}, t)$ describes the boundary condition, then (3.104) is written as
$$\begin{cases} u_t - \Delta u = f(\mathbf{x}, t) & \text{in } U \times (0, \infty) \\ u = g & \text{on } U \times \{t = 0\} \\ u = d & \text{on } \partial U \times (0, \infty) \end{cases} \tag{3.105}$$
When (3.103) is thought of as describing the propagation of heat, then d fixes the temperature at the boundary of U for each time.

- Neumann boundary condition: the normal derivative of u to the boundary of U is prescribed on $\partial U \times (0, \infty)$, i.e. if $n\colon \partial U \times (0,\infty) \to \mathbb{R}$ describes the boundary condition, then (3.104) is written as
$$\begin{cases} u_t - \Delta u = f(\mathbf{x}, t) & \text{in } U \times (0, \infty) \\ u = g & \text{on } U \times \{t = 0\} \\ \partial_\nu u = n & \text{on } \partial U \times (0, \infty) \end{cases} \tag{3.106}$$
where ν is the outer normal vector to ∂U. In terms of heat, this boundary condition sets the conduction heat transfer through U for each time.

The numerical treatment of boundary conditions is straightforward, speacially when a cartesian mesh on a rectangular domain is used.

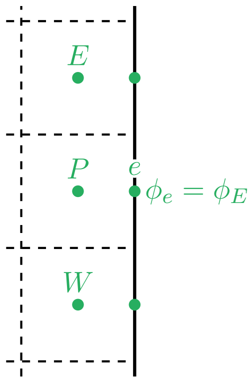


Figure 3.14. Dirichlet boundary condition.

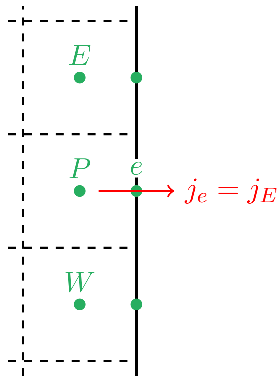


Figure 3.15. Neumann boundary condition.

In the case of a Dirichlet boundary condition, such as the one shown in figure 3.14, the value at face must be equal to the prescribed value at boundary, that is,

$$\phi_e = \phi_E \tag{3.107}$$

and flux per unit of surface can be easily computed as

$$j_e = -\Gamma_P \frac{\phi_e - \phi_P}{d_{Pe}} \tag{3.108}$$

In contrast, when a Neumann boundary condition with flux j_e, the value at face is

$$\phi_e = \phi_P - \frac{j_e d_{Pe}}{\Gamma_P} \tag{3.109}$$

This second situation is pictured in figure 3.15.

3.9 Solving algorithm

The procedure to solve of a transient convection–diffusion problem with 2D–cartesian mesh is shown in Algorithm 1.

Algorithm 1 Resolution of a transient convection–diffusion problem with 2D–cartesian mesh.

1 Input data:

1.1 Physical data: geometry, thermophysical properties, initial and boundary conditions.

1.2 Numerical data: mesh, Δt (time step), δ (convergence criterion).

2 Mesh generation: nodes position, faces position, distances, surfaces and volumes.

3 Initial map: $n \leftarrow 0$, $t^n \leftarrow 0$, $\phi^0[i][j] = \phi(x,y,t)|_{t=0}$.

4 Compute the new time step: $t^{n+1} = t^n + \Delta t$.

4.1 Initial estimated values: $\phi^*[i][j] \leftarrow \phi^0[i][j]$.

4.2 Evaluation of the discretization coefficients: $a_E[i][j]$, $a_W[i][j]$, $a_N[i][j]$, $a_S[i][j]$, $a_P[i][j]$, $b_P[i][j]$.

4.3 Resolution of the linear system

$$\begin{aligned} a_P[i][j]\,\phi[i][j] = a_E[i][j]\,\phi[i+1][j] + a_W[i][j]\,\phi[i-1][j] \\ + a_N[i][j]\,\phi[i][j+1] + a_S[i][j]\,\phi[i][j-1] + b_P[i][j] \end{aligned}$$

4.4 Is $\max\limits_{i,j} |\phi^*[i][j] - \phi[i][j]| < \delta$?

- Yes: continue.
- No: $\phi^*[i][j] \leftarrow \phi[i][j]$, go to **4.2**.

5 New time step?

- Yes: $n \leftarrow n+1$, go to **4**.
- No: continue.

6 Final computations, print results.

7 End.

Hereinafter, the term iteration will be used to refer to the iterative procedure to solve the linear system on step **4.3**. It must not be confused with the next time instant term.

On step **4** the next time instant is computed. This is the most computationally expensive step in the algorithm, specially part **4.3** where the resolution of the linear system of discretized equations is carried out. As a result of the convection–diffusion equations nature, the system matrix A and the vector of independent terms b change each time the convergence condition is not fulfilled on step (**4.4**). Since A and b depend on the previous iteration value of ϕ, that is to say, $A = A(\phi^*)$ and $b = b(\phi^*)$, the linear system of equations is

$$A(\phi^*)\,\phi = b(\phi^*) \tag{3.110}$$

In the case both A and b were constant, the algorithm needed to solve the linear system (3.110) would be clear at first glance. Since each node is influenced only by the adjacent nodes, A is a pentadiagonal by blocks matrix, therefore A is sparse, i.e. most of the elements are zero, hence an iterative method for solving the linear system would be convenient. Let A_{ij} denote the element in the i-th row and j-th column of A. If A is, in addition, a strictly diagonally dominant (SDD) matrix, that is to say,

$$|A_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^{n} |A_{ij}| \tag{3.111}$$

then Gauss–Seidel algorithm is guaranteed to converge and eventually solve the system. In terms of the discretization coefficients, condition (3.111) is written as

$$|a_P[i][j]| \geq |a_W[i][j]| + |a_E[i][j]| + |a_S[i][j]| + |a_N[i][j]| \tag{3.112}$$

Nonetheless the SDD condition is not assured as a consequence of the discretization coefficients.

As the execution of an iterative procedure to solve the linear system may diverge, arguably the most convenient is a direct method.

4 Parallel flow case

5 Diagonal flow case

5.1 Statement

The diagonal flow case is a steady state problem which takes places in a square domain $\Omega = (0, L) \times (0, L) \subset \mathbb{R}^2$ where $L > 0$ is a fixed length. In it the steady state general convection–diffusion equation with no source term, constant density and constant diffusion coefficient is considered. Under these hypothesis equation (2.15) is simplified to

$$\frac{\rho}{\Gamma}\mathbf{v} \cdot \boldsymbol{\nabla}\phi = \Delta\phi \tag{5.1}$$

The following boundary conditions are prescribed:

- $\phi = \phi_{\text{low}}$ on $[0, L) \times \{0\} \cup \{L\} \times [0, L)$.
- $\phi = \phi_{\text{high}}$ on $\{0\} \times (0, L] \cup (0, L] \times \{L\}$.

The velocity field is $\mathbf{v} = v_0 \cos(\alpha)\mathbf{i} + v_0 \sin(\alpha)\mathbf{j}$, with $v_0 > 0$ constant and $\alpha = \pi/4$. Therefore the left–hand side of (5.1) is

$$\frac{\rho}{\Gamma}\mathbf{v} \cdot \boldsymbol{\nabla}\phi = \frac{\rho}{\Gamma}\left(v_0\cos(\alpha)\frac{\partial\phi}{\partial x} + v_0\sin(\alpha)\frac{\partial\phi}{\partial y}\right) = \frac{\rho v_0\cos(\alpha)}{\Gamma}\left(\frac{\partial\phi}{\partial x} + \frac{\partial\phi}{\partial y}\right) \tag{5.2}$$

$$= \underbrace{\frac{\cos(\alpha)}{L}}_{\beta}\underbrace{\frac{\rho v_0 L}{\Gamma}}_{\text{Pe}}\left(\frac{\partial\phi}{\partial x} + \frac{\partial\phi}{\partial y}\right) = \beta\text{Pe}\left(\frac{\partial\phi}{\partial x} + \frac{\partial\phi}{\partial y}\right) \tag{5.3}$$

The resulting Cauchy problem is

$$\begin{cases} \Delta\phi - \beta\text{Pe}\left(\dfrac{\partial\phi}{\partial x} + \dfrac{\partial\phi}{\partial y}\right) = 0 & \text{in } \Omega = (0, L) \times (0, L) \\ \phi = \phi_{\text{low}} & \text{on } [0, L) \times \{0\} \cup \{L\} \times [0, L) \\ \phi = \phi_{\text{high}} & \text{on } \{0\} \times (0, L] \cup (0, L] \times \{L\} \end{cases} \tag{5.4}$$

and is summarized in figure 5.1.

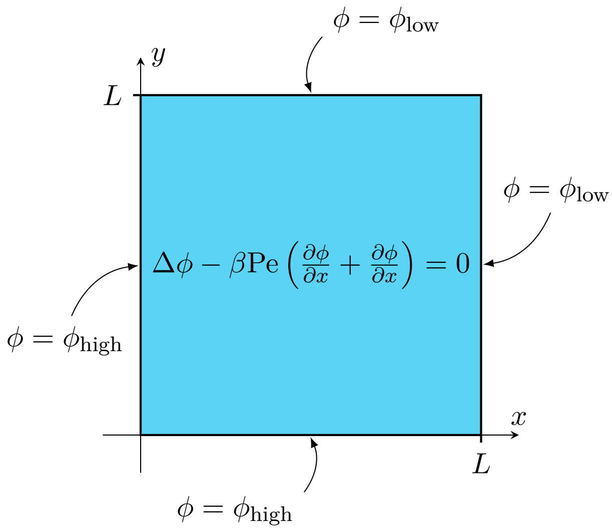


Figure 5.1. Cauchy problem for the diagonal flow case.

5.2 Analytical solution

Peclet's number is defined as the ratio between transport and heat difussion, $\mathrm{Pe} = \rho u L/\Gamma$. Whenever $\mathrm{Pe} \to +\infty$, it implies $\Gamma \to 0$ since infinite values for the density, velocity or characteristic length make no physical sense. Therefore the difussion coefficient tends to 0, which means the Laplacian term is negligible. Under this physical intuition, dividing the PDE from (5.4) results in the following Cauchy problem:

$$\begin{cases} \dfrac{\partial \phi}{\partial x} + \dfrac{\partial \phi}{\partial y} = 0 & \text{in } \Omega = (0, L) \times (0, L) \\ \phi = \phi_{\text{low}} & \text{on } [0, L) \times \{0\} \cup \{L\} \times [0, L) \\ \phi = \phi_{\text{high}} & \text{on } \{0\} \times (0, L] \cup (0, L] \times \{L\} \end{cases} \tag{5.5}$$

The PDE from (5.5) is known as the transport equation, which is a first order linear PDE. In our case it has constant coefficients, making it easier to solve analitically. So as to find the analytical solution to (5.5), we will follow a geometric approach as it is more intuitive.

by dividing the PDE from (5.4) by

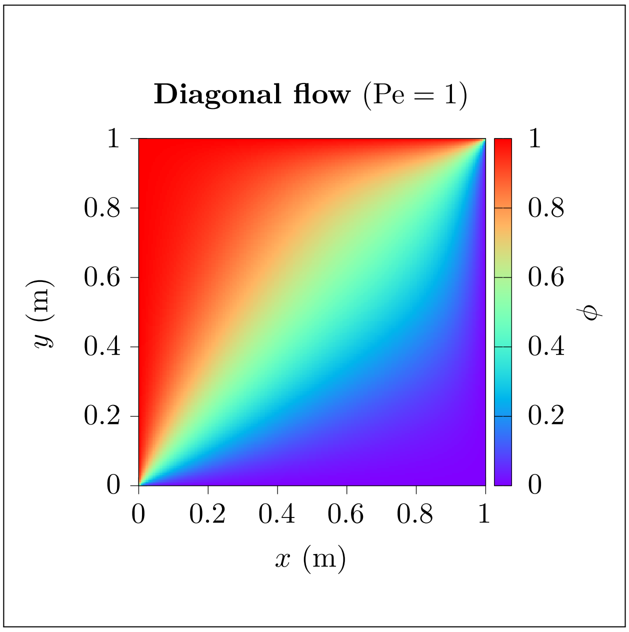


Figure 5.2. Test

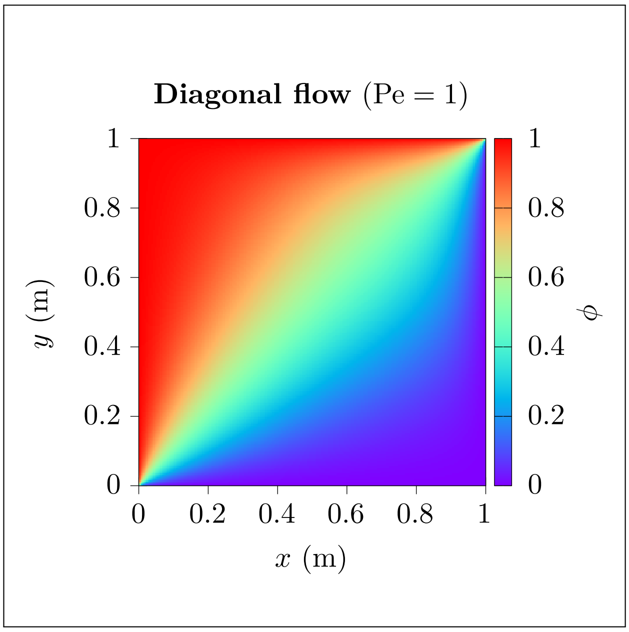


Figure 5.3. Test

6 Smith–Hutton case

6.1 Statement

This section deals with the steady state version of the problem proposed by Smith and Hutton (1982) described in [8]. The problem takes place in the domain $\Omega = (-L, L) \times (0, L) \subset \mathbb{R}$ where $L > 0$ is a constant. Both density and diffusion coefficient are assumed to be constant and known values. In Ω the steady state version of the general convection–diffusion equation with no source term is considered, that is,

$$\frac{\rho}{\Gamma}\mathbf{v} \cdot \boldsymbol{\nabla}\phi = \Delta\phi \tag{6.1}$$

On the boundary of Ω the following conditions are prescribed:

- $\phi = 1 + \tanh(10(2x + 1))$ on $[-L, 0] \times \{0\}$ (inlet flow).
- $\dfrac{\partial\phi}{\partial y} = 0$ on $(0, L] \times \{0\}$ (outlet flow).
- $\phi = 1 - \tanh(10)$ on $\{-L\} \times (0, L) \cup [-L, L] \times \{L\} \cup \{L\} \times (0, L)$.

The velocity field is given by $v_x = 2y(1-x^2)$ and $v_y = -2x(1-y^2)$, which verifies the incompressibility condition since $\boldsymbol{\nabla} \cdot \mathbf{v} = 0$. The Cauchy problem is the following:

$$\begin{cases} \Delta\phi - \dfrac{\rho}{\Gamma}\mathbf{v} \cdot \boldsymbol{\nabla}\phi = 0 & \text{in } \Omega = (0, L) \times (0, L) \\ \phi = 1 + \tanh(10(2x + 1)) & \text{on } [-L, 0] \times \{0\} \\ \dfrac{\partial\phi}{\partial y} = 0 & \text{on } (0, L] \times \{0\} \\ \phi = 1 - \tanh(10) & \text{on } \{-L\} \times (0, L) \cup [-L, L] \times \{L\} \cup \{L\} \times (0, L) \end{cases} \tag{6.2}$$

Problem (6.2) is summarized in figure 6.1.

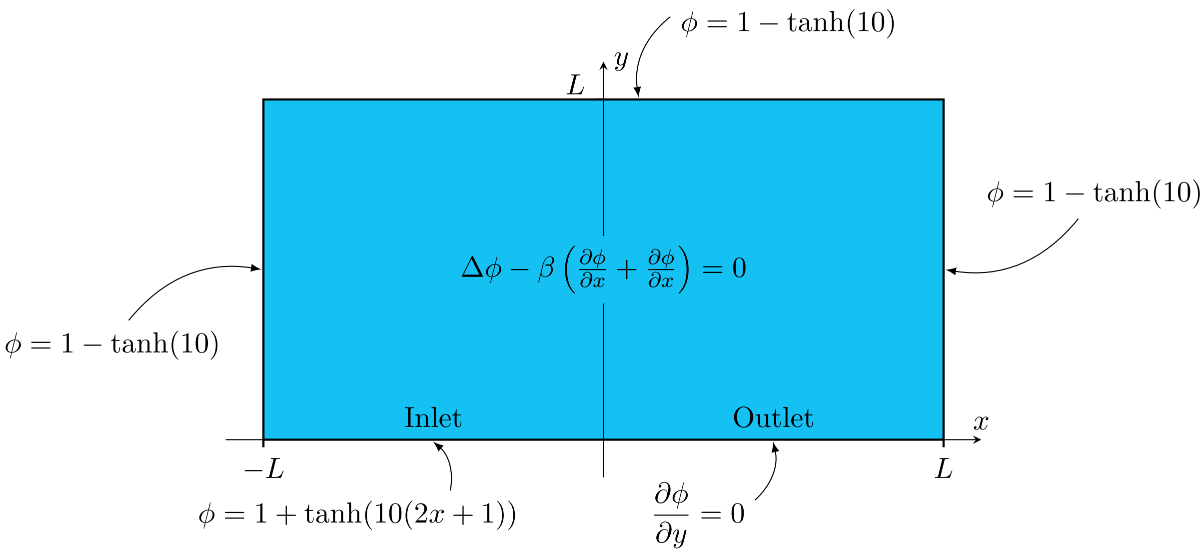
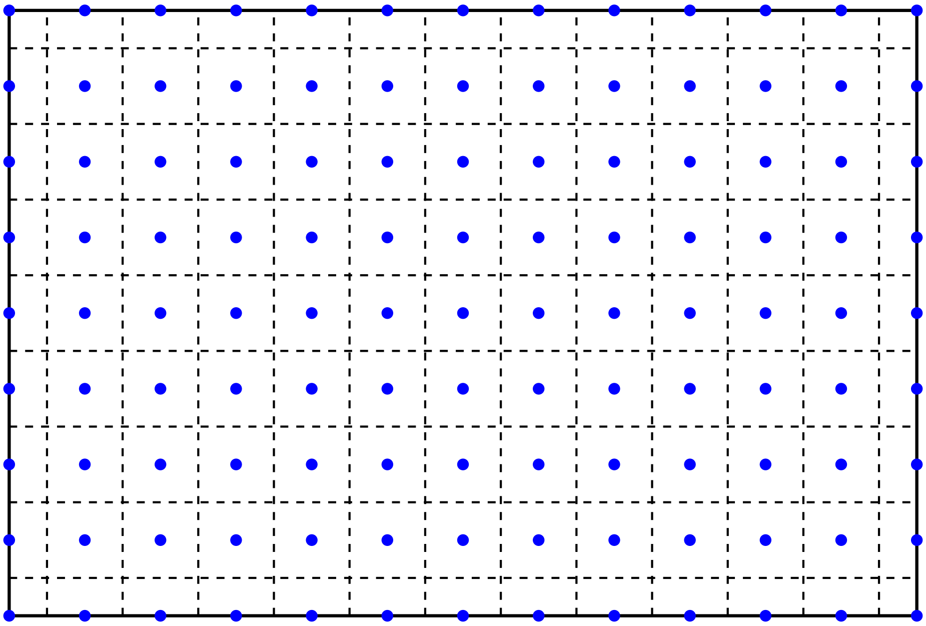
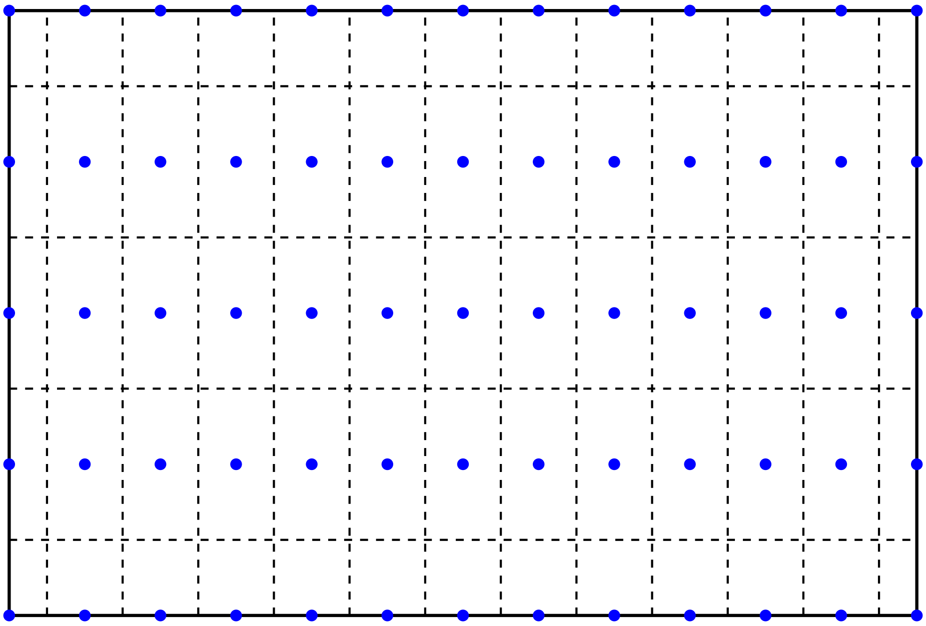


Figure 6.1. Cauchy problem for the diagonal flow case.

7 Test





References

[1] Wilfred Kaplan. *Advanced Calculus.* 5th ed. Pearson, 2002. Chap. 4, pp. 253–256.

[2] Pijush K Kundu, Ira M Cohen, and D Dowling. *Fluid Mechanics.* 6th ed. Elsevier, 2016. Chap. 3, pp. 99–102.

[3] Suhas V Patankar. *Numerical heat transfer and fluid flow.* McGraw–Hill Book Company, 1980.

[4] Joel H Ferziger, Milovan Perić, and Robert L Street. *Computational methods for fluid dynamics.* 4th ed. Springer, 2002. Chap. 2, pp. 28–33.

[5] PH Gaskell and AKC Lau. "Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm". In: *International Journal for numerical methods in fluids* 8.6 (1988), pp. 617–641.

[6] Joel H Ferziger, Milovan Perić, and Robert L Street. *Computational methods for fluid dynamics.* 4th ed. Springer, 2002. Chap. 5, pp. 144–146.

[7] CTTC. "Numerical resolution of the generic convection diffusion equation". Notes of the Course on Numerical Methods in Heat Transfer and Fluid Dynamics. July 2021.

[8] RM Smith and AG Hutton. "The numerical treatment of advection: A performance comparison of current methods". In: *Numerical Heat Transfer, Part A Applications* 5.4 (1982), pp. 439–461.

[9] Walter Rudin. *Real and Complex Analysis.* 3rd ed. McGraw–Hill, 1987.

[10] Lawrence C. Evans. *Partial Differential Equations.* 1st ed. Vol. 19. American Mathematical Society, 1998. Chap. E, p. 649.

A Some results on measure theory

In this appendix we gather two important theorems needed to justify some steps in the derivation of conservation laws in section 2. Despite these results are basic, a previous study of real analysis is required in order to understand and prove them. A good reference for the interested reader is Real and Complex Analysis of Walter Rudin [9].

A.1 Differentiation under the integral sign

Differentiation under the integral sign allows us to compute the derivative of an integral of a function of two parameters in a simple way. It is needed, for instance, when the mass conservation law or the heat diffusion equation are derived.

Let $(X, \mathcal{A}, \mu)$ be a measure space and let $[a,b] \subset \mathbb{R}$. Hereinafter we deal with functions $f\colon X \times [a,b] \to \mathbb{R}$, where $t \in [a,b]$ is the parameter on which f depends. We assume that $f(\cdot, t)$ is a measurable function for each $t \in [a,b]$.

Theorem A.1 (Differentiation under the integral sign)**.** Let $F(t) = \int_X f(\mathbf{x}, t)\,\mathrm{d}\mu$. Assume that

(i) $f(\mathbf{x}, t_0)$ is an integrable function for some $t_0 \in [a,b]$.

(ii) $\dfrac{\partial f}{\partial t}(\mathbf{x}, t)$ is defined for all $(\mathbf{x}, t) \in X \times [a,b]$.

(iii) There exists an integral function $g\colon X \to \mathbb{R}$ such that $\left|\dfrac{\partial f}{\partial t}(\mathbf{x}, t)\right| \leq g(\mathbf{x})$ for all $(\mathbf{x}, t) \in X \times [a,b]$.

Then F is a differentiable function and

$$F'(t) = \frac{\mathrm{d}}{\mathrm{d}t}F(t) = \int_X \frac{\partial f}{\partial t}(\mathbf{x}, t)\,\mathrm{d}\mu$$

For the applications needed in this project, $X = \mathbb{R}^m$ with $1 \leq m \leq 3$, $\mathcal{A}$ is the Borel σ–algebra on $\mathbb{R}^m$ and μ is Lebesgue's measure on $\mathbb{R}^m$, which for most of the "natural" sets of $\mathcal{A}$ coincides with the usual notion of m–dimensional volume.

A.2 Lebesgue's differentiation lemma

A common way to derive a conservation law is to integrate some functions in a control volume, then apply Differentiation under the integral sign to obtain an integral equation and finally get to a differential equation using Lebesgue's differentiation lemma.

An intuitive way to understand and to motivate Lebesgue's differentiation lemma is the following. Let $f\colon \mathbb{R} \to \mathbb{R}$ be a continuous function, let $a \in \mathbb{R}$ be a fixed point and let $F(x) = \int_a^x f(y)\,\mathrm{d}y$, which is a differentiable function. Due to a corollary of the Fundamental Theorem of Calculus, we have $F'(x) = f(x)$. Using the definition of derivative,

$$F'(x) = \lim_{h \to 0} \frac{F(x+h) - F(x)}{h} = \lim_{h \to 0} \frac{1}{h}\left\{\int_a^{x+h} f(y)\,\mathrm{d}y - \int_a^x f(y)\,\mathrm{d}y\right\} = \lim_{h \to 0} \frac{1}{h}\int_x^{x+h} f(y)\,\mathrm{d}y = f(x)$$

Notice that the integral is divided by the length of the interval $[x, x+h]$, otherwise the limit would be zero. Lebesgue's lemma generalizes the previous equality by considering functions $f\colon \mathbb{R}^n \to \mathbb{R}$ and integrating them on open balls $B(\mathbf{x_0}, r) = \{x \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{x_0}\| < r\}$. Furthermore, the integral is divided by the n–dimensional volume of $B(\mathbf{x_0}, r)$, which is denoted by $|B(\mathbf{x_0}, r)|$.

Theorem A.2 (Lebesgue's differentiation lemma [10])**.** Let $f\colon \mathbb{R}^n \to \mathbb{R}$ be a locally integrable function.

(1) Then for almost everywhere point $\mathbf{x_0} \in \mathbb{R}^n$,

$$\frac{1}{|B(\mathbf{x_0}, r)|} \int_{B(\mathbf{x_0}, r)} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} \to f(\mathbf{x_0}) \quad \text{as } r \to 0$$

(2) In fact, for almost everywhere point $\mathbf{x_0} \in \mathbb{R}^n$,

$$\frac{1}{|B(\mathbf{x_0}, r)|} \int_{B(\mathbf{x_0}, r)} |f(\mathbf{x}) - f(\mathbf{x_0})| \, \mathrm{d}\mathbf{x} \to 0 \quad \text{as } r \to 0$$

B Numerical resolution of linear systems

B.1 Gauss–Seidel algorithm

B.2 LU factorization