



ECOLE  
POLYTECHNIQUE  
DE BRUXELLES

UNIVERSITÉ LIBRE DE BRUXELLES

SUMMARY

---

# Numerical methods in aerothermodynamics

## MECA-H-407

---

*Author:*  
Enes ULUSOY

*Professor:*  
Gérard DEGREZ

Year 2016 - 2017

# Appel à contribution

## Synthèse Open Source



Ce document est grandement inspiré de l'excellent cours donné par Gérard DEGREGZ à l'EPB (École Polytechnique de Bruxelles), faculté de l'ULB (Université Libre de Bruxelles). Il est écrit par les auteurs susnommés avec l'aide de tous les autres étudiants et votre aide est la bienvenue ! En effet, il y a toujours moyen de l'améliorer surtout que si le cours change, la synthèse doit être changée en conséquence. On peut retrouver le code source à l'adresse suivante

<https://github.com/nenglebert/Syntheses>

Pour contribuer à cette synthèse, il vous suffira de créer un compte sur *Github.com*. De légères modifications (petites coquilles, orthographe, ...) peuvent directement être faites sur le site ! Vous avez vu une petite faute ? Si oui, la corriger de cette façon ne prendra que quelques secondes, une bonne raison de le faire !

Pour de plus longues modifications, il est intéressant de disposer des fichiers : il vous faudra pour cela installer L<sup>A</sup>T<sub>E</sub>X, mais aussi *git*. Si cela pose problème, nous sommes évidemment ouverts à des contributeurs envoyant leur changement par mail ou n'importe quel autre moyen.

Le lien donné ci-dessus contient aussi un README contenant de plus amples informations, vous êtes invités à le lire si vous voulez faire avancer ce projet !

## Licence Creative Commons

Le contenu de ce document est sous la licence Creative Commons : *Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)*. Celle-ci vous autorise à l'exploiter pleinement, compte- tenu de trois choses :



1. *Attribution* ; si vous utilisez/modifiez ce document vous devez signaler le(s) nom(s) de(s) auteur(s).
2. *Non Commercial* ; interdiction de tirer un profit commercial de l'œuvre sans autorisation de l'auteur
3. *Share alike* ; partage de l'œuvre, avec obligation de rediffuser selon la même licence ou une licence similaire

Si vous voulez en savoir plus sur cette licence :

<http://creativecommons.org/licenses/by-nc-sa/4.0/>

**Merci !**

# Contents

<b>1</b>	<b>Discretization methods</b>	<b>3</b>
1.1	Finite difference method . . . . .	3
1.1.1	Evaluation of derivatives by finite differences . . . . .	3
1.1.2	Finite difference formulas for partial differential equations . . . . .	5
1.1.3	Arbitrary geometries - irregular meshes . . . . .	6
1.2	Finite volume method . . . . .	6
1.2.1	Fundamental principles and variants of the method . . . . .	7
1.2.2	Evaluation of fluxes through faces . . . . .	8



# Introduction

Fluid dynamics is based on continuity hypothesis, all quantities can be expressed as a continuous function of time and space coordinates. The governing equations are partial differential equations. Because of the geometrical complexity of the domain and of the equations, we need strategies. The first one is to forget about the equations and to rely on experiments. The second is to consider simplified cases, and approximate theoretical analytic solutions (aerodynamics). The third approach is numerical approach. Disadvantages and advantages of the different methods can be listed as:

- **Experimental:** the advantage is that it is the most realistic, but requires equipment, there are scale problems (similarity), interferences (tests in finite space), measurement difficulties and operating costs.
- **Theoretical:** the advantage is that we have a mathematical expression and we don't have to repeat calculus, but it is restricted to simple geometries and linear problems.
- **Numerical:** the advantages are that we can deal with complex geometries, non linear problems and unsteady problems, but there are truncation errors, problems with boundary conditions like the finite space in experiment and the computation cost.

In reality these approaches are complementary. We can use the second method to simplify the numerical computations, crucial for example for costly operations like computations on turbulence. The evolution of numerical cost over the past 40 years has been particularly impressive, cost decreased dramatically. In the other hand, the experimental cost tends to increase (technical personal, material, ...). Nowadays we can measure many things impossible to measure before. This explains why the numerical computations have spread incredibly.

The design relies mainly on the numerical methods and less on the experimental testing, but it is still needed to confirm the data. We can use the numerical methods in many fields and we could call this « numerical physics ». We should deal with this in a single course of computational method and then to specialize it to the specific disciplines. An approximate solution to a problem is some kind of mathematical entity, an object, which depends on a finite number of real parameters, and which constitutes a representation of the continuous field under study. The numerical solution belongs to a finite dimension space whereas the theoretical to an infinite.

There are different types of numerical representation:

- **Discrete:** collection of either point values (samples of the solutions) or subdomain averages. We are not able to give an exact solution on basis of these points, but rather an estimation.
- **Functional:** the solution is expressed as a function  $u^*(x) = f(x, a_i)$ , depending on a set of variables. Most of the time the dependence is linear:  $u^* = \sum_{i=1}^n a_i v_i(x)$  where  $v_i(x)$  are a priori specified functions.

Once we have chosen the numerical representation method, we have to generate a system of algebraic equations linking the parameters from the representation and the governing equations. The last step is to solve the system. The step of generating the equations system is called **discretization**. For one problem, several discretizations are possible. Some examples are given, consult the syllabus for more details.

# Chapter 1

## Discretization methods

### 1.1 Finite difference method

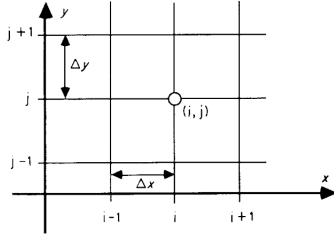


Figure 1.1

It is based on discrete representation of numerical solution consisting in values of the solution at the nodes of a Cartesian mesh of uniform spacing (Figure 1.1). The discretization will consist in estimating the partial derivatives appearing in the governing equations by some algebraic relations at the nodes. Because of the limitation of the mesh we cannot deal with curved geometries. To do so we need coordinate changes. In this type of mesh, each grid point can be identified by a set of indexes  $i, j, (k)$ , this type of mesh is called **structured mesh**. The neighbors are

implicitly given by the index identification.

#### 1.1.1 Evaluation of derivatives by finite differences

##### Difference formulas for the derivative $\partial u / \partial x$

The value of a function  $u(x)$  on a point of indexes  $i, j$  on the mesh is noted  $u_{ij}$ , for time dependent functions we use  $u_i^n$  where  $n$  denotes the time index and  $i$  the space index. Let's estimate  $(\partial u / \partial x)_{ij}$ , by definition:

$$\left. \frac{\partial u}{\partial x} \right|_{ij} = \left. \frac{\partial u}{\partial x} \right|_{x_0, y_0} = \lim_{\xi \rightarrow 0} \frac{u(x_0 + \xi, y_0) - u(x_0, y_0)}{\xi} \quad (1.1)$$

By taking  $\xi = \Delta x$  to fit our grid points we get:

$$\left. \frac{\partial u}{\partial x} \right|_{ij} \approx \lim_{\Delta x \rightarrow 0} \frac{u(x_0 + \Delta x, y_0) - u(x_0, y_0)}{\Delta x} = \frac{u_{i+1j} - u_{ij}}{\Delta x} \quad (1.2)$$

In order to find a systematic way of deriving the equations, let's build the Taylor expansion of  $u(x, y)$  around the mesh point  $ij$ :

$$\begin{aligned} u_{i+1j} &= u_{ij} + \Delta x \left. \frac{\partial u}{\partial x} \right|_{ij} + \frac{\Delta x^2}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} + \frac{\Delta x^3}{6} \left. \frac{\partial^3 u}{\partial x^3} \right|_{\xi} \quad x_0 \leq \xi \leq x_0 + \Delta x \\ \Leftrightarrow \left. \frac{\partial u}{\partial x} \right|_{ij} &= \frac{u_{i+1j} - u_{ij}}{\Delta x} - \underbrace{\frac{\Delta x}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} - \frac{\Delta x^2}{6} \left. \frac{\partial^3 u}{\partial x^3} \right|_{\xi}}_{\text{truncation error}} = \frac{u_{i+1j} - u_{ij}}{\Delta x} + \mathcal{O}(\Delta x) \end{aligned} \quad (1.3)$$

Giving the **forward difference formula**. We have then a truncation error which behavior is dominated by the first term when  $\Delta x \rightarrow 0$ , so that  $TE = \mathcal{O}(\Delta x)$ , meaning that there exists a bounded number  $K$  such that  $\Delta x < \epsilon \rightarrow |TE| < K\Delta x$ . The truncation error is always in the form  $TE = \mathcal{O}(\Delta x^q)$  where  $q$  is the order of accuracy. The forward finite difference approximation of  $\frac{\partial u}{\partial x}$  is first order accurate since  $q = 1$ . I

If the order of the method is larger, for example second order, it means that if  $\Delta x \rightarrow 0$ , after a certain  $\Delta x_{crit}$  the truncation error goes to 0 faster than the truncation error of a lower order method. If the mesh is not finer than a critical value, this is not true. When we go higher than a second order it is not clear in practice if we have something better because increasing the order allows the use of larger mesh, but is computationally expensive too.

The definition for the derivative is not unique, for instance the **backward difference formula**:

$$u_{i-1j} = u_{ij} - \Delta x \left. \frac{\partial u}{\partial x} \right|_{ij} + \frac{\Delta x^2}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} + H.O.T \Leftrightarrow \left. \frac{\partial u}{\partial x} \right|_{ij} = \frac{u_{ij} - u_{i-1j}}{\Delta x} + \mathcal{O}(\Delta x) \quad (1.4)$$

We have an infinity of finite difference formula if make the linear combination of the two last expressions. For example if we sum half of the two we get the **central finite difference formula**:

$$\left. \frac{\partial u}{\partial x} \right|_{ij} = \frac{u_{i+1j} - u_{i-1j}}{2\Delta x} + \mathcal{O}(\Delta x^2) \quad (1.5)$$

We see that the central difference formula is more accurate than the others and involves the same mesh distance. We could thus get as high order as desired, but at the cost of increasing the number of neighboring grid points in the equation and thus the computational cost.

### General method to obtain finite difference formulas

- Choose the stencil (set of points involved in the expression);
- write Taylor series expansion of all the points in the stencil around the point where the derivative is to be evaluated;
- write the finite difference formula as a linear combination of stencil point values and adjust the coefficients such that it approximates the derivative to be evaluated with the desired order of accuracy.

EXAMPLE: Let's compute the finite difference formula for  $\partial^2 u / \partial x^2$  using  $i-1j$ ,  $ij$  and  $i+1j$  (first step). The second step gives:

$$\begin{aligned} u_{i+1j} &= u_{ij} + \Delta x \left. \frac{\partial u}{\partial x} \right|_{ij} + \frac{\Delta x^2}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} + \frac{\Delta x^3}{6} \left. \frac{\partial^3 u}{\partial x^3} \right|_{ij} + \frac{\Delta x^4}{24} \left. \frac{\partial^4 u}{\partial x^4} \right|_{ij} + H.O.T \\ u_{ij} &= u_{ij} \\ u_{i-1j} &= u_{ij} - \Delta x \left. \frac{\partial u}{\partial x} \right|_{ij} + \frac{\Delta x^2}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} - \frac{\Delta x^3}{6} \left. \frac{\partial^3 u}{\partial x^3} \right|_{ij} + \frac{\Delta x^4}{24} \left. \frac{\partial^4 u}{\partial x^4} \right|_{ij} + H.O.T \end{aligned} \quad (1.6)$$

The third step gives:

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} = au_{i+1j} + bu_{ij} + cu_{i-1j} = (a+b+c)u_{ij} + (a-c)\Delta x \left. \frac{\partial u}{\partial x} \right|_{ij} + (a+c) \left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} + H.O.T \quad (1.7)$$



Depending on the accuracy needed, establish a system of 3 equations of 3 variables and solve by imposing the value for the different terms. For example if we want to approximate exactly we should cancel all the terms except the second order derivative term.

We can repeat this method again and again to obtain various finite difference formulas. You can consult pages 14 and 15 of the syllabus to see the list, not useful, just know that we can express the mixed second derivatives like  $\partial^2 u / \partial x \partial y$  too.

### Derivation of finite difference formulas using operators

In order to make the writing more compact, let's introduce some operators:

$E_x^{+1} u_{ij} = u_{i+1j}$	Forward shift	$E_x^{-1} u_{ij} = u_{i-1j}$	Backward shift
$\delta_x^+ u_{ij} = u_{i+1j} - u_{ij}$	Forward difference	$\delta_x^- u_{ij} = u_{ij} - u_{i-1j}$	Backward difference
$\mu_x u_{ij} = \frac{1}{2} (u_{i+\frac{1}{2}j} + u_{i-\frac{1}{2}j})$	Averaging	$\delta_x u_{ij} = u_{i+\frac{1}{2}j} - u_{i-\frac{1}{2}j}$	Centered difference

Another operator for the centered difference can be used:

$$\bar{\delta}_x = \frac{1}{2} (\delta_x^+ + \delta_x^-) \quad \Rightarrow \quad \bar{\delta}_x u_{ij} = \frac{1}{2} (u_{i+1j} - u_{i-1j}) \quad (1.8)$$

All these operators are also valid for  $y$  coordinate. The following relations are verified:

$$\delta^+ = E^{+1} - 1 \quad \delta^- = 1 - E^{-1} \quad \bar{\delta} = \mu \delta = \delta \mu \quad (1.9)$$

It is easy to derive finite difference formulas with these operator. For example, the Taylor series expansion of  $u(x)$  is:

$$\begin{aligned} u(x + \Delta x) &= u(x) + \Delta x \frac{\partial u}{\partial x}(x) + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2}(x) + \dots \\ \Leftrightarrow Eu(x) &= \left(1 + \Delta x D + \frac{(\Delta x D)^2}{2} + \dots\right) u(x) \quad \Leftrightarrow Eu(x) = \exp(\Delta x D) u(x) \end{aligned} \quad (1.10)$$

where we clearly see the Taylor expansion of  $\exp(\Delta x D)$  and where  $D_x = \frac{\partial u}{\partial x}$ . We can then make the following manipulation:

$$E = \exp(\Delta x \partial) \Leftrightarrow \ln(E + \mathbf{1} - \mathbf{1}) = \ln(1 + \delta^+) = \Delta x D \quad \Rightarrow \quad D = \frac{\ln(1 + \delta^+)}{\Delta x} \quad (1.11)$$

And finally if we make the Mac Laurin expansion:

$$D = \frac{\delta^+}{\Delta x} - \frac{\delta^{+2}}{2\Delta x} + \frac{\delta^{+3}}{3\Delta x} + \dots \quad (1.12)$$

By keeping the first term we find the first order forward difference formula, by keeping the second term we find the second order one, and so on.

#### 1.1.2 Finite difference formulas for partial differential equations

There is two strategy to express equations:

- **Strategy 1:** simply assemble the finite difference formula for each individual derivative;
- **Strategy 2:** same strategy used to find the finite difference in many steps, select the stencil, Taylor expansion on each point of the stencil, write the FD formula as a linear combination of the stencil points values and select the coefficients.

The first method is the most used. For the Laplace equation  $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$  we have:

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{ij} = \frac{\delta_x^2 u_{ij}}{\Delta x^2} + \mathcal{O}(\Delta x^2) \quad \left. \frac{\partial^2 u}{\partial y^2} \right|_{ij} = \frac{\delta_y^2 u_{ij}}{\Delta y^2} + \mathcal{O}(\Delta y^2) \quad (1.13)$$

If we sum this up we get:

$$\frac{\delta_x^2 u_{ij}}{\Delta x^2} + \frac{\delta_y^2 u_{ij}}{\Delta y^2} = 0 \quad \Rightarrow \quad \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{\Delta y^2} = 0 \quad (1.14)$$

The equation can contain a first order derivative and there can thus exist several discretization (forward, backward, ...).

### 1.1.3 Arbitrary geometries - irregular meshes

The method we have seen is very simple, we love it. But the expressions rapidly become very difficult when dealing with irregular meshes. In addition, the order of accuracy is lower when irregular meshes compared to the regular one with same size mesh. The formulas become intractable for more than 3 points. We cannot only use uniform meshes for at least two reasons:

- **Computational domain geometry:** when the boundary is curved, it is quasi impossible to use uniform rectangular mesh. On the aerofoil example below, one can see that the grid points not always intersect the nodes on the geometry.
- **Presence of regions where the solution varies rapidly:** for example, in fluid mechanics, there are regions such as the boundary layer where the fluid properties vary more rapidly than anywhere else. It is thus interesting to have finer mesh there and larger mesh somewhere where we don't care.

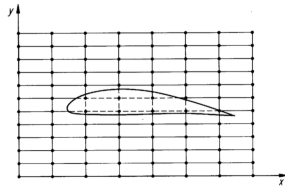


Figure 1.2

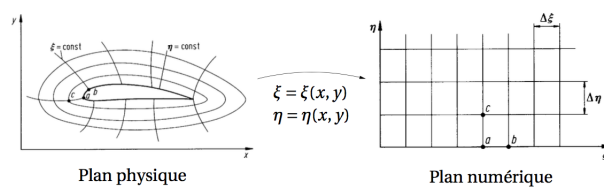


Figure 1.3

To tackle these problems, one can use coordinate transformation as suggests Figure 1.3. One can thus first fit a certain geometry, but also achieve a local concentration of mesh points. There are two disadvantages to this: we transform the geometrical complexity into equation complexity, and it is very difficult to find these transformation (numerical methods needed). The good news are that in the numerical plane, the mesh is regular and numerical algorithms have high efficiency, the transformation will be discussed later.

## 1.2 Finite volume method

The main idea is to take advantage of conservation equations whose fundamental form is the **integral form**, we discretize the integral. The principle consist in the application of the control volume method (macroscopic balances) in a large scale. The classical example is to have a bent tube, the flow exerts a force on the elbow and we can easily estimate it by momentum balance. We just take several small volumes where to apply this.

The great advantage of this method is to use arbitrary polygons (2D) or polyhedra (3D) as control volume so that it offers great **flexibility**. Unlike the finite difference method, the finite volume method can accommodate arbitrary control volume shapes, this eases mesh generation dramatically. There are some independent variables (time and space) that does not need flexibility it is the time variable, this is why we still use finite differences for time discretization. In addition, since the integral form is discretized, it allows the computation of **weak solutions** of the flow.

### 1.2.1 Fundamental principles and variants of the method

Let's consider the integral form of a general system of conservation equations:

$$U = \begin{pmatrix} \rho \\ \rho \vec{u} \\ \rho E \end{pmatrix} \Rightarrow \frac{\partial U}{\partial t} + \nabla \cdot \vec{F} = Q, \quad (1.15)$$

where  $\vec{F}$  is the **flux vector** and  $Q$  the **source term**. If we take the momentum equation and the conservation equation, we can see that there is a part independent of the derivative of  $U$  (convective term) and a diffusive term dependent of  $\nabla U \rightarrow \vec{F} = \vec{F}(U, \nabla U)$ :

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\underbrace{\rho \vec{u} \otimes \vec{u} + p \vec{1}}_{\text{convective}} - \underbrace{\bar{\tau}}_{\text{diffusive}}) = \rho \vec{g} \quad (1.16)$$

The corresponding integral form is the basic original form obtained by integration of the equation over a control volume  $\Omega$ :

$$\frac{d}{dt} \int_{\Omega} U d\Omega + \oint_{\partial\Omega} \vec{F} \cdot \vec{n} dS = \int_{\Omega} Q d\Omega \quad (1.17)$$

Remark that discontinuities are allowed in this integral form since we do not have to verify the differentiation everywhere in the domain. If we subdivide the domain in elementary volumes and use the average value of  $U$  on that volumes  $\int_{\Omega_k} U d\Omega = U_k \Omega_k$  these are chosen as the parameters of the discrete representation, and assume the control volume to be a polygon ( $\Gamma_m$  the faces), we have:

$$\frac{d}{dt} (\Omega_k U_k) + \sum_{\Gamma_m \in \partial\Omega_k} \int_{\Gamma_m} \vec{F} \cdot \vec{n} dS = \int_{\Omega_k} Q d\Omega \quad (1.18)$$

To make the discretization, we need to evaluate the remaining surface and volume integrals in terms of neighboring control volume averages. How to build the control volumes? First a mesh of non-overlapping elementary surfaces/volumes is generated, these are called cells. The design of control volumes must respect a certain number of conditions:

- the union of CVs must cover the whole domain of interest;
- the CVs may overlap but the boundaries of a CV should be either lying on the domain boundary or belong to the boundary of another CV. Each CV boundary must be shared by two CVs;
- the expression of the flux integral on a common edge should be the same for the two CVs it belongs to.

Consider two CVs K and L with a common face  $\Gamma_c$  and make the sum:

$$\begin{aligned}
& \frac{d}{dt}(\Omega_K U_K) + \sum_{\Gamma_m \in \partial\Omega_K} \int_{\Gamma_m} \vec{F} \cdot \vec{n} dS = \int_{\Omega_K} Q d\Omega \\
& \frac{d}{dt}(\Omega_L U_L) + \sum_{\Gamma_m \in \partial\Omega_L} \int_{\Gamma_m} \vec{F} \cdot \vec{n} dS = \int_{\Omega_L} Q d\Omega \\
\Rightarrow & \frac{d}{dt}(\Omega_K U_K + \Omega_L U_L) + \sum_{\Gamma_m \in \partial\Omega_K \cup \partial\Omega_L \setminus \Gamma_c} \int_{\Gamma_m} \vec{F} \cdot \vec{n} dS = \int_{\Omega_K \cup \Omega_L} Q d\Omega
\end{aligned} \tag{1.19}$$

where we can observe that the common boundary integral disappears since the flux should be the same but the normals are opposite to each others. This last property is called **telescopic property** that ensures the conservation at the discrete level and the capture of discontinuities. Indeed, if the flux was different on K and L, the term would remain.

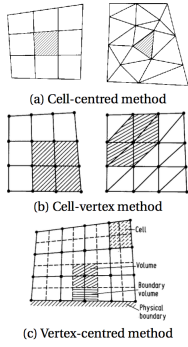


Figure 1.4

Several arrangement methods exists:

- the CV coincide with the mesh cell  $\rightarrow$  cell-centered method;
- the CV is made out of mesh cells having a common vertex  $\rightarrow$  cell-vertex method;
- the CV is made out of part of mesh cells sharing a common vertex  $\rightarrow$  vertex centered method.

In the two last ones, it is common to associate the volume average to the corresponding vertex as done in finite difference. Only these will be considered. Let's mention that it is not compulsory to use the same CVs for different equations of a system of equations.

### 1.2.2 Evaluation of fluxes through faces

In general we will approximate the integral over a length/surface  $\Gamma_m$  by a one point quadrature integration formula:

$$\int_{\Gamma_m} \vec{F} \cdot \vec{n} dS \approx \vec{F}_m \cdot \vec{n}_m S_m \tag{1.20}$$

This is sufficient for first order and second order methods, for higher order you have to use more points quadrature. Moreover, higher order are not easy to construct, this is why we have finite element methods. Let's discuss about aerothermodynamic problems ( $\vec{F} = \vec{F}(U, \nabla U)$ ) in 1D for simplicity. We will consider a vertex-centered 1D FV method and the CVs are segments. The discretization becomes:

$$\frac{d}{dt}(\Delta x_i U_i) + F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}} = 0 \tag{1.21}$$

How to express the  $F_{i+1/2}$  and the other in function of cell averages? One needs to specify a **numerical flux formula** which plays the same role as finite difference formulas in finite difference method. We will say that:

$$F_{i+\frac{1}{2}} \approx \Phi(U_{i-k+1}, \dots, U_{i+k}) \tag{1.22}$$

For the method to be at least of order one, the approximation should be exact for a uniform field  $\Phi(U, \dots, U) = F(U)$ . The simplest choice is to take an arithmetic average of the fluxes or of the variables:

$$\Phi(U_i, U_{i+1}) = (F_i + F_{i+1})/2 \quad \Phi(U_i, U_{i+1}) = F\left(\frac{U_i + U_{i+1}}{2}\right) \quad (1.23)$$

This applied to (1.21) gives:

$$\frac{d}{dt}(\Delta x_i U_i) + \frac{F_i + F_{i+1}}{2} - \frac{F_{i-1} + F_i}{2} = 0 \quad \Rightarrow \quad \frac{dU_i}{dt} + \frac{F_{i+1} - F_{i-1}}{2\Delta x_i} = 0 \quad (1.24)$$

which is the same expression as obtained by central finite difference formula. One can retrieve the first order forward and backward finite difference formula by choosing  $\Phi(U_i, U_{i+1}) = F(U_{i+1})$  and  $\Phi(U_i, U_{i+1}) = F(U_i)$ .

Let's come back to the nature of the finite volume numerical representation. It consists of a set of average values over subdomains and is thus clearly a discrete representation. For cell-centered or vertex-centered methods (not overlapping) it is easy to reconstruct a functional representation out of the averages. A piecewise constant reconstruction and a linear reconstruction are illustrated below.

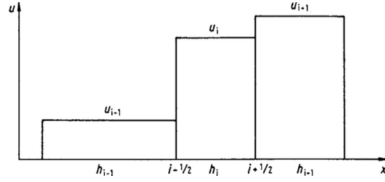


Figure 1.5

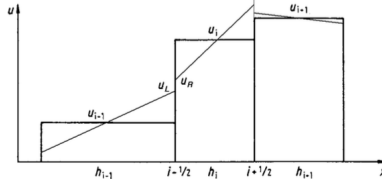


Figure 1.6

In the second, the solution gradient is estimated in each CV but the reconstruction remains discontinuous at the boundaries so that it does not eliminate the need for a numerical flux function to compute the flux across the boundaries. But it allows to easily construct more accurate flux functions, starting from a two variable flux function:

$$F_{i+\frac{1}{2}} \approx \Phi(U_i, U_{i+1}) \quad (1.25)$$

associated with the constant reconstruction, one obtains more accuracy by replacing  $U_i$  and  $U_{i+1}$  by  $U_L$  and  $U_R$ . For instance using the backward flux formula  $\Phi(U_L, U_R) = F(U_L)$  and a gradient estimation in CV i based on the back point:

$$\left(\frac{\partial U}{\partial x}\right)_i \approx \frac{U_i - U_{i-1}}{\Delta x} \quad \Rightarrow \quad U_{L,i+\frac{1}{2}} = \frac{3}{2}U_i - \frac{1}{2}U_{i-1}, \quad (1.26)$$

One can obtain the following space discretization:

$$\frac{dU_i}{dt} + \frac{1}{\Delta x} \left( F\left(\frac{3U_i - U_{i-1}}{2}\right) - F\left(\frac{3U_{i-1} - U_{i-2}}{2}\right) \right) = 0 \quad (1.27)$$

and for a particular  $F(U) = au$ , we have:

$$\frac{dU_i}{dt} + a \frac{3U_i - 4U_{i-1} + U_{i-2}}{2\Delta x} \quad (1.28)$$

which is the one we found in previous section. Generally, for a polynomial reconstruction of order  $k$  we shall obtain a discretization of order at least  $k + 1$ . For the diffusive fluxes we have to estimate the gradient of variables on the faces which can be done directly or by averaging the estimated gradients in the two neighboring CVs. This is done by Green-Gauss theorem:

$$\int_{\Omega} \nabla U \, d\Omega = \oint_{\Gamma} U \vec{n} \, dS \quad (1.29)$$

by choosing an auxiliary control volume  $\Omega$  centered on the point where one wishes to estimate the gradient.