Chapter 1

Introduction

1.1 Overview

The main purpose of this work is to contribute to the development of numerical techniques for computational heat transfer and fluid flow, suitable for low cost parallel computers[1].

It is focused on implicit integration schemes, using finite control volumes with multigrid algorithms. Thus, our scope is in the intersection of heat transfer, fluid dynamics, multigrid algorithms and parallel computing.

Heat transfer and fluid dynamics are critical aspects for many scientific and engineering applications. Some of them are well known, such as heat exchangers [1], thermal energy storage [2, 3], active [4] and passive [5] solar energy, contaminant transport [6], combustion [7], weather prediction [8, 9], etc. Others are more recent, such as estimation of the impact of human activity in climate [10]; medical applications like blood circulation models [11] or microbiology [12, 13, 14].

In many of these applications, advances could be used to enhance the living conditions of many persons. Consider, for instance, the potential in energy saving by means of improving the passive thermal behaviour of buildings or the possibility to develop cheap high temperature solar collectors that could be used for solar energy based refrigeration systems. Both are mainly heat transfer and fluid flow problems. In both cases, the capability to predict correctly the behaviour of the new designs from their specifications, with a reduced number of full scale experiments, is crucial to reach better designs. The only way to do it is to formulate and solve the governing equations that describe their behaviour.

The partial derivative equations governing fluid dynamics, Navier-Stokes equations, have been formulated for more than a century and a half[2]. However, they are so complex that analytical solutions (this is, that can be expressed in terms of primitive functions and evaluated without a computer) are in general restricted to idealized conditions. So, until the development of modern digital computers and the numerical analysis applied to fluid dynamics from the 50s, the Navier-Stokes equations were mainly of theoretical interest. This is why traditional fluid dynamics had two different and distant approaches: experimental and theoretical, sometimes leading to apparently paradoxical situations [15, section II]. Only recently, when CFD has allowed the solution of realistic, time-dependent flows, both approaches have been unified. In many cases, CFD has allowed the direct use of the governing equations, without simplifications, as tool for contemporaneous engineering use. Thus, concepts from partial differential equations theory and numerical analysis have become common tools for engineers.

Nevertheless, not all the problems of interest can be solved yet using CFD. This ultimate stage of

[1] For shortness, the term Computational Fluid Dynamics and its acronym CFD will be used to refer to numerical fluid dynamics and heat transfer. The acronym PCFD is sometimes used to refer to Parallel Computational Fluid Dynamics, this is, CFD carried out with parallel computers. There is a list of all the acronyms used in appendix C.

[2] By Navier in 1822 and Stokes in 1845 that independently completed Euler's model, formulated around 1755, with viscosity effects.

CFD will probably never be achieved. Experimental techniques, not directly used in this dissertation, are still very important.

Multigrid algorithms were introduced in the 60s by Russian scientists, [16, 17] and they began to be important in the 80s, mainly after the works of Brandt (e.g., [18, 19]). Among sequential algorithms to solve PDEs, multigrid is one of the best options. However, the efficient implementation of multigrid (MG) algorithms on parallel computers is a difficult task, specially for the case of *loosely coupled* systems (section 4.2.1), such as clusters of workstations. Different types of MG algorithms are considered in this work. An important part of the numerical work done in CTTC is based on the ACM variant, so the main target of this work was to provide a way to parallelize it, using affordable computers. The majority of the examples of this work have been executed using the JFF cluster (appendix B).

Parallel computers are already important for scientific computation and probably will become even more important in the near future. The majority of the work currently done for development of new numeric algorithms is somehow related to parallel computing, or at least has parallel processing in mind.

One of the main reasons for the renewed interest for parallel algorithms in the context of scientific computing is the emergence of a new class of parallel computers: the Beowulf clusters (http://beowulf.gsfc.nasa.gov). They are a group of low cost (PC class) computers, running Linux, connected through a dedicated network. For the price typically paid for a fast workstation in the mid 90s, it is now possible to buy an Intel based cluster of about 20 nodes. Both the low cost of the hardware and the availability of a credible operating system for the Intel 386 architecture[3], capable to compile and execute existent numerical codes, have been crucial to promote these affordable clusters. Nevertheless, compared with the traditional parallel computers, they have additional problems that are to be overcomes to be able to extract their potential.

Outline of the dissertation:

- This section is a short summary of the physical models that are to be used.
- Section 2 is an overview of different numerical analysis techniques used in CFD. The main difficulties for parallel computing are introduced.
- Section 3 presents different variants of the multigrid algorithm, pointing out the advantages of ACM. A detailed description of segregated ACM is given. A turbulent natural convection flow is presented as an application example. A short description of the coupled ACM variant is given.
- Section 4 provides some background information of parallel computing technology, discussing the key aspects for its efficient use in CFD and the limitations of low cost parallel computers: high latency and low bandwidth. An overview of different control-volume based PCFD and linear equation solvers is done. As an example, a code to solve reactive flows using Schwartz Alternating Method that runs particularly well on Beowulf clusters is given.
- Section 5 shows that the main problem associated with parallel MG is the latency. DDV, a latency-tolerant geometric multigrid algorithm (originally described in [20]) is presented, highlighting the modifications introduced in this work to deal with two-dimensional domain decompositions.
- Section 6 gives a detailed description of the Schur complement algorithm, to be used as an auxiliary procedure for DDACM algorithm. An enhanced version, specific for situations with constant matrices, such as pressure-correction equations is presented.

[3] This is, the Intel 386, 486, the different Pentiums, and compatible microprocessors by AMD and other vendors. However, as Linux runs also in other architectures, future Beowulf clusters could use other processors such as the Alpha family if they become cost effective. In the long term, clusters of very low cost, single purpose computers, such as video games consoles, could be used if an appropriated operating system is available.

- Section 7 uses the results obtained in previous sections to develop a latency-tolerant, algebraic algorithm called DDACM.
- Section 8 provides a discussion of the results obtained and points out areas for further research in parallel multigrid algorithms.

Since aspects related with different disciplines are considered in the different sections, one of the main aims has been to provide a document readable by people from different backgrounds. Thus, different sections might be of little interest for different persons. Parallel computing people can safely skip sections 4.2 and 4.3; Multigriders section 3.1; CFD and heat transfer people, the rest of this chapter and section 2.

1.2 Problem model

Incompressible, time-accurate Navier-Stokes (NS) equations plus energy transport equation will be used as a *problem model*:

$$\nabla \cdot \mathbf{u} = 0 \tag{1.1}$$

$$\frac{\partial u_1}{\partial t} + \mathbf{u} \cdot \nabla u_1 = -\frac{1}{\rho}\frac{\partial p}{\partial x_1} + \nu \nabla^2 u_1 + f_1 \tag{1.2}$$

$$\frac{\partial u_2}{\partial t} + \mathbf{u} \cdot \nabla u_2 = -\frac{1}{\rho}\frac{\partial p}{\partial x_2} + \nu \nabla^2 u_2 + f_2 \tag{1.3}$$

$$\frac{\partial u_3}{\partial t} + \mathbf{u} \cdot \nabla u_3 = -\frac{1}{\rho}\frac{\partial p}{\partial x_3} + \nu \nabla^2 u_3 + f_3 \tag{1.4}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \frac{k}{\rho c_p}\nabla^2 T + \frac{Q_v}{\rho c_p} \tag{1.5}$$

where the independent variables are the position vector $\mathbf{x} = (x_1, x_2, x_3)$ and the time t. The unknowns are: u is the velocity vector $\mathbf{u} = (u_1, u_2, u_3)$, T is the temperature and p is dynamic pressure. The (known) parameters are: the thermal conductivity k, the kinematic viscosity ν , the constant pressure specific heat c_p , the density ρ and the energy transferred to the medium[4] Q_v . Q_v is usually a (given) function of $\mathbf{x}, t$. In *natural convection* problems, the body force vector $\mathbf{f}$ is a function of local density and thus of temperature.

The equations are respectively a balance of mass, momentum in each direction and energy. Their deduction from the conservation principles plus Stokes and Fourier constitutive laws can be found elsewhere (e.g., [21, 22, 23, 24]). This set of equations, governing for instance natural convection flows, is not only a popular benchmark for numerical methods [25, 26, 27] but also has questions to be answered for the case of turbulent flows. Natural convection in closed cavities has been the subject of numerous research, e.g., [28, 29, 30, 27, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41].

Scope of the model This problem model is known to describe fluid flow and convection-conduction heat transfer phenomena under the following assumptions:

- The flow is single-phase, single-component, incompressible and non-reactive.
- Physical properties are constant, except for the density variations for which the Boussinesq approximation has been used. If x_3 is in the vertical axis, the expression of $\mathbf{f}$ is then

  $$f_1 = f_2 = 0 \qquad f_3 = g\beta\,(T - T_0) \tag{1.6}$$

  where $\beta = -\frac{1}{\rho}\frac{\partial \rho}{\partial T}$ is the thermal volumetric expansion coefficient.

[4] Due to other processes not modeled by the set, such as Joule effect or chemical reaction

- Viscous dissipation has been neglected.
- Thermal radiation is neglected.
- The fluid is Newtonian.

The sentence “laminar flow” was traditionally included in the hypothesis assumed in the model (1.1-1.4). Nowadays, it is usually considered that this model also represents turbulent flows as well (for any Re and/or Ra number) as long as care is taken to obtain grid independent numerical solutions. This is known as the direct numerical simulation approach, section 1.3. Of course, depending on the situation, this can either be very difficult or impossible, but equations are not to blame for that: they would give the right answers *also* for turbulent flows if we could solve them.

From a computational point of view, the assumption of *two-dimensional* flows,

$$\frac{\partial \mathbf{\Phi}}{\partial x_3} = 0 \tag{1.7}$$

where $\mathbf{\Phi}$ is a vector that groups all the unknowns, $\mathbf{\Phi} = (\mathbf{u}, p, T)$, is a very interesting simplification of model as two-dimensional flows retain the main difficulties of the three-dimensional problems (as formulated in this section), but are easier to visualize and understand, and require less computational resources. This is why they have been used for many examples in this work.

The role of the problem model Equations (1.1-1.5) are considered a problem model (and not the *real* problem) denoting that they do not contain *all* the phenomena to be found in CFD and heat transfer but they are representative of the majority the difficulties to be found. For instance, the assumption of constant physical properties is not always possible. It reduces the non-linearity of the set, but in general, a more accurate treatment is (usually) not a problem as the dominant source of non-linearity is already present in the convective terms $(\mathbf{u} \cdot \nabla u_i)$. However, our goal is not just to solve equations (1.1-1.5): our numerical tools should allow the easy extension of the models, maybe at a higher CPU cost, but without much trouble. Examples of other models of interest are given in section 1.4.

On the other hand, there are important classes of *easier* problems in which it is not necessary to solve the complete Navier-Stokes set. For instance, Euler's equations, are easier to solve because they neglect the terms involving friction and thermal conduction. This is, the terms $\nu \nabla^2 u_i$ and $\frac{k}{\rho c_p} \nabla^2 T$ are assumed to be null.

Analytics solutions of the problem model As aforementioned, up to date, analytical solutions of the set (1.1-1.5) can only be found in idealized or exceptional situations as the example given in section 1.3.1. A summary of the “classic” solutions can be found in [24, 21]. A transient, three-dimensional analytic solution, with boundary conditions that do not correspond to any technically relevant problem is presented in [42]. It has been constructed mainly as a benchmark for CFD codes. Unfortunately, no major advances are to be expected in this area.

The difference between numerical and analytical solutions is not allways totally clear. In a certain sense, analytic PDE solutions are also numerical, as they usually involve complex expressions such as infinite series that also requiere a computer for their evaluation with the desired (allways finite) accuracy. Even if only the usual primitive functions (e.g., exponential or sinus) are needed to express the solution, the computer evaluates them approximately, using polynomials or other numerical techniques.

The main advantage of analytic solutions (when they are treatable), is that they can be directly used to discuss the results as a function of the governing parameters, or to optimize designs, etc.

Boundary conditions Different boundary conditions are to be imposed depending on the situation to be solved. Specification of well-posed boundary conditions that model the physical system correctly is a difficult problem. Typical (easy) examples are:

- For the mass conservation and momentum equations, non-slip ($\mathbf{u} = 0$) conditions are imposed to the velocity in the solid contours and known velocity profiles in the inlets $\mathbf{u} = \mathbf{u}_{in}$.
- For the energy equation, two frequent boundary conditions are: known temperature $T = f(\mathbf{x}, t)$ and known heat flux $-k\frac{\partial T}{\partial n} = f(\mathbf{x}, t)$. The particular situation of *adiabaticity*, $\frac{\partial T}{\partial n} = 0$ is frequently used as a model.

  The heat flux can be function of the boundary temperature. For instance, if the boundary of the domain is a solid and the outer parts of the domain (not modeled) are a fluid medium surrounded by other solids, expressions such as $-k\frac{\partial T}{\partial n} = h\left(T - T_g\right)$ are typically used. Here T_g is a representative temperature of the fluid medium, in a point not close to the boundary and h is a heat transfer coefficient, that can itself be a function of the boundary temperature [43].

As a more complex example, consider natural-convection induced flows in open cavities [44, 43].

Other formulations The set of equations (1.1-1.5) is expressed in terms of the *primitive variables* pressure and velocity. In two dimensions, the elimination of pressure from the two momentum equations leads to a stream-function / vorticity formulation for the continuity and momentum equations. The main advantage of this formulation is that the continuity equation is identically satisfied. See for instance [45] as an example. It has been extended to three dimensions (see, for instance, [46, 47]).

However, specially in the three-dimensional case, the advantages of these formulations are perhaps not enough to compensate their lack of a clear physical meaning and the difficulties associated with the boundary conditions.

With reference to other formulations of the Navier-Stokes set, it is also interesting to point out that all the transient formulations (see for instance the formulation used in [47]), contain a continuity constraint without time derivative (like equation 1.1), that will be discussed, is a source of problems if parallel computers are used for the integration of the set (1.1-1.5).

Other notations For brevity, a vector equation is usually used to express the set of momentum equations (1.2-1.4),

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \tag{1.8}$$

but, as each component of the velocity is an unknown to be solved, from a computational point of view the scalar expression of the equations is perhaps clearer. To describe certain algorithms (like Newton-Raphson method in section 2.5), the opposite is true, and a single vector is used for *all* the unknowns.

Dimensionless forms For many reasons, it is profitable to express the model problem in dimensionless form. Here it is used only to introduce different dimensionless groups used in the dissertation. Consider the following set of dimensionless independent variables and unknowns:

$$\begin{aligned} \mathbf{x}^* &= \mathbf{x}/L \\ t^* &= t/\left(L^2/\alpha\right) \\ \mathbf{u}^* &= \mathbf{u}/\left(\alpha/L\right) \\ T^* &= \left(T - T_0\right)/\Delta T \\ p^* &= \left(p - p_0\right)/\left(\rho\alpha^2/L^2\right) \end{aligned} \tag{1.9}$$

where $\alpha = \frac{k}{\rho c_p}$ is the thermal diffusivity, ΔT is a reference temperature increment (i.e., the temperature difference between the walls of the domain, and L is a representative length).

If they are introduced into equations (1.1-1.5), using expression (1.6) for the body forces and assuming that $Q_v = 0$, we obtain:

$$\nabla \cdot \mathbf{u}^* = 0 \tag{1.10}$$

$$\frac{\partial u_1^*}{\partial t^*} + \mathbf{u}^* \cdot \nabla^* u_1^* = -\frac{\partial p^*}{\partial x_1^*} + Pr \nabla^{2*} u_1^* \tag{1.11}$$

$$\frac{\partial u_2^*}{\partial t^*} + \mathbf{u}^* \cdot \nabla^* u_2 = -\frac{\partial p^*}{\partial x_2^*} + Pr \nabla^{2*} u_2^* \tag{1.12}$$

$$\frac{\partial u_3^*}{\partial t^*} + \mathbf{u}^* \cdot \nabla^* u_3^* = -\frac{\partial p^*}{\partial x_3^*} + Pr \nabla^{2*} u_3^* + PrRaT^* \tag{1.13}$$

$$\frac{\partial T^*}{\partial t^*} + \mathbf{u}^* \cdot \nabla^* T^{2*} = \nabla^2 T^* \tag{1.14}$$

Note that the starred vector operators are referred to dimensionless position vector, i.e. $\nabla^* = L\nabla$ and $\nabla^{2*} = L^2\nabla^2$. The expressions (1.9) are a typical choice for natural convection. Of course, other sets of dimensionless variables can be used.

There are only two parameters[5] in the new version of the equations, Prandtl number and Rayleigh number:

$$Pr = \frac{\nu}{\alpha} \tag{1.15}$$

$$Ra = \frac{\beta \Delta T L^3 g}{\alpha \nu} \tag{1.16}$$

They are called *dimensionless governing numbers*. They are the only parameters needed to reproduce the behavior of the flow (for a given geometry and boundary conditions). This is, $\mathbf{\Phi}^* = (\mathbf{u}^*, T^*, p^*) = f(\mathbf{x}^*, \mathbf{t}^*, Pr, Ra)$.

Pr depends only on the fluid properties while Ra, that governs the intensity of the natural convection, depends also on g, ΔT and L^3. All the sets of possible parameters with the same Pr and Ra are equivalent with respect to equations (1.10-1.14) and thus to equations (1.1-1.5) if the dimensional variables are recovered.

For forced convection, $\mathbf{u}^*$ value is defined from U, a reference (known) velocity of the flow, such as an inlet velocity, and Re number appears in the dimensionless equations as:

$$Re = \frac{LU}{\nu} \tag{1.17}$$

It can be interpreted as a measure of the degree of non-linearity of a flow [48].

Implications of incompressibility The assumption of incompressibility, (i.e., constant density $\frac{\partial \rho}{\partial p} = 0$), at first glance looks like a simplification, but it actually causes troubles to the numerical solution of equations (1.1-1.5). It has two major effects over mass conservation equation:

- p is not directly present in mass conservation equation, so there is no equation from which to solve p.
- As sound velocity can be expressed as

  $$c = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_T} \tag{1.18}$$

  incompressibility is equivalent to assume that sound waves propagate at infinite velocity. An equivalent argument is that, as there is no transient term in mass conservation equation, there may be local changes affecting all the domain for each time step, even if it is small. These are bad news for parallel computing. Additionally, fully explicit algorithms are not possible for incompressible flows, as discussed in section 2.4.2.

[5] Instead of 8 parameters (ν, ρ, k, Cp, ΔT, L, g, β).

Incompressibility is, of course, an approximation, but a good one for many heat transfer problems. The conditions under which flows can be considered incompressible can be found for instance in [49]. We can ignore the strict incompressibility condition and use general compressible algorithms for our almost perfectly incompressible flows. However, this is not an important difference on many cases: mass conservation equation is still a long range equation, compared with the rest of the set. Consider for instance typical indoor natural convection air flows, where fluid velocity $|\mathbf{u}| \approx 0.1 - 10m/s$ while sound velocity $c \approx 340m/s$.

1.3 Laminar and turbulent flows. Turbulence modeling.

1.3.1 Turbulence

Lagrange[6], in 1788 wrote: “One owes to Euler the first general formulas for fluid motion[7] ... presented in the simple and luminous notation of partial differences ... By this discovery, all fluid mechanics was reduced to a single point of analysis, and if the equations involved were integrable, one could determine completely, in all cases, the motion of a fluid moved by any forces ...”.

Even though nowadays we know that Lagrange’s claim is not correct, as the assumption of null viscosity in Euler equations leads to wrong predictions in many cases [15, section I], the core of the question still holds: can all the fluid dynamics, including turbulent flows, be reduced to the integration of Navier-Stokes[8] equations ?

To answer it, experimental results must be contrasted with solutions of NS equations. Consider, as an example, the flow through a long straight tube (Poiseuille flow). This example is important, because if steadiness and symmetry are assumed, this situation is one of the few with analytic solution. Expressed in cylindrical coordinates:

$$u_x = a\left(c^2 - r^2\right); u_r = u_\theta = 0 \tag{1.19}$$

where u_x, u_r and u_θ are the components of the velocity vector.

This solution is in good agreement with the experimental solution for *laminar* flows. However, if velocity is increased, for $Re = \frac{DU}{\nu} > 2300$, the observed flow is *turbulent*: it is neither steady nor symmetrical. Here, D is the diameter of the tube and U is the time and space average of u_x. The exact value of the critical Reynolds number depends on many parameters such as the polishing of the tube or the inlet velocity profile.

This discrepancy does not prove that the NS equations are wrong. The solution of the puzzle is that our analytic integration was wrong. There is a subtle mistake: if we assume steadiness and symmetry, we get steadiness and symmetry. But this does not mean that there are not other non-steady, non symmetrical, three-dimensional (turbulent) solutions of the equations.

For low Re numbers, the solution is stable: small non-symmetric perturbations (such as changes in the inlet profile) decay and the flow recovers its steadiness and symmetry. But beyond the stability limit, the perturbations do *not* decay and the flow becomes turbulent. Quoting [15], “Although symmetric causes must produce symmetric effects, nearly symmetric causes need not produce nearly symmetric effects: a symmetric problem need have no stable symmetric solution”. Another example, with no analytic solution, is presented in section 3.3.3: a problem with steady, symmetric boundary conditions has a non symmetric non-steady solution. However, the same type of “error” is made there: a two dimensional flow is assumed while it is actually three-dimensional. Three dimensional perturbations would grow, leading into a three-dimensional flow. No assumptions of steadiness, null space derivatives or symmetry can be made in the integration of the NS equations, unless it is positively known that the flow is steady, two dimensional or symmetric.

Nowadays, it is commonly agreed that both laminar and turbulent flows are subject to the same mass, momentum and energy conservation, so both are governed by the same equations. Thus,

[6] This quote can be found in [15, section I].

[7] Euler’s equations, section 1.2.

[8] In its range of applicability (continuous hypothesis).

integration of equations (1.1-1.5) subject to the appropriated initial and boundary conditions can predict the physics of the turbulent flows. Or, at least, their time averaged values. Long term time-accurate solutions are impossible. Unavoidable errors in initial conditions or in the numerical integration process, necessarily done with finite precision, act as perturbations and lead to errors of amplitude growing with time. An example is given in section 3.3.3.

The main charateristics of turbulent flows are:

- They are non-symmetric, three-dimensional and unsteady (even if the experiment is charaterised by symmetric, two-dimensional and steady boundary conditions, as in the case of a natural convection).
- Chaotic.
- Characterized by a large number of three-dimensional vortex elements (eddies) varying in size and fluctuating over a large range of spatial and temporal frequencies.

It has been said [48] that "Understanding turbulent flows is a *grand challenge* comparable to other prominent scientific problems such as the large-scale structure of the Universe or the nature of subatomic particles".

If turbulence is just the general solution of the Navier-Stokes equations, the expressions "understand turbulent flows" and "integrate the Navier-Stokes equations" could be considered equivalent. Turbulence is a complex structure arising from the apparently "simple" non-linear Navier-Stokes equations, like for instance the Mandelbrot set [50] (or similar structures), that also contain information in a wide range of scales. Non-linearity produces these annoying but beautiful effects, as we can see in many situations. Chaotic dynamics and complexity seem to be a consequence of mathematics and hence appear in a broad range of physical systems [51].

1.3.2 Turbulence modeling

Turbulence is not just important from a scientific or mathematic point of view. There is a huge interest to solve [9] turbulent flows. This is because, in many cases, flows of interest in engineering are turbulent. If NS equations could be solved for turbulent flows like they are now solved for laminar flows, countless fields of engineering would be benefited. To do so, two different approaches are currently in use:

- **DNS**. The direct use of the governing equations (i.e., 1.1-1.5), without any modeling assumption is known as Direct Numerical Simulation (DNS).

  The main problem of this approach is that the ratio of the length of the domain to the smallest scales of motion is proportional to $Re^{3/4}$, which implies that the number of grid points required grows at least with $Re^{9/4}$. The ratio of the largest sale motion to the smallest scale motion is proportional to $Re^{1/2}$ [52]. With the need of more time steps and a higher operation count (due to the increasing cost of the solution algorithms, as will be seen in this work), the simulation cost has been estimated to be at least proportional to Re^4 [53] [10].

  So, due to the limitations of current computers *and* current numerical methods, this approach is still not apt for engineering purposes. And, for arbitrarily large *Re* numbers it will never be possible unless computers capable of doing an arbitrarily large number of operations per second were available.

- **Turbulence modeling**. The standard approach to the turbulence problem is known as *turbulence modeling*. The goal is to be able to predict the averaged physics of turbulent flows without solving all the spatial and temporal scales implicated. Huge efforts have been devoted to turbulence modeling. There are two classes of approaches:

[9] On an averaged sense or to be able to calculate short time accurate predictions.
[10] Other estimations can be found, see for instance reference [54], section 9.2.

- RANS method. A time average operator is applied to the NS equations, and the instantaneous values are expressed as mean values plus fluctuations. By doing so, the *Reynolds Averaged Navier-Stokes* equations are obtained. They are formally similar to equations (1.1-1.5) but include additional terms that can not be solved without the previous knowledge of the fluctuating quantities. To *close* the RANS equations, many approaches which require experimental parameters have been proposed. However, unfortunately, [55, 56] none of them can be considered as an accurate and universal model for all flows. Usually, but not necessarily, the RANS method is used to obtain steady-state flow solutions. So, from a computational point of view, we can imagine RANS equations as steady-state NS equations augmented by additional non-linearities and transport equations.
- LES method. The basic idea of the *Large Eddy Simulation* method is to compute the large fluid flow structures and model only the small structures which can not be solved due to the lack of computational resources. The approach used is based on a space-filtering of the NS equations. The turbulence model used to describe the scales that are not computed can be much simpler and universal that the models used in the case of RANS. However, as space (and not time, like in RANS) filtering is used, a complete time-accurate simulation of the flow problem has to be done. Time filtering is done a posteriori, once the solutions have been obtained.

For both LES and RANS, the accuracy of the mathematical models of turbulence is not established for all the situations of interest. Before using the numerical simulations of turbulent flows, the turbulence models used should be verified experimentally.

Due to the difficulty in obtaining all the data that would be of interest to develop and validate new turbulence models, sometimes the models are compared with DNS results rather than actual experiments. This is a reason to advance in DNS, in spite of the pessimistic predictions about the simulations cost.

1.3.3 Laminar and turbulent flows from a computational point of view

A possible classification of the incompressible flows described by equations (1.1-1.5), respect to the CFD methodology to be applied for their solution could be:

- **Laminar flows**, which can currently be solved either in steady state or transient problems, unless there are additional difficulties such as complex geometry, coupling with other phenomena, reactive flows, non-Newtonian flows, etc.
- **Turbulent flows**, which are currently difficult to solve (in general terms). From the numerical analysis point of view, two subgroups can be identified: RANS and LES/DNS. In both cases, the computational effort is much higher than in the case of laminar flows, but the difficulties are different:
  - **RANS models**. The set of PDEs expressing RANS models has additional non-linearities, not present in the original NS equations. The main problem is the robustness (capability to obtain a convergent solution) of the solution algorithms. RANS models are used to solve steady-state problems or the larger time scales of transient problems (not filtered by the temporal average) so the number of time steps to be solved is relatively small (compared with LES/DNS problems).
  - **LES models and DNS**. The main problem here is not the non-linearity but the solution of fast transients with large (or very large) meshes. The ability to deal efficiently with the non-linear terms is not so important, as the time steps are short and the solution of a time step is always a very good initial guess for the next one. The main emphasis is to compute the solution of each time step as fast as possible. The most important part of the effort is spent in the implicit solution of mass conservation equation (sections 1.2 and 2.4.2).

1.4 Additional heat transfer phenomena missing in the problem model

As mentioned in section 1.2, our problem model contains only a part of the difficulties found in numerical heat transfer and CFD. Other problems as well as their computational implications will be concisely described[11]:

- Mixed solid-fluid domains. Unlike pure fluid flow problems, in heat transfer problems it is quite frequent to need the simultaneous solution of temperature fields in solids as well as in fluids. Frequently [57, 44, 58], the domain is treated as a continuous medium with areas of very high viscosity (the solids). This is not a problem for SIMPLE-like methods [59] but it can be for projection [60] and other algorithms requiring an explicit treatment of boundary conditions with given boundary conditions (non-slip conditions, in particular).

- Solid-liquid phase change is an important process with many applications, mainly related to energy storage and/or decrease temperature fluctuations [2, 43]. It is essentially a mixed solid-fluid domain with moving interface. Energy equation is augmented with an additional model to include latent heat storage. A detailed description of such models can be found in [44].

- Mixed gas-liquid domains and free-surfaces. For certain applications, [61] it is important to be able to solve the position of the surface. This is a considerable complication. There are two approaches. In the first (VOF-like methods), the problem is again treated as a continuous domain and a variable is used to express the fraction of each control volume occupied by one of the fluids. The other approach uses an explicit boundary condition and tracks the position of the surface remeshing the domain after each time step.

- Multi-component fluid flows are important in many scientific and engineering applications such as material processing or gaseous contaminant transport [62, 63, 64, 65, 66]. Equations (1.1-1.5) are augmented with *mass transport equation*, that is formally identical to energy equation but with different diffusivities. Sometimes, the concentration gradients induce buoyancy forces that are to be considered in the body force vector $\mathbf{f}$. This is the so-called *double-diffusion problem* [6].

- Reactive flows. In many applications, multi-component flows are also reactive. In this case, the concentrations of the different components interact. Consider for instance the chemical processes suffered by atmospheric contaminants. Another important example of these problems is combustion simulation [7]. This problem will be considered as an example of application of a parallel CFD algorithm in section 4.7.

- Porous media. There are many industrial areas in which it is necessary to deal with porous flows. Among them, ground water hydrology, radioactive waste management, thermal insulation or building heat transfer. A summary of different models for saturated porous media can be found in [67].

- Radiation heat transfer. Although sometimes it is negligible, radiation heat transfer is always present so it is perhaps the most important element missing in our problem model. It can be classified into two categories:
  - Non-participant medium radiation. Radiation heat transfer has to be evaluated only at the solid-fluid interfaces. Radiosity method is usually used [44]. In the solid-fluid boundaries the surface heat balances are modified to include the radiation heat exchange.
  - Participant medium radiation is important for many applications such as combustion, atmospheric interaction of solar radiation (important for climate change prediction) or transparent insulation. In this case, a radiation transfer equation has to be solved for each point of the domain. The negative divergence of the heat radiation is added to the Q_v term of energy equation [43, 68].

[11] Turbulence modeling has already been discussed in section 1.3

> The computational implications of radiation modeling are important, specially when parallel computers are to be used. This is because radiation, as well as incompressibility, has the annoying feature of coupling all the points of the domain at each time step.

There are many important problems still missing in this list (like for instance, non-Newtonian fluids or interactions with electromagnetic fields). Only the models already implemented, or to be implemented soon[12] in the code DPC have been outlined here.

[12] Porous media flow.

1.5 Nomenclature

c	sound velocity
c_p	specific heat
$\mathbf{f}$	body force vector
h	heat transfer coef.
p	dynamic pressure
k	thermal conductivity
Q_v	heat generation
g	gravity acceleration
p_0	reference pressure
Pr	Prandtl number
Ra	Rayleigh number
Re	Reynolds number
t	time
T	temperature
T_0	reference temperature
ΔT	reference temperature difference
T_g	medium temperature
$\mathbf{u}$	velocity vector
U	reference velocity
$\mathbf{x}$	position vector

Greek symbols

α	thermal diffusivity
ν	cinematic viscosity
ρ	density
β	thermal volumetric expansion coef.
$\mathbf{\Phi}$	vector with all the unknowns
k	thermal conductivity

Subindices

1,2,3	Cartesian components
x	axial velocity component
r	radial velocity component
θ	angular velocity component

Superindices

$*$	dimensionless