THE FINITE-VOLUME METHOD APPLIED TO COMPUTATIONAL RHEOLOGY: II- FUNDAMENTALS FOR STRESS-EXPLICIT FLUIDS

F. T. Pinho

Centro de Estudos de Fenómenos de Transporte, DEMEGI, Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal, fpinho@fe.up.pt

Keywords: Computational rheology, finite-volumes, staggered grids, SIMPLEC, yield stress fluids

Abstract

In the first paper of this series (Pinho and Oliveira, *e-rheo.pt* 1 (2001) 1) a brief history on the use of finite-volumes in the context of computational rheology was presented. In the same work, the relevant conservation equations and the constitutive equations, of stress-explicit and stress-implicit types, for purely viscous and viscoelastic fluids respectively, were presented.

This paper is the second in the series and in it, the basics of the finite-volume methodology is outlined. We quickly review the method starting with diffusion and convection of a general quantity ϕ, such as the thermal energy, then the specificities of solving the momentum equation and of coupling pressure and velocity fields are dealt with in the context of staggered meshes. The problems encountered in the discretization of the convective terms are also discussed and the specific issues related to the calculation of variable viscosity, and especially of yield stress, typical of some non-Newtonian fluids, are also addressed. To address the problem of unbounded viscosity in yield stress fluids, modifications of the yield stress law by the bi-viscosity, modified bi-viscosity and Papanastasiou models are suggested and the methods are compared.

1. Introduction

We need to start from the very beginning for the benefit of the newcomer but, since the fundamentals of the finite-volume method are well established and extensively explained in such reference books as Patankar [1], Versteeg and Malalasekera [2] and Ferziger and Perić [3] we do so quickly and give a more detailed presentation on issues which are less well explained or absent in those references and on topics specific to non-Newtonian fluids.

As our main concern is to present a series of papers that are pedagogically sound we present here the finite-volume method following the classical approach of using staggered grids and orthogonal coordinates. In one of the next papers the generalisation of the method to non-orthogonal coordinates and colocated grids will be presented.

This paper is organised as follows: the set of equations to be solved for a non-Newtonian stress-explicit fluid is written down, then the finite-volume methodology is explained in the classical way in order to obtain the solution of the conservation equation for a general quantity ϕ: first, the equation for pure diffusion with source terms, then the equation for convection-diffusion. Various discretization schemes will be discussed for convection. This will be followed by the solution of the momentum equation and the explanation of the strategy to ensure coupling

between the pressure and the velocity fields via staggered meshes. From Section 4 to the end the paper discusses issue s involving the solution of purely viscous non-Newtonian fluids and their specific difficulties, with emphasis on handling yield stress fluids. Some results of pipe flow calculations are presented.

2. Fundamental equations

The equations to be solved and the coordinate system were presented in the first paper (Pinho and Oliveira [4]), but here we briefly remember them for the sake of completeness when the model is the stress-explicit Generalised Newtonian Fluid (GNF). The coordinate system is the Cartesian system x_i and the equations are:

- conservation of mass for incompressible fluids (non-Newtonian fluids are liquids)

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{1}$$

where u_i represents the velocity field;

- conservation of linear momentum

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i + \frac{\partial \tau_{ij}}{\partial x_j} \tag{2}$$

where t is the time, ρ is the fluid density, p is the pressure field and g_i is the aceleration of gravity;

- and the differential constitutive equation, here an explicit expression for the ij component of the stress tensor, τ_{ij}

$$\tau_{ij} = \eta(\dot{\gamma})\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\eta(\dot{\gamma})\frac{\partial u_k}{\partial x_k}\delta_{ij} \tag{3}$$

The second term on the right-hand-side vanishes according to continuity. However, especially when dealing with viscoelastic fluids, there are numerical advantages in keeping this term. In this paper, and henceforth, it will be dropped.

The viscosity function $\eta(\dot{\gamma})$ can be given by any of the models presented in the previous paper. Alternatively, we can also use a viscoelastic model as a constitutive equation but, since we are only interested on explicit models on the stress τ_{ij}, the choice is limited. One such example is the Criminale- Eriksen-Filbey constitutive equation

$$\tau_{ij} = 2\eta(\dot{\gamma})D_{ij} + 4\left[\frac{1}{2}\Psi_1(\dot{\gamma}) + \Psi_2(\dot{\gamma})\right]D_{ik}D_{kj} - \Psi_1(\dot{\gamma})\frac{DD_{ij}}{Dt} \tag{4-a}$$

where D/Dt is the Jaumann derivative

$$\frac{DD_{ij}}{Dt} \equiv \frac{\partial D_{ij}}{\partial t} + u_k\frac{\partial D_{ij}}{\partial x_k} + \left(\Omega_{ik}D_{kj} - D_{ik}\Omega_{kj}\right) \tag{4-b}$$

Ω_{ij} the vorticity tensor is

$$\Omega_{ij} \equiv \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right) \tag{4-c}$$

and Ψ_1 and Ψ_2 represent the first and second normal stress difference coefficients, respectively. Other stress-explicit constitutive equations are of integral type but a discussion on these in postponed to the future. In the following we concentrate on dealing with GNF fluids (Eq. 3).

When trying to solve equations (1) and (2) for GNF fluids, in the context of a finite-volume method, there are two possibilities:

i) The explicit constitutive equation is substituted into the momentum equation (2), thus eliminating variable τ_{ij}. This results in an equation which is similar to that for Newtonian fluids with variable viscosity, as appears in turbulent Newtonian flows or in laminar Newtonian flows with temperature-dependent viscosity. Equation (2) becomes

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i + \frac{\partial}{\partial x_j}\left[\eta(\dot\gamma)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \tag{5}$$

and, as with Newtonian fluids, it is advantageous to rearrange the viscous stress terms as

$$\frac{\partial}{\partial x_j}\left[\eta(\dot\gamma)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] = \frac{\partial}{\partial x_j}\left(\eta(\dot\gamma)\frac{\partial u_i}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left(\eta(\dot\gamma)\frac{\partial u_j}{\partial x_i}\right) \tag{6}$$

so that we have finally

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} - \frac{\partial}{\partial x_j}\left(\eta(\dot\gamma)\frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_i} + \rho g_i + \frac{\partial}{\partial x_j}\left[\eta(\dot\gamma)\frac{\partial u_j}{\partial x_i}\right] \tag{7}$$

The momentum equation (7) is written here in the so-called strong formulation which is advantageous to finite-volumes because it automatically ensures global momentum conservation in the calculation methods and also with all the normal diffusion terms appearing on the left-hand-side and the cross diffusion terms on the right-hand-side (rhs). As usual in the specialised literature, those terms on the left-hand-side of a conservation equation will be dealt with implicitly whereas those appearing on the right-hand-side are handled explicitly. The actual meaning of this will become clear at a later stage.

ii) Alternatively, the constitutive equation (3) is kept separately, i.e, there will be an equation on the stress τ_{ij} to be solved in each case. In this case, and for reasons of stability that will become clear in a future paper on stress-implicit viscoelastic fluids of the differential type, it is advantageous to add and subtract a diffusion term and to rewrite the momentum equation (2) as

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} - \frac{\partial}{\partial x_j}\left(\eta(\dot\gamma)\frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_i} + \rho g_i + \frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\eta(\dot\gamma)\frac{\partial u_i}{\partial x_j}\right] \tag{8}$$

Again, those terms on the left-hand-side will be handled implicitly whereas those on the right-hand-side will be solved explicitly.

The advantage of using equation (8) is its generality, i.e, the momentum equation is now independent of the constitutive model adopted. However, there are added diffusion terms in the momentum equation for the sake of convergence stability and these need to be adequately handled, especially when doing transient calculations, to avoid spurious effects.

On the other hand, if our aim is to solve purely viscous fluids it is worth to stay with the first strategy which leads to the same code used for solving turbulent Newtonian flows or non-isothermal laminar flows with temperature-dependent properties, where the turbulent and laminar viscosities vary throughout the calculation domain. So, the adaptation of such codes to deal with Generalised Newtonian fluids is straighforward.

Often, we also need to solve the internal energy equation (ϕ would be the temperature) or an equation of conservation for a chemical species (less frequent in non-Newtonian calculations) so we adopt the general form of the diffusion-convection equation which is written, in the strong formulation, as (see Section 2.3 of Pinho and Oliveira [4])

$$\frac{\partial(\rho\phi)}{\partial t}+\frac{\partial}{\partial x_i}(\rho u_i\phi)-\frac{\partial}{\partial x_i}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right)=S \tag{9}$$

Henceforth, we will be dealing with steady flows so we drop the first term on the left-hand-side. Due account of unsteady flows will be addressed in a future paper.

Equation (9) is very general and is not exclusively for scalar quantities: Equations (7) and (8) have the same form provided $\phi = u_j$ and S accounts for everything standing on the right-hand-side. So, conservation of mass, momentum, energy, and also the constitutive equation for viscoelastic fluids (as will be seen in a future paper), can all be written in the form of Equation (9) and it is with this equation that we describe the finite-volume method in the next section.

3. The Finite-Volume Methodology

In this method, the first step is the integration of a generic transport equation for quantity ϕ over a three-dimensional control volume V and there is here no approximation whatsoever

$$\int_V\frac{\partial}{\partial x_i}(\rho u_i\phi)dV-\int_V\frac{\partial}{\partial x_i}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right)dV=\int_V S_\phi dV \tag{10}$$

Next, we apply Gauss' divergence theorem to the left-hand-side of Equation (10), that is

$$\int_V\frac{\partial\phi_i}{\partial x_i}dV=\int_A\phi_i n_i dA \tag{11}$$

where n_i is the unit vector normal to the surface element dA. This theorem transforms volume integrals of divergence terms into surface integrals of fluxes all around the control volume, Gauss' theorem leads to the following integrated conservation equation

$$\int_A n_i(\rho u_i \phi)dA - \int_A n_i\left(\Gamma \frac{\partial \phi}{\partial x_i}\right)dA = \int_V S_\phi dV \qquad (12)$$

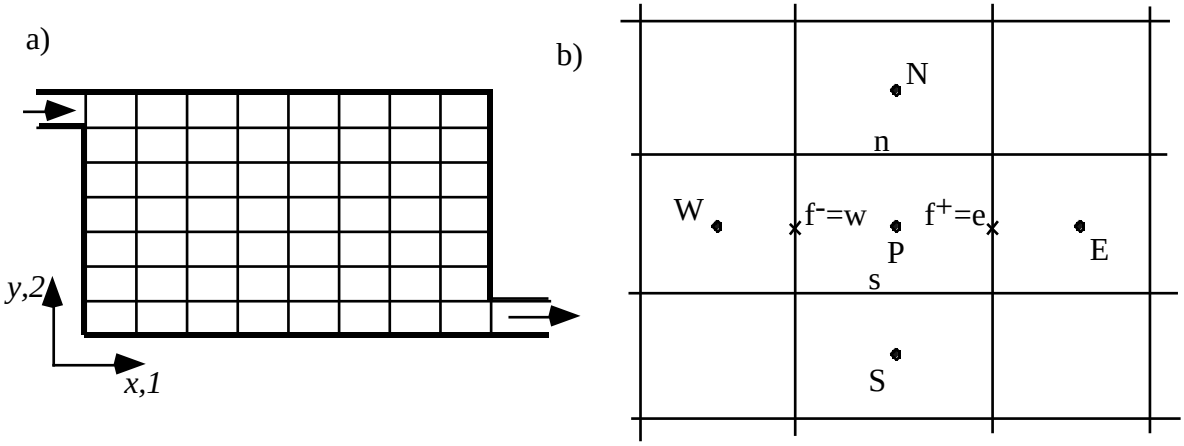


Figure 1- Two-dimensional control volumes in an orthogonal grid. a) General grid; b) A control volume and its neighbours.

In general, this equation cannot be solved analytically and a numerical solution will require that we need transform it into an algebraic equation. This transformation is called the discretization; it requires approximations to evaluete integrals and to do interpolations. For simplicity, we shall illustrate the procedure using the Cartesian two-dimensional grid represented in **Figure 1** instead of the actual three-dimensional grid. For the method explained here the mesh must be orthogonal, not necessarily Cartesian, and the treatment of the generalised coordinates is left to a future paper.

Figure 1-a) shows the grid mapping of a specific calculation domain in the x-y plane representing, for instance, flow in and out of a box. Figure 1-b) isolates a computational molecule made of a control volume surrounded by its nearest neighbours, and presents the corresponding notation. The central control volume is represented by its node *P*, where the fluid and flow quantities are stored and its neighbour nodes are denoted by *W*, *E*, *S*, *N* for west, east, south and north cells, respectively. In the z-direction, not shown here, there are the *B* (bottom) and *T* (top) cells. Nodes are always represented by capital letters and the faces of the control volume *P* adjacent to a given neighbour cell *F*= *W*, *E*, *S*, *N* is identified by the corresponding small letter *f*= *w*, *e*, *s*, *n*.

In discretizing Equation (12) the following approximations are applied:

i) The integral of g is approximated in terms of the values of g at several locations on the cell face. The simplest approximation adopted here is the midpoint rule, which is of second order, whereby $\overline{g}$ is the value at the center of the cell face (represented by crosses in Figure 1-b), i.e.

$$\int_{A_f} g dA = \bar{g}_f A_f \tag{13}$$

In this integral $\bar{g}_f$ may represent a convection flux $\left(n_i \rho u_i \phi\right)_f$ or a diffusion flux $\left(n_i \Gamma \partial\phi / \partial x_i\right)_f$. It is noted that other higher-order approximations could be applied but they lead to an integration rule requiring more points on the cell face.

ii) the cell face values of the function g_f are approximated in terms of the nodal values g_F. Also here, a second order accurate scheme will be used to calculate the cell face values of any function g from their values at central nodes.

3.1- Steady-state diffusion

We start the discretization of Equation (12) by looking at one-dimensional diffusion (along direction *x*) for which the equation to be solved simplifies to

$$-\int_V \frac{d}{dx}\left(\Gamma \frac{d\phi}{dx}\right) dV = \int_V S_\phi dV \quad \rightarrow \quad -\int_A n_x \left(\Gamma \frac{d\phi}{dx}\right) dA = \int_V S_\phi dV \tag{14}$$

With the above two approximations, the left- and right-handd-side terms become

$$\int_A n_x \left(\Gamma \frac{d\phi}{dx}\right) dA = \left(\Gamma A \frac{d\phi}{dx}\right)_e - \left(\Gamma A \frac{d\phi}{dx}\right)_w \tag{15-a}$$

$$\int_V S_\phi dV = \bar{S} \Delta V \tag{15-b}$$

with the overbar denoting here the average value over the control volume *P*.

Equation (15-a) indicates that the gradients of ϕ are to be evaluated at the center of faces *e* and *w* and this requires further assumptions, this time for the variation of fluid and flow properties between nodal points. For the flow properties the simplest assumption is a linear variation which is called central differencing, but this is not always an adequate approximation for fluid properties as will be shown in Section 3.3. For an uniform grid the central difference approximation leads to

$$\left(\Gamma A \frac{d\phi}{dx}\right)_e = \Gamma_e A_e \left(\frac{\phi_E - \phi_P}{\delta x_{EP}}\right) \tag{16}$$

and similarly for diffusion fluxes along the other directions. In Equation (16) δx_{EP} represents the distance between nodes *E* and *P* along *x*. The source term $\bar{S}$ may be a function of ϕ and in this case it should be linearised as (Patankar, 1980):

$$\bar{S} \Delta V = S_u + S_P \phi_P \tag{17}$$

Grouping all terms, we arrive at the following discretized equation for one-dimensional steady-state diffusion

$$-\left[\Gamma_e A_e \left(\frac{\phi_E - \phi_P}{\delta x_{EP}}\right) - \Gamma_w A_w \left(\frac{\phi_P - \phi_W}{\delta x_{PW}}\right)\right] = S_u + S_P \phi_P \tag{18}$$

which can be rearranged into

$$\left(\frac{\Gamma_e A_e}{\delta x_{PE}} + \frac{\Gamma_w A_w}{\delta x_{WP}} - S_P\right)\phi_P = \left(\frac{\Gamma_w A_w}{\delta x_{WP}}\right)\phi_W + \left(\frac{\Gamma_e A_e}{\delta x_{PE}}\right)\phi_E + S_u \tag{19}$$

Equation (19) is often written in the canonical form

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u \tag{20-a}$$

with coefficients

$$a_W = \frac{\Gamma_w A_w}{\delta x_{WP}}; \quad a_E = \frac{\Gamma_e A_e}{\delta x_{PE}}; \quad a_P = a_E + a_W - S_P \tag{20-b}$$

We can generalise Equation (20-a) to a three-dimensional problem

$$a_P\phi_P = \sum_{F=1}^{6} a_F\phi_F + S_u \tag{21-a}$$

with coefficients

$$a_F = \frac{\Gamma_f A_f}{\delta x_{FP}}; \; a_P = \sum_{F=1}^{6} a_F - S_P \tag{21-b}$$

where F runs over the 6 near-neighbour nodes W, E, S, N, B and T and the small letters represent the corresponding cell-faces (adjacent to cells *F* and *P*). ϕ_P and ϕ_F represent the unknown values of ϕ that need to be numerically determined.

As we can see, the diffusion term on the left-hand-side of the conservation Equation (14) gave rise to the terms with unknowns ϕ_P and ϕ_F, i.e, terms that are dealt with implicitly. On the other hand, if S_u depends on ϕ_P and ϕ_F in any way, for the solution of Equation (20) or (21) S_u is assumed known, either because the field ϕ was initialised to allow the determination of S_u or, in an iterative process, the field ϕ used to calculate S_u is the solution from the previous iteration.

3.2- Boundary-conditions

Solution of Equation (21) requires boundary conditions, as in the original differential equations, which are usually set as values of ϕ, or its flux at the boundaries. If a particular boundary condition provides the value of ϕ, then we know ϕ at specific cell faces located at the boundary ϕ_f. Then, the flux of Equation (16) for the cell *P* closest to the border (which is the west boundary in the example in Figure 2) uses the cell face value ϕ_f and is given by

$$\left(\Gamma A\frac{d\phi}{dx}\right)_e = \Gamma_e A_e\left(\frac{\phi_e - \phi_P}{\delta x_{eP}}\right) \tag{22}$$

i.e., ϕ_E is no longer an unknown for this flux.

If the boundary condition provides $\dot{Q}$ instead, Equation (16) is substituted by Equation (22) with ϕ_e given by

$$\phi_e = \phi_P + \frac{\dot{Q}\delta_{xeP}}{\Gamma_e A_e} \tag{23}$$

So, in any case, a value of ϕ east of cell *P*, either ϕ_E or ϕ_P, is no longer an unknown. The two typical boundary conditions are illustrated in Figure 2 for a cell *P* with a boundary in its west face.

The boundary conditions have an immediate consequence on writing Equation (21) at a boundary. The discretised conservation equation at the boundary control volume *P* still obeys the same general form but, since ϕ is now known at the boundary, the corresponding coefficient $a_F = a_{FB}$ (subscript *FB* stands for neighbour cell at the boundary domain) is set to zero with the corresponding algebraic modification carried over to the source term, i.e., the source term at a boundary cell is now calculated as $S_{u,B} = S_u + a_{FB}\phi_{FB}$.

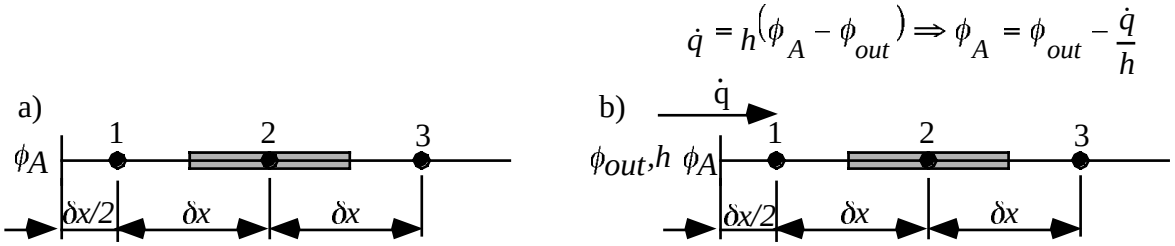


Figure 2- Examples of boundary conditions: a) Value of ϕ at boundary; b) Flux $\dot{q}$ of ϕ at boundary.

In Figure 2-a) ϕ is known at the west face of the one-dimensional domain of constant cross-section area *A*. The equation for the control volume centered in node *P*= 1 is

$$\left(\frac{\Gamma A}{\delta x} + \frac{\Gamma A}{\delta x/2}\right)\phi_P = \left(\frac{\Gamma A}{\delta x/2}\right)\phi_A + \left(\frac{\Gamma A}{\delta x}\right)\phi_E \tag{24}$$

which is written in canonical form with coefficients

$$a_E = \frac{\Gamma A}{\delta x};\ a_W = 0;\ S_P = -\frac{2\Gamma A}{\delta x} \text{ and } S_u = -\frac{2\Gamma A}{\delta x}\phi_A \tag{25}$$

Handling the flux boundary condition of Figure 2-b) is similar.

3.3- Fluid properties at interfaces

Whereas the variation of flow properties across cell faces is adequately calculated using linear interpolation and providing secon-order accuracy, the calculation of fluid properties may have to follow a different strategy whenever there are large variations in properties across the cell face. A linear variation is physically incorrect and fails miserably as is well illustrated by the problem of heat conduction across an interface of materials of very different thermal conductivities. The sudden change in this property has similarities, in the context of non-Newtonian fluids, to large changes in viscosity for strongly shear-thinning fluids and especially yield stress fluids: in the latter case, the issue of determining the correct viscosity at the interface between the yielded and the plug regions of yield stress fluid flows becomes especially important and it is fundamental to use the strategy explained below for accurate predictions.

The solution to this problem is also described in the classical literature on finite-volumes and is easy to understand by analogy with the one-dimensional heat conduction across interfaces of solid materials. The objective is to obtain the correct flux of ϕ, say the heat flux, so the rule to calculate the fluid property Γ_f at interface f is the same as that for obtaining the equivalent thermal conductivity of two consecutive slabs of different thermal conductivities, i.e.

$$\Gamma_f = \left(\frac{1 - L_f}{\Gamma_P} + \frac{L_f}{\Gamma_F} \right)^{-1} \tag{26}$$

where L_f is an interpolation factor defined by

$$L_f \equiv \frac{\delta x_{fF}}{\delta x_{PF}} \tag{27}$$

3.4- Steady-state convection and diffusion

Now, we extend the previous analysis of Equation (12) by including the convection term but continue to use a one-dimensional equation for simplicity. The 1-D conservation equation for ϕ becomes

$$\frac{\partial}{\partial x}(\rho u \phi) - \frac{\partial}{\partial x}\left(\Gamma \frac{\partial \phi}{\partial x} \right) = S \tag{28}$$

The unknown is still ϕ, and the flow field u satisfies the continuity and momentum equations. On Equation (28) the finite-volume method is applied: the equation is volume-integrated, Gauss' theorem is applied and a discretization is performed leading to

$$(\rho u A \phi)_e - (\rho u A \phi)_w - \left[\left(\Gamma A \frac{d\phi}{dx} \right)_e - \left(\Gamma A \frac{d\phi}{dx} \right)_w \right] = \bar{S} \Delta V \tag{29}$$

The diffusion and source terms are handled as in Section 3.1. It is useful at this stage to introduce the concept of diffusion conductance $D \equiv \Gamma / \delta x$. Regarding the convection terms, the convective flux is defined as $F \equiv \rho u$ and now we rewrite Equation (29) as

$$F_e A_e \phi_e - F_w A_w \phi_w - \left[D_e A_e (\phi_E - \phi_P) - D_w A_w (\phi_P - \phi_W) \right] = S_u + S_P \phi_P \tag{30}$$

As we can see, whereas diffusion directly gave rise to values of ϕ at the nodes (ϕ_P and ϕ_F), convection gave values of ϕ at faces (ϕ_f). These are unknowns that need to be determined as a function of nodal values via an interpolation function, which os adequate for convection. The obvious choice is the linear interpolation of the neighbour nodal values, already used for diffusion, which is called the central differencing scheme (CDS). It is a second order scheme and is able to provide accurate solutions, but has the disadvantage of being unstable when convective fluxes are supercritical, i.e., when $F_f / D_f \geq 2$. It brings us to the issues of stability, accuracy and other properties of discretization schemes. Although this is a very important matter, we do not spend much time on it as there are no specific problems related to GNF or viscoelastic stress-explicit constitutive equations. Therefore, the reader interested in details should consult the works of Patankar [1], Versteeg and Malalasekera [2] and Ferziger and Perić [3].

3.4.1. Properties of discretisation shemes

Five properties are discussed here: consistency, stability, convergence, boundedness and transportiveness.

Consistency requires the truncation error of the discretization method to become zero as the mesh is refined. Consistency also requires that, in the balance of any quantity over a CV, the flux calculated at a cell face to be the same as the flux at the same face viewed from the adjacent control volume. All schemes used here and in the future papers in this series, and those we recommend, are formulated to obey consistency.

Stability requires the method not to amplify any disturbance that appears in the course of the calculation. If stable the method does not diverge and, for time-dependent problems, the solution of the algebraic set of equations is bounded when the differential equation is bounded.

The numerical method is convergent when the solution of the set of algebraic equations tends to the solution of the original differential equations with mesh refinement. As this is a difficult property to demonstrate, the user is usually requested to validate the numerical method against various known and relevant experimental, numerical or analytical solutions. With consistent, stable and convergent schemes, mesh refinement provides grid-independent results in the course of the validation procedure.

As mentioned in Section 2, one of the advantages of finite-volumes is that by writing the conservation equations in a strong formulation, as they have been throughout the paper, conservation of the quantities is guaranteed for each individual cell and the overall calculation domain. This advantage can not be overstated and its strength has fostered the appearance of hybrid finite-volume/ finite-element methods, as in Wapperom and Webster [5].

The numerical solution of the algebraic equations should be bounded within some limits which usually have a physical origin. For instance, in the absence of sources or sinks of thermal energy then its value should be bounded by the values imposed by the boundary conditions. Boundedness is guaranteed by some first-order discretisation and interpolation methods and by appropriately formulating and bounding higher-order methods. Lack of boundedness is also usually associated with stability and convergence problems and consequently, unbounded methods should be avoided. A sufficient condition for a convergent iterative method is to have

$$\frac{\sum_F |a_F|}{|a_P|} \begin{cases} \leq 1 \text{ at all nodes} \\ < 1 \text{ at one node at least} \end{cases} \tag{31}$$

Eq. (31) results in matrices which are diagonally dominant and we must ensure that all coefficients a_P, a_F of the discretised equations have the same sign, usually positive. The reader should note that the definition embodied in Equation (21) guarantees positive coefficients for a_F and for a_P provided $S_P < 0$, which is the case.

The discretization of the differential equations should be accurate using methods that are at least of second order accuracy, ie., the progression of the error decrease with mesh refinement should at least be quadratic to reduce the need for highly refined meshes which become excessively expensive in computational resources.

Due to the presence of convection, a given quantity ϕ that is locally produced will not spread equally in all directions, a situation that would occur in pure diffusion. The measure of the relative strengths of diffusion and convection is the Peclet number

$$Pe \equiv \frac{F}{D} \tag{32}$$

In pure diffusion the Peclet number is equal to zero and ϕ will spread equally in all directions, whereas for a Peclet number of infinity ϕ will be transported exclusively in the direction of convection, away from its source. It is important for any interpolation scheme for convection to be able to reproduce, as faithfully as possible, the ratio of these two strengths.

3.4.2. Discretization schemes for convection

For simplicity uniform meshes will be considered in the following description of discretization schemes. In Equation (30) the cell face values ϕ_f must be calculated from nodal values so that the resulting equation is written in the canonical form of Equation (21).

3.4.2.1- Central Differencing Scheme (CDS)

Using linear interpolation, also called CDS, to determine the cell face values ϕ_f from the adjacent nodal values ϕ_P and ϕ_F, Equation (30) is written in the canonical form of Equation (21) with coefficients *a* assuming the form

$$a_w = D_w + \frac{F_w}{2};\ a_e = D_e + \frac{F_e}{2};\ a_P = a_W + a_E + \left(F_e - F_w\right) \tag{33}$$

In one-dimensional flow continuity ensures the last term of a_P to be zero, so a_P is just the sum of the neighbour coefficients. These neighbour coefficients can only be positive if $Pe < 2$, otherwise CDS becomes unstable. When convection is too strong in comparison to diffusion ($Pe > 2$), CDS violates boundedness. It can also be seen that CDS does not possess transportiveness as it brings into ϕ_f a diffusive behaviour, i.e. CDS does not obey the transportiveness property but it has the advantage of being second order accurate.

3.4.2.2- Upwind Differencing Scheme (UDS)

An alternative method, which is unconditionally stable and obeys transportiveness, is the upwind scheme (UDS). It basically states that if the flow is from West to East, then $\phi_w = \phi_W$ and $\phi_e = \phi_P$, i.e, the cell face value ϕ_f takes the value of ϕ at the node immediately upstream. With UDS adopted for convection, the coefficients *a* of the canonical equation (21-a) take the form

$$a_w = D_w + \max(F_w, 0);\ a_e = D_e + \max(0, -F_e);\ a_P = a_W + a_E + (F_e - F_w) \quad (34)$$

The coefficients a_P and a_F are now always positive and this feature makes the method unconditionally stable. However, the corresponding Taylor series expansion and truncation shows us that the method is only first-order accurate. UDS also suffers from what is called false (or numerical) diffusion which is the smearing of transported properties, as happens in diffusion dominated processes, when the direction of the flow is oblique to the mesh used.

3.4.2.3- The Hybrid Differencing Scheme (HDS)

The hybrid differencing scheme (HDS) is a combination of UDS and CDS and was derived by Spalding [6]. The idea was to combine the advantages of UDS and CDS, but since its accuracy remains of first order, the scheme lost popularity in the late eighties when issues of accuracy became very important. With HDS the *a* coefficients become

$$a_w = \max\left(F_w, D_w + \frac{F_w}{2}, 0\right);\ a_e = \max\left(-F_e, D_e - \frac{F_e}{2}, 0\right);\ a_P = a_W + a_E + (F_e - F_w) \quad (35)$$

Equation (35) shows that HDS switches from CDS to UDS whenever the absolute value of the Peclet number exceeds 2. If convection is strong, HDS becomes the upwind scheme and this explains the accuracy of the method which tends to that of the poorest scheme used unless the mesh is refined to reduce the Peclet number.

An alternative to HDS is the power law scheme. It is based on the exact solution of the one-dimensional convection-diffusion problem (Patankar [1]), but again the method is first-order accurate and is very seldom used.

3.4.2.4- Linear Upwind Differencing Scheme (LUDS)

This scheme has the advantage of second order accuracy, obeys transportiveness by bringing some upwinding, but is not unconditionally stable. In one-dimensional flow the cell face value ϕ_f is evaluated by linear extrapolation from the two closest upstream nodal values along the same coordinate (see Figure 3). So, for ϕ_e we have

$$\phi_e = \begin{cases} \phi_P + (\phi_P - \phi_W)(1 - L_w), & \text{if} \quad F_e > 0 \\ \phi_E + (\phi_E - \phi_{EE})(1 - L_{ee}), & \text{if} \quad F_e < 0 \end{cases} \quad (36)$$

To build the coefficients *a* for LUDS the notation must be further extended according to Figure 1. When referring to a given cell *P*, and for any given direction, say direction *x* or 1, cell faces f^- and f^+ refer to the faces on the negative (west, south, bottom) and positive (east, north, top) sides of *P*, respectively. Then, for a given cell face (f^- or f^+) there are two different interpolation factors L_f^- and L_f^+ which are defined as

$$L_f^+ = \frac{[\Delta l]_{fP}}{[\Delta l]_{fP} + [\Delta l]_{fF}} \text{ and } L_f^- = 1 - L_f^+ \quad (37)$$

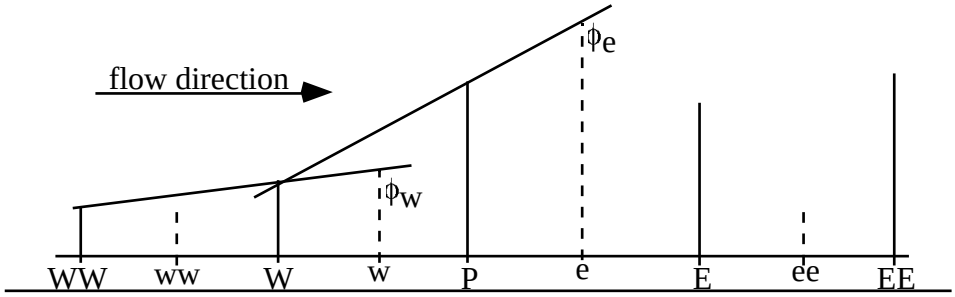


Figure 3- Representation of the linear upwind scheme.

The novelty of LUDS is that it brings the need to account not only for the near-neighbours but also for the far neighbours. *EE* represents the node to the east of node *E* and *ee* refers to the cell face adjacent to *E* and *EE* (see Figure 3). Consequently, for the one-dimensional problem there are now four a_F coefficients plus the central coefficient a_P:

$$a_W = D_w + \max(F_w, 0)\left(2 - L_{ww}^{+}\right) + \max(0, F_e)\left(1 - L_w^{+}\right)$$

$$a_E = D_e + \max(0, -F_e)\left(1 + L_{ee}^{+}\right) + \max(0, -F_w)L_e^{+}$$

$$a_{WW} = -\max(F_w, 0)\left(1 - L_{ww}^{+}\right)$$

$$a_{EE} = -\max(0, -F_e)L_{ee}^{+}$$

$$a_P = a_W + a_E + a_{EE} + a_{WW} \qquad (38\ \text{a,b,c,d,e})$$

However, for reasons of stability the far-neighbour nodal terms $a_{FF}\phi_{FF}$ are not considered as unknowns, i.e., they are handled explicitly y being calculated on the basis of the solution of the previous iteration. Since the far-neighbour coefficients can be negative, the total convective term is not handled as in the case of UDS but is split as in Equation (39)

$$F_f\phi_f = \sum_F a_F(\phi_P - \phi_F) + \sum_{FF} a_{FF}(\phi_P - \phi_{FF}) \qquad (39)$$

Now, the first term on the right-hand-side is treated implicitly, i.e, ϕ_P and ϕ_F will be unknowns in each iteration and so are kept on the left-hand-side of Equation (30) whereas the second term on the right-hand-side is treated explicitly, i.e., it goes to the right-hand-side of the conservation Equation (30) to be included as part of the source term (contribution $S_{u_{FF}}$ and $S_{p_{FF}}$). In doing so the term is linearised, i.e., the term will be written as

$$S_u + S_P\phi_P = \sum_{FF} a_{FF}(\phi_{FF} - \phi_P) = \sum_{FF} a_{FF}\phi_{FF} - \sum_{FF} a_{FF}\phi_P \qquad (40)$$

Then, the linearised contribution term $S_P\phi_P$ goes to the left-hand-side of the conservation Equation (30) and this explains why a_P includes the far-neighbour contributions a_{FF}, but $a_{FF}\phi_{FF}$ are not considered unknowns and were thus moved to the right-hand-side of Equation (30).

3.4.2.5- The Deferred Correction

As explained above, due to stability issues the implementation of LUDS differs from the rules set out in previous sections. Similarly, the use of CDS was seen to lead also to stability problems when the Peclet number exceeded 2. Other discretization schemes, such as the second order accurate quadratic upwind (QUICK), obey transportiveness and suffer from stability issues unless the convective terms are split into implicit and explicit terms as was done for LUDS.

Clearly, a more systematic approach is required to address this issue and one such approach is called the deferred correction. For this purpose it is important to recall that the first order UDS scheme has the enormous advantage of being unconditionally stable. The deferred correction is a general method that combines the unconditional stability of UDS with the higher accuracy of conditionally stable schemes, at a price of slower convergence. The starting point of the deferred correction is the algebraic Equation (30) which is here repeated.

$$F_e A_e \phi_e - F_w A_w \phi_w - \left[D_e A_e \left(\phi_E - \phi_P \right) - D_w A_w \left(\phi_P - \phi_W \right) \right] = S_u + S_P \phi_P$$

One aims at determining face values ϕ_f of the convective terms on the left hand side as a function of nodal values ϕ_F, ϕ_P using some interpolating scheme, which we take in this example to be CDS. First, we add and subtract similar convective terms to the right hand side of the equation and rearrange it so that it becomes

$$\begin{aligned} \left(F_e A_e \phi_e - F_w A_w \phi_w \right)_{UDS} - \left[D_e A_e \left(\phi_E - \phi_P \right) - D_w A_w \left(\phi_P - \phi_W \right) \right] = S_u + S_P \phi_P + \\ \left(F_e A_e \phi_e - F_w A_w \phi_w \right)_{UDS} - \left(F_e A_e \phi_e - F_w A_w \phi_w \right)_{CDS} \end{aligned} \quad (41)$$

As we can see in Equation (41), the convective term on the left hand side is processed using UDS and the term is treated implicitly to benefit from the unconditional stability of the upwind scheme. On the right hand side of Equation (41) the added and subtracted convective terms are processed with UDS, and the second order accurate interpolating scheme CDS. Both terms on the right-hand-side become part of the source term and are treated explicitly, i.e., they are calculated using nodal values from the previous iteration.

As the solution is approached the values of field ϕ from consecutive iterations tend to be equal, so that both UDS terms cancel and only the convective term interpolated with CDS or any other high accuracy scheme remains.

3.4.2.6- Other High-Order Schemes

Other interpolation schemes for convection are the quadratic upwind scheme (QUICK) of Leonard [7], which uses two nodal values upstream of the cell face and one downstream, or schemes that combine in a special way the expressions from QUICK, CDS, UDS which are called high-resolution schemes (Harten [8]). High-resolution schemes combine advantages of various schemes to ensure accuracy, stability and boundedness. The application of high-resolution schemes to the prediction of viscoelastic fluid flows in finite volumes is quite recent and examples are the works of Alves et al [9,10].

Interpolation schemes of third and higher order can also be used, but deeper changes must be introduced to the various approximations in the whole method if one is to benefit clearly in the results. The approximations used in the discretization of the integrated differential equation in Section 3 are of second order and they should be upgraded to be combined with higher order interpolation schemes, so that the results reflect the improved accuracy. Increasing only the order of accuracy of the interpolating schemes for convection above 2, without improvements in the other approximations, wil have a minimum impact on the final results but an increase in computational cost.

3.5- Solution of the Momentum Equation in Steady Flows

So far, a conservation equation for a general quantity ϕ has been addressed and the flow field has been assumed known. However, in a real CFD problem the flow field is often unknown and the momentum Equation (2) must also be solved, i.e., ϕ can also be u_i. Besides, the momentum equation for u_i also depends of the other components of the velocity vector $u_i (i \neq j)$, i.e., the momentum equations are non-linear whereas previously the equation for ϕ was linear. The non-linearity issue is addressed by solving the momentum equations sequentially and by assuming that, for each component of the velocity the other components are known.

A second problem of solving the flow field results from an unknown pressure field and the absence of a pressure equation. If the pressure gradient is known, solution of the momentum equation provides the velocity field in the same way as the conservation equation for ϕ. However, in general the pressure field is unknown.

In compressible flows the continuity equation can be solved to determine the fluid density, then the thermodynamic constitutive equation determines the pressure and so the solution of the momentum equation follows the above guidelines for solving ϕ.

If the flow is incompressible the fluid density is constant and is not related to pressure. Although there are four unknowns for four equations (the three momentum equations plus the continuity equation) there is no explicit equation for the pressure. However, continuity imposes a constraint because only a correct pressure field will result in a velocity field that obeys both the momentum and continuity equations.

The solution to this problem is an iterative algorithm where the continuity equation is transformed into an equation for the pressure field. Here, we will present the original SIMPLE strategy of Patankar and Spalding [11] and two of its improvements, SIMPLER and SIMPLEC.

There is yet another problem with the momentum equation. Let us consider again a one-dimensional flow in the *x*-direction of Figure 1-b). Integrating Equation (2), following the finite-volume guidelines, with a uniform grid for simplicity, and considering linear interpolation for the pressure, the pressure gradient term becomes

$$\frac{\partial p}{\partial x} = \frac{p_e - p_w}{\delta x_{ew}} = \frac{\frac{p_E + p_P}{2} - \frac{p_P + p_W}{2}}{\delta x_{ew}} = \frac{p_E - p_W}{2\delta x_{ew}} \tag{42}$$

According to Equation (42) the pressure gradient centered on node *P* is independent of p_P and is calculated with a coarser grid. The same result is obtained in the other two directions, so that, for instance, an unphysical situation of a checkerboard field pattern of pressure can result in the same velocity field as a well-behaved zero pressure gradient field. This lack of coupling between the velocity and pressure fields requires a special remedy which was devised by Harlow and Welch [12]: the use of grids for the velocity components which are staggered in relation to the original pressure (or scalar) grid. In 1983 Rhie and Chow [13] formulated a new pressure-velocity coupling strategy that avoided the need for staggered grids and opened the way for non-orthogonal collocated grids which are far more advantageous, but that will be the issue of a later chapter. Next, the staggered grid approach is adopted and explained.

3.5.1- Pressure-Velocity Coupling in Staggered Grids

To ensure coupling between the pressure and the *x*-direction velocity field (*u*) a second grid, which is staggered in the *x*-direction relative to the original grid, is used for the velocity component *u*, with the pressure being calculated in the original grid. For a three-dimensional flow three staggered grids are required, one for each velocity component, and the original grid is used for the pressure and other scalars. Figure 4-a) shows the original grid for pressure and Figure 4-b) shows the staggered grid for the *x*-velocity component and the original grid for pressure. In Figure 4-b) the full lines represent the faces of the control volumes of the original grid and the filled circles the location of their central nodes. Similarly, the dashed lines represent the cell faces of the staggered grid and the corresponding central nodes are marked with open circles. The capital symbols designate the coordinates referred to the original grid whereas the small symbols do the same for the staggered grid. It is clear from the figure that the faces of the staggered grid coincide with the nodes of the original grid and vice-versa.

In this way, for the *u*-velocity control volume centered on (*i*,*J*), the pressure gradient is now calculated as

$$\frac{\partial p}{\partial x} = \frac{p_P - p_W}{\delta x_{WP}} \tag{43}$$

which only involves consecutive pressure nodes and the checkerboard pattern is no longer a possible solution to a physical problem.

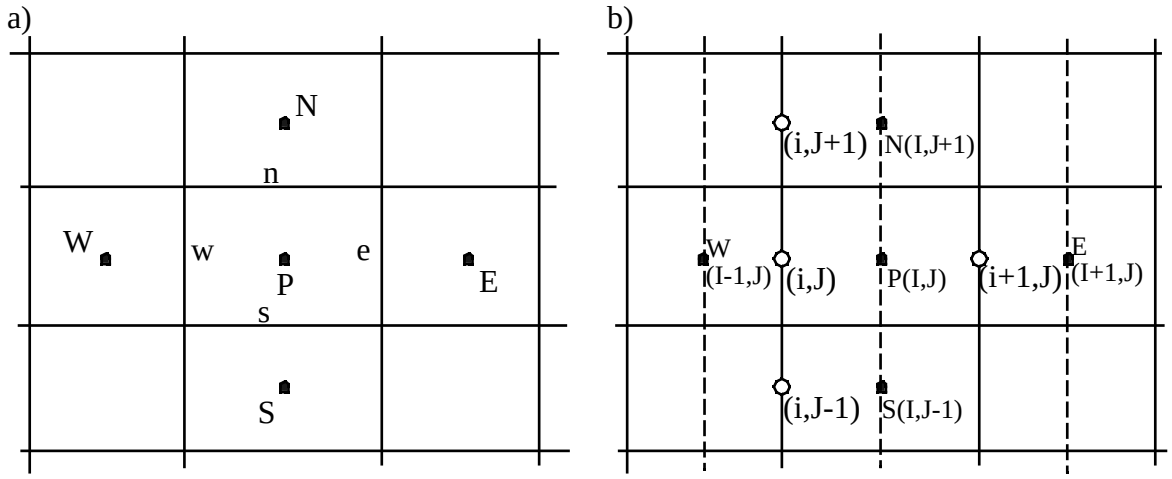


Figure 4- a) The original grid for pressure and other scalars; b) The original grid and the staggered grid for *u*-velocity.

A side benefit of the staggered grid is the treatment of convective terms of the ϕ conservation equation, when ϕ is not the velocity. The original grid stores the pressure as well as other quantities ϕ but the velocity field, so mass fluxes at the faces of the original control volumes are known because they coincide with the nodes of the staggered grids where velocities are stored, thus eliminating the need for interpolations for those velocity components.

Once the staggered grids are established and assuming previous knowledge of the pressure field (say, the result of the previous iteration), the momentum equation is solved in the same way as the equation for the scalar ϕ except that the values of the velocity are determined at the nodes of the staggered grid and those of ϕ at the nodes of the original grid. So, in a 3-D flow the three velocity components and the remaining quantities are all calculated at different locations; if, at the post-processing stage, data are required at the same location it will be just a matter of interpolation.

Since the solution of the momentum equation assumes previous knowledge of the pressure field it is necessary to devise a method to obtain the pressure field from the continuity equation.

3.5.2- Solution Algorithms: SIMPLE

The continuity Equation (1) is discretized into

$$\sum_{f=1}^{6} (-1)^f F_f A_f = 0 \qquad (44)$$

which expresses the fact that the sum of incoming mass flow rates equals the sum of outgoing mass flow rates. It is important to realise that, since the continuity equation will be used to determine the pressure field, it is derived in the original grid for pressure. Hence, the flow rates at the faces of the original CV's readily use the staggered grid velocities without any interpolation.

In order to turn the algebraic Equation (44) into an equation for pressure, the algorithm SIMPLE (Semi- Implicit Method for Pressure-Linked Equations) or one of its extensions/ improvements is adopted.

In explaining SIMPLE, a 2-D steady flow problem is considered but for conciseness only the equation for the *x*-velocity component will be used. In SIMPLE, a pressure field p^* is initially assumed known (from the previous iteration or initial conditions) and the discretised momentum Equation (45) is solved for u^* at the central node (i,J) (see Figure 4-a)

$$a_{i,J}u^*_{i,J} = \sum a_F u^*_F + \left(p^*_{I-1,J} - p^*_{I,J}\right)A_{i,J} + S_{i,J} \tag{45}$$

where the stars indicate that the quantities are not correct in the current iteration, because they are based on an initially guessed pressure field, and *S* is related to the source term of *u*. The correct pressure and velocity are given by

$$p \equiv p^* + p' \tag{46}$$

$$u \equiv u^* + u' \tag{47}$$

where the prime indicates a correction. If the pressure field *p* used in the momentum Equation (45) had been right, the result would have been the correct velocity field *u*, i.e, the momentum equation would then be

$$a_{i,J}u_{i,J} = \sum a_F u_F + \left(p_{I-1,J} - p_{I,J}\right)A_{i,J} + S_{i,J} \tag{48}$$

Subtracting Equation (45) from Equation (48), and using the definitions in Equations (46) and (47), the following relationship between velocity and pressure corrections is derived

$$a_{i,J}u'_{i,J} = \sum a_F u'_F + \left(p'_{I-1,J} - p'_{I,J}\right)A_{i,J} \tag{49}$$

To avoid the need to solve another system of equations, Equation (49) is modified by dropping $\sum a_F u'_F$. In this way, an explicit equation for the velocity correction is obtained

$$a_{i,J}u'_{i,J} = \left(p'_{I-1,J} - p'_{I,J}\right)A_{i,J} \tag{50}$$

from which the correct velocities are determined as

$$u_{i,J} = u^*_{i,J} + \frac{\left(p'_{I-1,J} - p'_{I,J}\right)A_{i,J}}{a_{i,J}} \tag{51}$$

The approximation from Equation (49) to Equation (50) does not preclude an exact solution because, in both equations, the velocity and pressure corrections u' and p' tend to zero as the correct velocity and pressure fields are approached, i.e, the neglected $\sum a_F u'_F$ also tends to zero as the solution is approached and no conservation principle is being violated.

Expanding Equation (44) in the original grid and substituting the flow rates by their definitions gives Equation (52)

$$\left[(\rho uA)_{i+1,J} - (\rho uA)_{i,J}\right] + \left[(\rho vA)_{I,j+1} - (\rho vA)_{I,j}\right] = 0 \tag{52}$$

Now, the expression for the correct velocity $u_{i,J}$ (Equation 51), and similar expressions for the other velocities, are substituted in Equation (52) yielding an equation on the pressure corrections.

$$a_{I,J}^{p} p'_{I,J} = \sum_{F} a_{F}^{P} p'_{F} + b_{P} \tag{53}$$

written here in the canonical form. In Equation (53) the superscript p indicates that the coefficient a refers to the pressure correction equation and differs from other definitions of a coefficients. Since the pressure, and obviously the pressure correction, is calculated in the original grid (at (I,J) and its neighbours), subscripts P and F appear again to designate the central node and its near neighbours, respectively. The coefficients of the pressure correction equation are

$$a_{I+1,J}^{p} = \left(\frac{\rho A^2}{a}\right)_{i+1,J} ; a_{I-1,J}^{p} = \left(\frac{\rho A^2}{a}\right)_{i,J} ; a_{I,J-1}^{p} = \left(\frac{\rho A^2}{a}\right)_{I,j+1} ; a_{I,J+1}^{p} = \left(\frac{\rho A^2}{a}\right)_{I,j}$$
$$a_{I,J}^{p} = \sum_{F} a_{F}^{P} \; ; \; b_{P} = \left(\rho u^{*} A\right)_{i,J} - \left(\rho u^{*} A\right)_{i+1,J} + \left(\rho v^{*} A\right)_{I,j} - \left(\rho v^{*} A\right)_{I,j+1} \tag{54}$$

In Equation (53) b_P represents the mass imbalance of the incorrect starred velocity field which tends to zero as the calculation proceeds towards the correct solution.

The pressure correction equation often diverges and, to prevent this, underrelaxation is used to determine improved pressures as

$$p^{new} = p^{*} + \alpha_p p' \tag{55}$$

where the underrelaxation factor α_p takes a value of less than one. For the same reason the velocity field is also underrelaxed by factor α_u using Equation (56)

$$u_{new} = \alpha_u u + (1 - \alpha_u) u^{(n-1)} \tag{56}$$

where u represents the corrected velocity value (present iteration) prior to underrelaxation and $u^{(n-1)}$ is the correct velocity from the previous iteration ($u_{new}^{(n-1)}$).

To summarize, the SIMPLE algorithm proceeds as follows in each iteration:

i) There is an initial guess of pressure and velocity p^{*}, u_i^{*}, taken from the previous iteration ($p^{*} = p^{(n-1)}$, $u_i^{*} = u_i^{(n-1)}$);

ii) Equation (45) is solved to determine a new u_i^{*} at the present iteration;

iii) Equation (53) determines the pressure correction field p';

iv) Pressure is corrected using Equation (46) and velocity is corrected with Equation (51). Now, the corrected pressure and velocity fields conserve momentum and mass within a certain residual. Naturally, if underrelaxation is being used Equations (55) and (56) are now used adequately to determine p^{new}, u^{new};

v) Finally, it is time to solve the conservation equations for any other scalar quantities ϕ, such as temperature;

vi) If convergence has been attained within a pre-defined residual the iterative calculation stops, otherwise the newly formed quantities are made equal to p^*, u_i^* and we proceed again to step (i) to initiate a new iteration.

3.5.3- SIMPLER and SIMPLEC

In 1980 Patankar [1] revised his SIMPLE method to address its weakness in the pressure calculation. The result was a new algorithm called SIMPLER (SIMPLE revised). The difference between SIMPLE and SIMPLER is at the initial steps: in SIMPLER there is an initial set of steps aimed at determining an intermediate pressure field, which is then made equal to p^*, after which the algorithm proceeds in the way of the SIMPLE method. In this way, the problems of divergence that resulted from the omission of $\sum a_F u_F'$ in Equation (49) and lead also to the need for under-relaxation, are reduced and convergence is significantly improved.

To determine the intermediate pressure field the continuity equation is used to obtain a pressure equation, as explained below. One starts again with the momentum Equation (48) where u and p are the values from the previous iteration.

$$u_{i,J} = \frac{\sum a_F u_F + S_{i,J}}{a_{i,J}} + \frac{\left(p_{I-1,J} - p_{I,J}\right)A_{i,J}}{a_{i,J}} \tag{57}$$

In Equation (57) the first term on the rhs is made a pseudo-velocity $\hat{u}_{i,J}$ via

$$\hat{u}_{i,J} \equiv \frac{\sum a_F u_F + S_{i,J}}{a_{i,J}} \tag{58}$$

so that Equation (57) becomes

$$u_{i,J} = \hat{u}_{i,J} + \frac{\left(p_{I-1,J} - p_{I,J}\right)A_{i,J}}{a_{i,J}} \tag{59}$$

which bears similarity to Equation (51). The discretised continuity equation is still Equation (52) and next, Equation (59), and its equivalent expressions for the other nodes are used to substitute terms in the continuity Equation (52), thus resulting the following equation for the pressure field written in canonical form

$$a_{I,J}^P p_{I,J} = \sum_F a_F^P p_F + b_P \tag{60}$$

with all the coefficients a_P, a_F still given by Equation (54), but now with b_P representing an imbalance of pseudo-velocities

$$b_P = b_{I,J} = (\rho \hat{u} A)_{i,J} - (\rho \hat{u} A)_{i+1,J} + (\rho \hat{v} A)_{I,j} - (\rho \hat{v} A)_{I,j+1} \tag{61}$$

The pressure calculated by Equation (60) now serves as an initial pressure field guess which, together with the velocity field from the previous iteration, are used to continue with the SIMPLE algorithm.

So, SIMPLER is made of the following steps:

i) First, there is an initial guess of pressure and velocity fields p^*, u_i^* where they are usually the results of the previous iteration;
ii) Pseudo-velocities are calculated with Equation (58);
iii) The pressure Equation (60) determines an intermediate pressure field p and then p^* is set equal to p;
iv) This better estimate of p^* and the initial velocity guess u_i^* are now the starting point of the original SIMPLE method, which is part of SIMPLER. This better guess of p^* does not impose such a burden on the pressure correction and Equation (45) is solved next to determine a new u_i^*;
v) A pressure correction p' is now obtained with Equation (53);
vi) The pressure field pressure is corrected with Equation (46) and velocity is corrected with Equation (49). At this stage the corrected pressure and velocity fields obey the momentum and continuity equations within a certain residual. In contrast to SIMPLE there should be no need for under-relaxation in SIMPLER;
vii) Finally, the equations for any other quantities ϕ are solved;
viii) If convergence has been attained within a pre-defined residual the iterative calculation process stops, otherwise the newly formed quantities are made equal to p^*, u_i^* and we proceed again to step (i), to initiate a new iteration.

Although SIMPLER resulted in less problems of convergence than SIMPLE, it meant more calculations because of the extra intermediate pressure field calculation.

The problem with SIMPLE was the neglect of $\sum a_F u_F'$ in Equation (49) so Van Doormal and Raithby [14] came up with an alternative to SIMPLE and SIMPLER which they called SIMPLEC (SIMPLE Consistent). Instead of neglecting $\sum a_F u_F'$ in Equation (49), which puts too much of a burden into the pressure correction, $\sum a_F u_F'$ is approximated to a non-zero value, but in a way that avoids the need to solve another system of algebraic equations as in SIMPLER. Instead of setting u_F' to zero as in SIMPLE, Van Doormal and Raithby [14] realised that $\sum a_F u_F'$ is of the order of $\sum a_F u_{i,J}'$. The underlying assumption of $u_F' = u_{i,J}'$ is thus less severe than the assumption in SIMPLE but still has the advantage of leading to an explicit equation on the velocity correction

$$a_{i,J} u_{i,J}' = \sum a_F u_{i,J}' + \left(p_{I-1,J}' - p_{I,J}' \right) A_{i,J} \tag{62}$$

So, SIMPLEC is equal to SIMPLE except that instead of Equation (51) the velocity correction is given by

$$u_{i,J} = u_{i,J}^* + \frac{\left(p_{I-1,J}' - p_{I,J}' \right) A_{i,J}}{a_{i,J} - \sum a_F} \tag{63}$$

In SIMPLEC, correction to the velocity field is not done exclusively by the pressure correction term, as in SIMPLE, but is shared between two terms. This reduces the amount of

pressure correction p' needed during the calculations which is enough to impart stability to the algorithm and to reduce the need for underrelaxation. Everything else remains the same as in SIMPLE, i.e, the sequence of steps in SIMPLEC is identical to the sequence of steps in SIMPLE, except that in step (iv) the velocity is instead corrected with Equation (63).

Both SIMPLEC and SIMPLER are more efficient than SIMPLE, but experiments have shown that it is not easy to determine which is the best (Versteeg and Malalasekera [2], Ferziger and Perić [3]).

3.6- Convergence Criteria

Any iterative computation must terminate at some stage and there should be a criterion to define such moment. If an equation of the form

$$a_P\phi_P - \sum_F a_F\phi_F = Q \tag{64}$$

is being solved iteratively, the calculation would ideally stop when the solution at iteration n (ϕ^n) differed from the exact solution by less than a certain pre-defined error ε. Obviously, except for some control and benchmark cases, the exact solution is unknown and there must be another way of checking that the solution is being approached. An alternative is suggested by writing equation (64) in the following form:

$$a_P\phi_P - \sum_F a_F\phi_F - Q = 0 \tag{65}$$

This equation is only true when the solution ϕ is known, ie, in fact, at some stage in the iterative computation the known field ϕ is such that

$$a_P\phi_P - \sum_F a_F\phi_F - Q = R^n \tag{66}$$

where R^n is called the residual at iteration n. As the residual decreases the exact solution is approached and the error ε^n decreases. Since there is one Equation (66) for each computational cell, all the residuals must decrease for the solution to converge.

In practice, one does not check all the residuals of all equations everywhere, but combine them in a way and check the progression of this result. Two different definitions are commonly used: the L_1 and L_2 norms.

The L_1 norm of the residual is the sum of the absolute residuals, i.e.

$$L_1 = \sum_{j=1}^{N}\left|a_P\phi_P^j - \sum_F a_F\phi_P^j - Q\right| \tag{67}$$

whereas the L_2 norm is the r.m.s

$$L_2 = \sum_{j=1}^{N}\sqrt{\left(a_P\phi_P^j - \sum_F a_F\phi_P^j - Q\right)^2} \tag{68}$$

In both cases N is the number of computational cells in the domain and the equation to be solved should be adequately normalised, ie ϕ should represent a nondimensional quantity.

Although there is proportionality between the error and the residual, it must be stressed that such relationship is problem-dependent. Therefore, if a specific residual is synonymous to a converged solution, within an acceptable error, that does not mean that the same value of the residual will always garantee the same error. Equation (64) can be written in a more compact form as

$$\mathbf{A}\phi = \mathbf{Q} \tag{69}$$

with ϕ representing the vector of unknowns. By definition, at iteration n the error is

$$\varepsilon^n = \phi^n - \phi \tag{70}$$

and the residual is

$$R^n = \mathbf{A}\phi^n - Q \tag{71}$$

Manipulation of Equations (69-72) gives

$$R^n = \mathbf{A}\varepsilon^n \tag{72}$$

This equation clearly shows that in order to compare residuals from different problems the order of magnitude of matrix **A** must be the same in both cases. Amongst other things, the next section illustrates this issue particularly well: a similar pipe flow calculation with yield stress and nonyield stress fluids requires convergence to significantly different residuals in order to attain results having similar errors.

4. Generalized Newtonian Fluids with no Yield Stress

The calculation of laminar flows with GNF fluids has similarities to calculations of turbulent Newtonian flows using turbulence models based on the concept of an eddy viscosity or to laminar Newtonian flows with temperature-dependent viscosity. In these three cases the viscosity varies in the computational field and the similarity facilitates the adaptation of a turbulent Newtonian code for calculations of non-Newtonian laminar flows, as it suffices to eliminate the eddy viscosity and to include the appropriate expression for the molecular viscosity. Since the viscosity depends of a shear rate

$$\dot{\gamma} \equiv \sqrt{2D_{ij}D_{ij}} \quad \text{and} \quad D_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{73}$$

the code must include the calculation of $\dot{\gamma}$, to be carried out on the original grid.

On a turbulent Newtonian flow code this is usually not necessary as the calculation of $\dot{\gamma}$ is already included as part of turbulence models

When the viscosity constitutive equation is a power law

$$\eta(\dot{\gamma}) = K\left(\dot{\gamma}^2\right)^{\frac{n-1}{2}} \tag{74}$$

there is potential for trouble in flows where $\dot{\gamma}$ is very close to zero or attains a value of zero, as in planes of symmetry or in regions of stagnated flow, because the viscosity becomes unbounded and the code may diverge or converge with difficulty.

Our experience has shown this problem to be especially important in the first few iterations, because the initialisation of the velocity fields to constant values produces zero values of $\dot{\gamma}$ everywhere. There are three possible remedies to this problem:

1) To set the viscosity to a constant large value whenever the shear rate $\dot{\gamma}$ is below a predefined small value. If the high viscosity is too large and is set up over a large flow region, the matrices become very stiff and it can take a while for the code to converge to the desired level. However, if the need to set-up a high viscosity is very localised, as in a plane of symmetry, there is usually no problem because in the nearest cells the viscosity is already well-behaved. Convergence with power law fluids has usually no problems although it can take a bit longer than for the similar Newtonian problem. When the degree of shear-thinning is intense, say for n= 0.2, the required number of iterations can be two to three times those for the equivalent Newtonian flow simulation;
2) If it is known in advance that there are no stagnated flow regions, or that such regions are very localised, the calculation can be initialised, and some iterations performed, with a constant viscosity, after which the calculation proceeds with the true variation of viscosity;
3) To substitute the power law by a viscosity model which possesses a first Newtonian plateau at low shear rates, followed by a power law region. One such case is the simplified Carreau-Yasuda model given by

$$\eta(\dot{\gamma}) = \eta_0 \left[1 + (\lambda \dot{\gamma})^a \right]^{\frac{n_c - 1}{a}} \tag{75}$$

The user must be careful to set well the various model parameters: naturally, $n_c = n$ and $K = \eta_0 \lambda^{n_c - 1}$ but the selection of η_0 and λ requires an idea of the range of flow rates (or time scales) prevailing in the flow. Equations (74) and (75) are represented in Figure 5; λ represents the reciprocal of the shear rate separating the constant and power law viscosity regions.

Thus, the value of λ must be well above the estimated reciprocal of the minimum shear rate in the flow. A good rule of thumb is $\lambda \approx 100/\dot{\gamma}_{\min}$. Note that, if $\dot{\gamma}_{\min} = 0$ as in a given plane of symmetry, $\dot{\gamma}$ should not be calculated at the plane of symmetry, but at the nearest cell centre. Parameter a of the Carreau model determines the rate at which the viscosity of the Carreau-Yasuda model changes from a power law behaviour to a constant viscosity behaviour in the vicinity of $\dot{\gamma} = 1/\lambda$. The most common value of a used for fitting true viscosity data is a= 2, for which the simplified Carreau model is recovered, but a higher value can be used. The higher the value of a the more sudden the transition between these two regions will be.

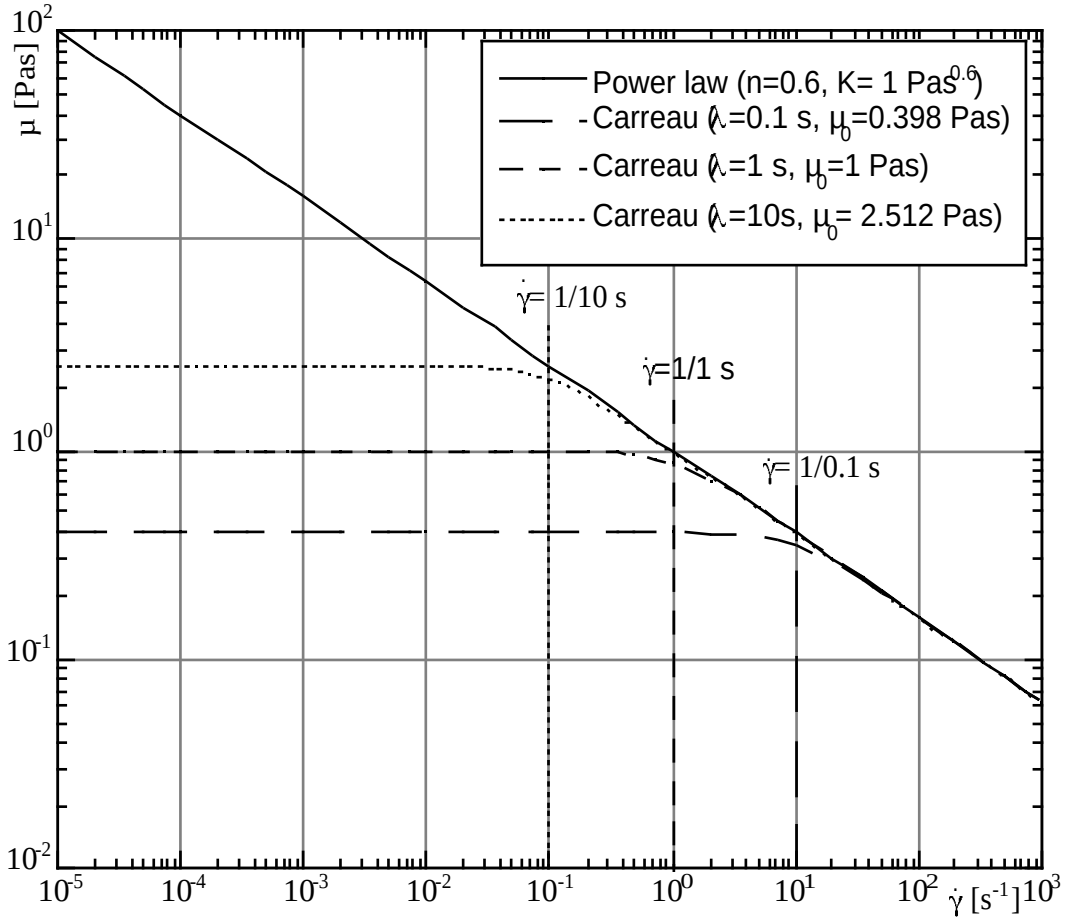
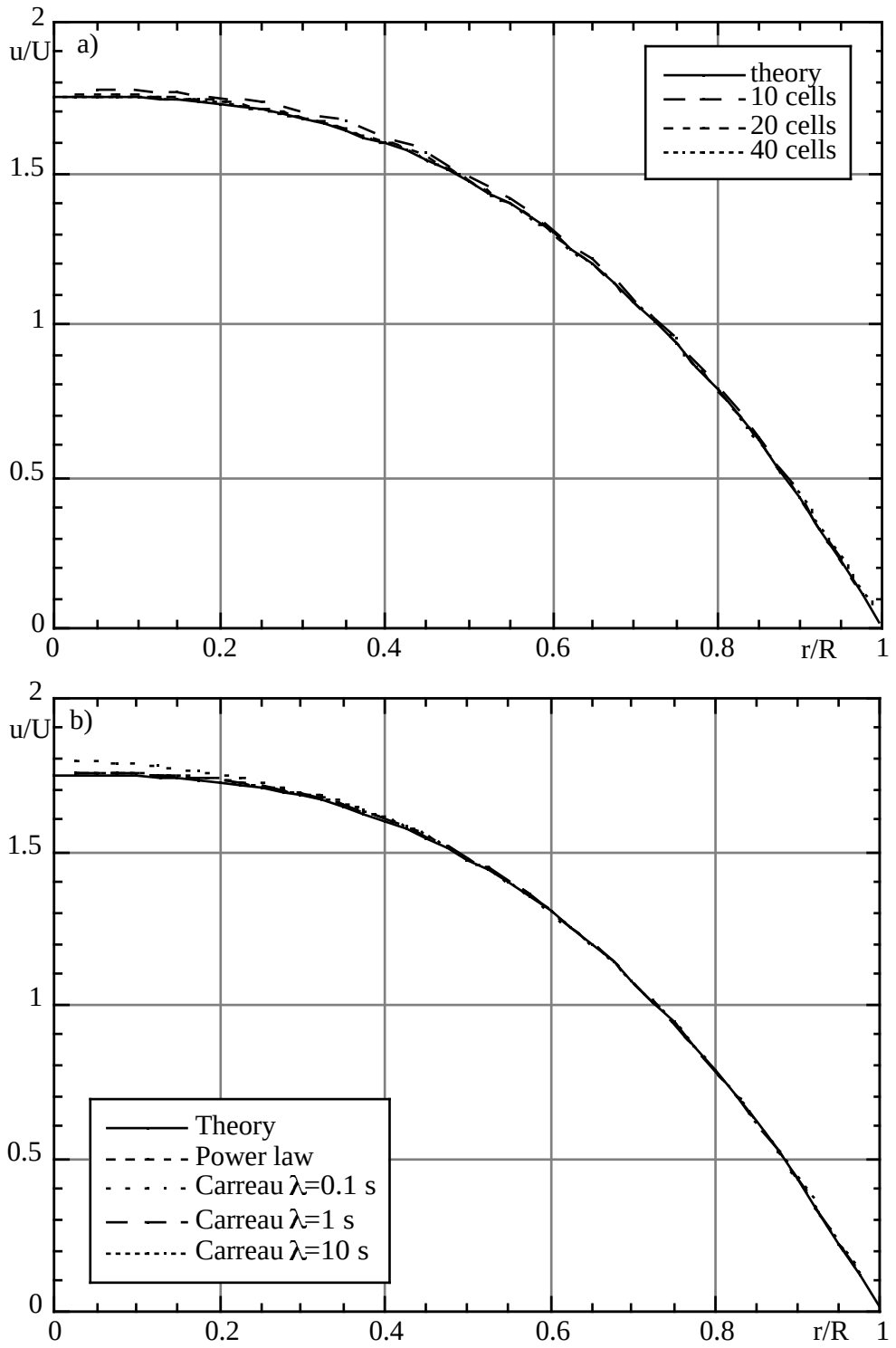


Figure 5- The effect of λ on the comparison between a simplified Carreau model (a= 2) and a power law.

Table I- Comparison between the calculated and theoretical friction factors for the fully-developed laminar pipe flow of power law and simplified Carreau fluids (Re_{gen} = 5.261).

Cases	f	f_{theory}	Error [%]
Power law: 10 cells	12.314	12.165	1.22
Power law: 20 cells	12.200	12.165	0.29
Power law: 40 cells	12.185	12.165	0.16
Carreau 20 cells: λ = 0.1 s	12.124	12.165	-0.34
Carreau 20 cells: λ =1 s	12.210	12.165	0.37
Carreau 20 cells: λ = 10 s	12.211	12.165	0.38

In Figure 6-a) the normalised theoretical velocity profile for laminar pipe flow of the power law of Figure 5 is compared with predictions made in uniform meshes with different degrees of refinement. The calculations were performed with the hybrid scheme for the convective term of the momentum equation. The use of 20 cells in the radial direction provides a velocity profile



Figures 6- Velocity profiles in fully-developed laminar pipe flow for a power law fluid with n= 0.6. **a)** Mesh refinement effect and comparison with theory; **b)** Comparison between theory and predictions with the power law and the simplified Carreau model with different values of λ (20 radial cells grid). All calculations converged to normalised residual of 10^{-4}.

very close to the theoretical curve and the corresponding values of the Darcy friction factor are listed and compared in Table I.

For the same pipe flow, simulations were carried out adopting the Carreau model of Figure 5 in the 20 cell mesh and the results are compared in Figure 6-b) and Table I. Except for $\lambda = 0.1$ s, the simulations with $\lambda = 1$ s and $\lambda = 10$ s predict well the power law velocity profile and the corresponding friction factor. From the power law calculation, the minimum value of shear rate was around 0.3 s^{-1} in the 20 cells grid and of 0.095 s^{-1} for the 40 cells grid. Compare these values with Figure 5 to understand the limitations of the Carreau model in substituting a power law equation. An objective criteria for this kind of substitution has been formulated by Escudier et al [15] in the context of annular flows for various viscosity models. The criteria can be easily adapted to other flow geometries.

5. Yield Stress Fluids

In contrast to non-yield stress fluids, calculations with yield stress fluids pose severe problems of convergence and of accuracy. In regions of unyielded fluid the shear rates are zero and the viscosities become unbounded as given by the Herschel-Bulkley viscosity model of Equation (67).

$$\begin{aligned} \eta(\dot{\gamma}) &= K\dot{\gamma}^{n-1} + \frac{\tau_Y}{\dot{\gamma}} \qquad \text{for } \tau > \tau_Y \\ \dot{\gamma} &= 0 \qquad\qquad\qquad\quad \text{for } \tau \le \tau_Y \end{aligned} \tag{76}$$

Even when using an adequate modification of the yield stress viscosity model, to be presented in this section, the viscosities in the unyielded regions are very high and convergence becomes extremely slow.

The number of iterations required for convergence of a specific flow problem can be 10 to 100 times larger than for the equivalent non-yield stress fluid problem. Two reasons contribute to this discrepancy:

i) The high viscosities in the unyielded flow regions increase the stiffness of the matrices and this slows down convergence considerably;

ii) As will be seen in Section 5.4, the required convergence criteria (normalised residual), for a given level of accuracy, must be 100 to 1000 times smaller than for the equivalent non-yield stress flow problem.

A first impression of the problem in hand can be grasped by looking at Figure 7 which plots the theoretical velocity and shear stress profiles for a Bingham plastic flow in a pipe, the case which will be used for simulations later in the section.

To avoid unbounded viscosities the original viscoplastic model must be modified following one of the three strategies explained below: the bi-viscosity model of Beverly and Tanner [16], a modified bi-viscosity model and the model of Papanastasiou [17].

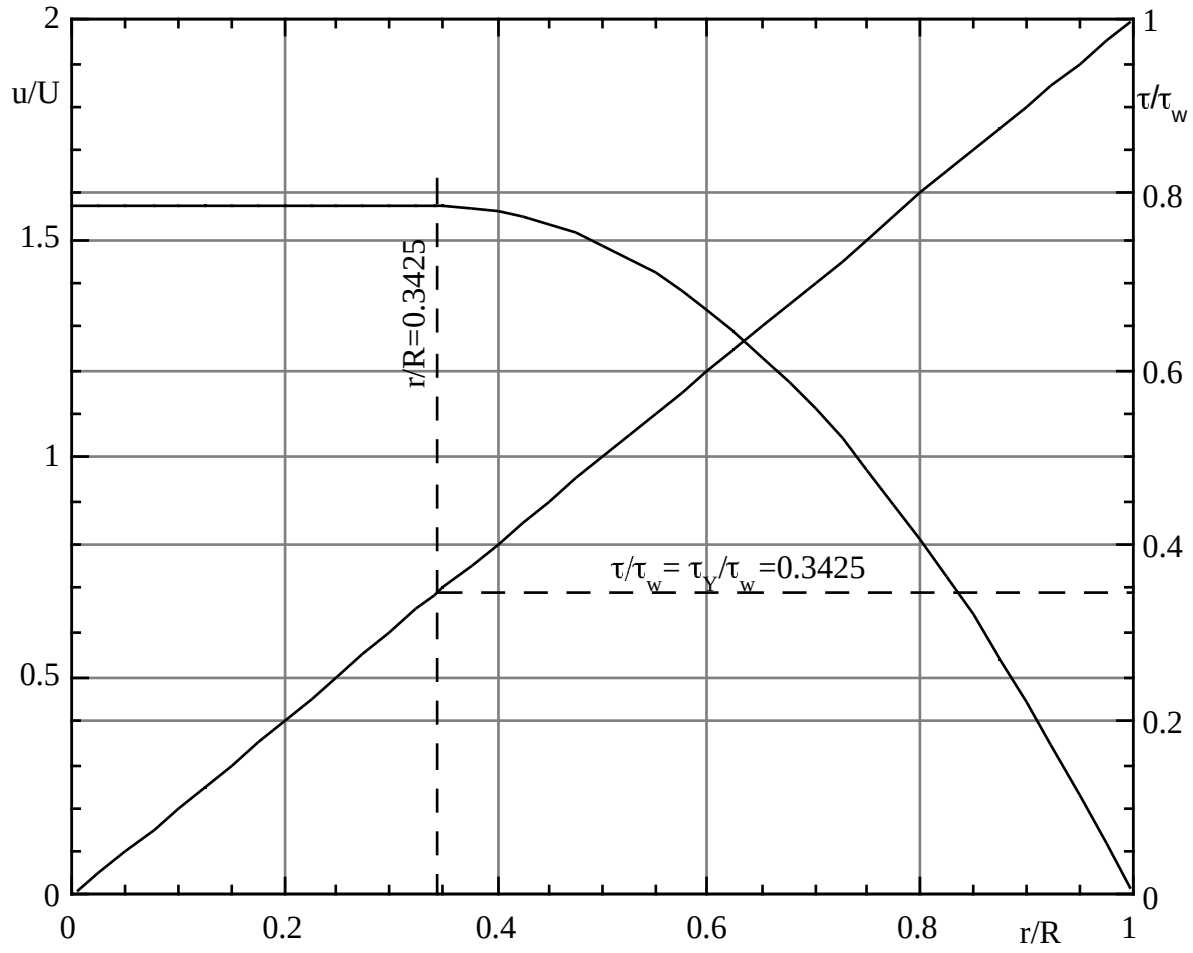


Figure 7- Normalised velocity and shear stress radial profiles for laminar pipe flow of a Bingham plastic: K = 0.2 Pas, n= 1, τ_Y = 10 Pa, U = 0.1 m/s (see Equation 76). τ_w is the full shear stress at fully-developed flow and U is the bulk velocity.

5.1. The bi-viscosity model

The bi-viscosity model was originally introduced by Beverly and Tanner [16] for Bingham plastics but is easily extended to other fluids, as Soares et al [18] did for the Herschel-Bulkley fluid. The bi-viscosity modification of the Herschel-Bulkley fluid is

$$\begin{aligned} \eta &= \eta_{lr} && \text{for } |\dot{\gamma}| \le \dot{\gamma}_c \\ \eta(\dot{\gamma}) &= K\dot{\gamma}^{\,n-1} + \frac{\tau_Y}{\dot{\gamma}} && \text{for } |\dot{\gamma}| > \dot{\gamma}_c \end{aligned} \tag{77}$$

where the critical shear rate $\dot{\gamma}_c$ results from the intersection of the two expressions and is given by the non-linear implicit equation

$$\eta_{lr}\dot{\gamma}_c = K\dot{\gamma}_c^{\,n} + \tau_Y \tag{78}$$

This equation becomes explicit for the case of a Bingham plastic , where μ is the plastic viscosity ($\mu = K$ if n= 1 in Equation 74)

$$\dot{\gamma}_c = \frac{\tau_Y}{\eta_r - \mu} \tag{79}$$

The value of η_r must be high to better represent the original model but, if it is too high, matrices become too stiff and a converged solution will be difficult to obtain. A good compromise for Bingham plastics is $300 \le \eta_r/\mu \le 1000$ and this is recommended by Beverly and Tanner [16], Soares et al [18] and Vradis and Ötügen [19]. Another criterion, recommended by Vradis and Ötügen [19], is $\eta_r \ge 1000\,\tau_Y\,R/U$ which in some cases gives higher values of η_r than the previous criterion.

The use of such modifications of the original constitutive equation does not provide a solution to all difficulties. The bi-viscosity model, or any of the other remedies to be presented, is a cure to the unbounded viscosity issue but convergence is very slow because of the high viscosities and the strict convergence criteria required for accurate solutions. Therefore, before embarking in a calculation programme the researcher must carefully perform a series of test calculations, for which there are reliable results, in order to select the adequate range of values of η_r for his own problem. In this preliminary study, of particular concern should be the type of result pretended: are they simply the bulk flow characteristics, such as the friction factor or a Nusselt number, for which the convergence criteria need not be so tight, or is it important to be able to predict very accurately the flow field with emphasis at separating yielded and unyielded regions?

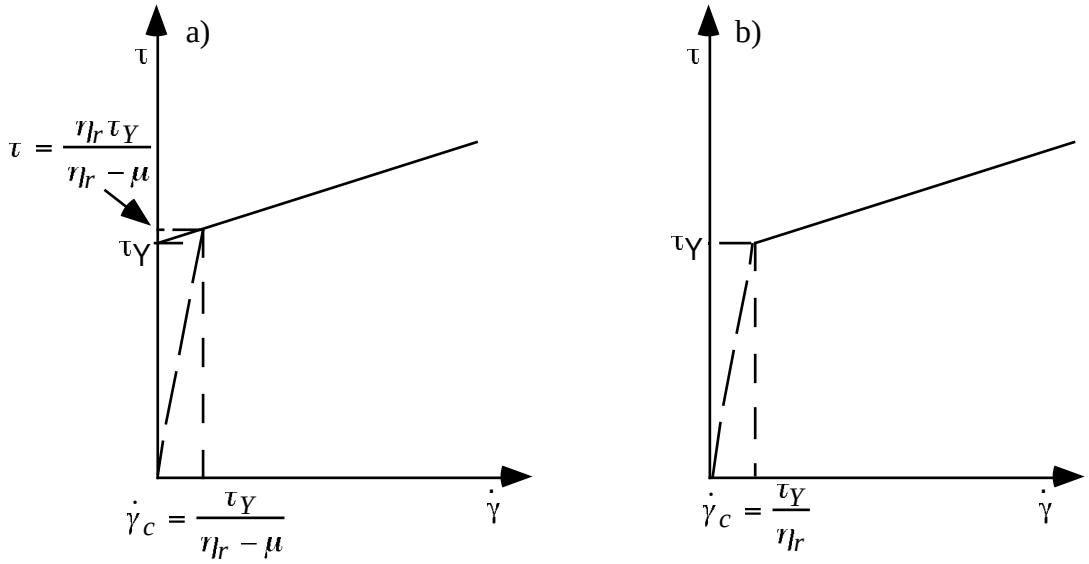


Figure 8- Schematic representation of the biviscosity (a) and modified biviscosity (b) models applied to a Bingham plastic.

5.2. The modified bi-viscosity model

The problem with the bi-viscosity model is that it is less straighforward in two aspects:

i) For an Herschel-Bulkley fluid the value of $\dot{\gamma}_c$ must be obtained by solving the nonlinear Equation (78) and the result is not known *a priori*, i.e, there is no general expression for $\dot{\gamma}_c$;

ii) The yield stress value τ_Y occurs (see Figure 8-a for the rheogram of a Bingham plastic) for a shear rate corresponding to the high viscosity region, i.e, the value of shear stress that marks the separation between yielded and unyielded flow regions pertains to the high viscosity range of shear rates which should correspond only to an unyielded region. An alternative would be to consider that the shear stress marking the yield/unyielded transition in the bi-viscosity model is equal to $\tau_c = K\dot{\gamma}_c^n + \tau_Y$ but that would still lead to a non-linear implicit expression on τ_c, an undesirable feature according to (i) and the critical shear stress would now be larger than the yield stress τ_Y, though by a very small amount.

This inconsistency can be resolved by using the modified bi-viscosity model which is given by

$$\begin{aligned} \eta &= \eta_r && \text{for } |\dot{\gamma}| \le \dot{\gamma}_c \\ \eta(\dot{\gamma}) &= K\dot{\gamma}^{\,n-1} + \frac{\tau_Y - K\dot{\gamma}_c^n}{\dot{\gamma}} && \text{for } |\dot{\gamma}| > \dot{\gamma}_c \end{aligned} \tag{80}$$

with $\dot{\gamma}_c$ now given by $\dot{\gamma}_c = \frac{\tau_Y}{\eta_r}$. As regards the value of η_r, the recommendation is exactly the same as for the original biviscosity model of the previous section.

In the modified bi-viscosity model, represented schematically in Figure 8-b), there is an intersection of the high viscosity region with an equation which represents a slight modification of the Herschel-Bulkley equation. The difference between the biviscosity and modified biviscosity models is small given the range of values for parameter η_r recommended in the literature, as confirmed in Figure 9, but the use of the modified equation is to be preferred by those who wish to have no ambiguity concerning the values of $\dot{\gamma}_c$ and the corresponding value of the shear stress. In Figure 9, a specific rheogram of the Bingham model is compared with that of the corresponding bi-viscosity, modified bi-viscosity and Papanastasiou models for a set of parameters. The graph was zoomed in the region of interest and differences between the bi-viscosity, the Papanastasiou and the original Bingham model cannot be discerned for shear rates above 0.2 s^{-1}. However, the modified bi-viscosity model is slightly below the Bingham equation in the range of shear rates displayed. The difference is negligible though and diminishes with $\dot{\gamma}$. At low shear rates the Papanastasiou equation, to be explained below, is clearly different from the two bi-viscosity models and approaches the original Bingham law in a much better way.

5.3. The Papanastasiou model

The Papanastasiou model, represented in Figure 9 with small dashes, can also be adapted to any yield stress viscosity model and in Equation (81) it is presented as a substitute of the Herschel-Bulkley equation

$$\eta(\dot{\gamma}) = K\dot{\gamma}^{n-1} + \frac{\tau_Y}{\dot{\gamma}}\left(1 - e^{-m\dot{\gamma}}\right) \tag{81}$$

The advantage of this model is that it is written as a single equation but its disadvantage is the capacity to distinguish between yielded and unyielded regions. In a calculation the values of shear rate will never be zero, so the user must define a criterion below which the shear rate will be considered as pertaining to an unyielded region. For the bi-viscosity and modified bi-viscosity models, that is set by the value of η_r in one way or another, whereas in the Papanastasiou equation such separation is not inherent to the model.

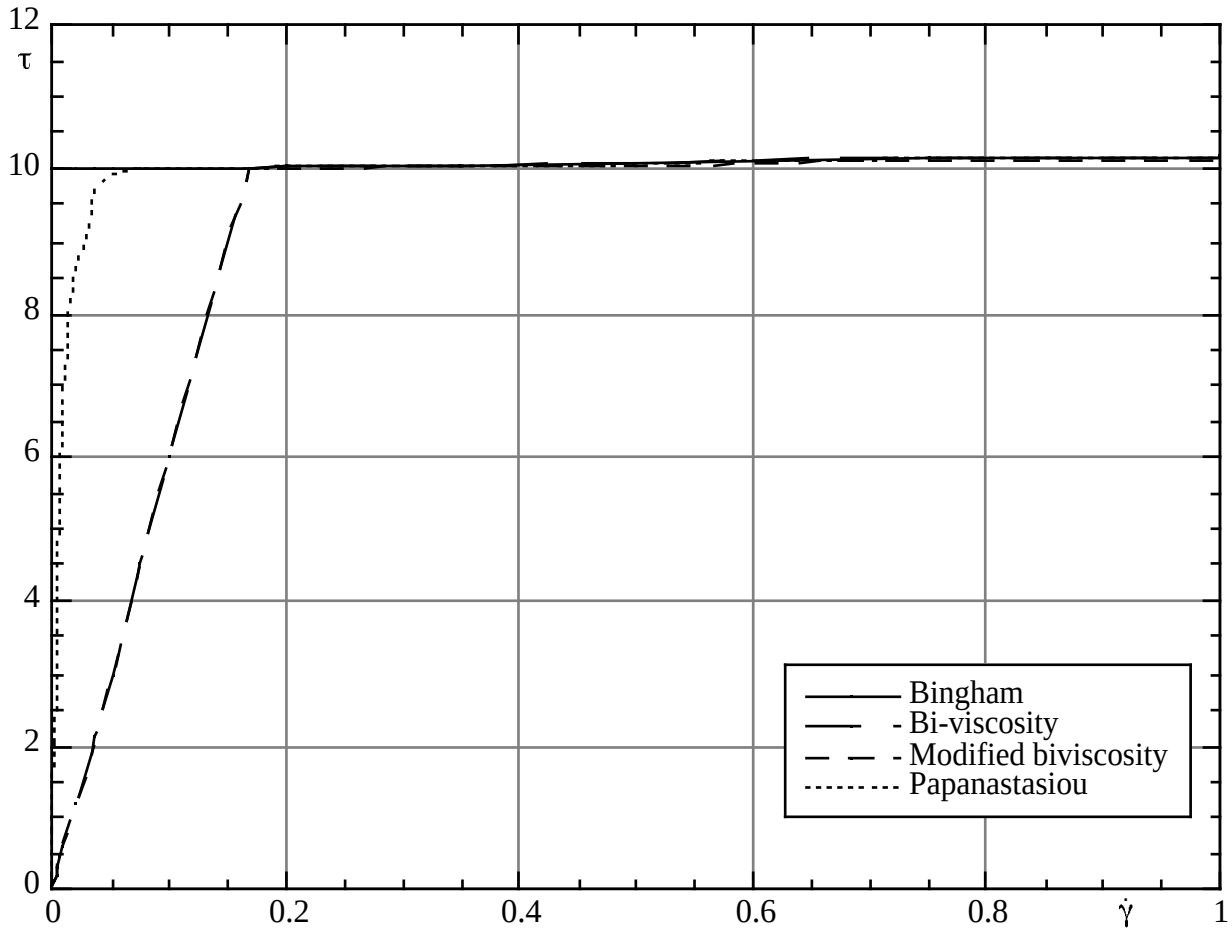


Figure 9- Comparison between th shear stress versus shear rate of a Bingham plastic (μ = 0.2 Pas, τ_Y = 10 Pa), with that of the corresponding biviscosity ($\eta_r = 60\mu$), modified biviscosity ($\eta_r = 60\mu$) and Papanastasiou ($m = 100$) models.

Recommended values for parameter m are $m \geq 100$ (Papanastasiou [17]) or $mBi \geq 300$ (Meuric et al [20]), where the Bingham number (Bi) is defined as $Bi \equiv \tau_Y D_h / (\mu U)$ with D_h representing the hydraulic diameter. As with the bi-viscosity models, the user is advised to test the sensitivity of the results to the numerical value of the parameter m.

The mathematical simplicity of the Papanastasiou model may convince the reader of its apparent superiority, but that is misleading. Our own experience and of João [21] have shown that the Papanastasiou model can take much longer to converge to a solution of equal accuracy than any of the biviscosity models, at least when following the recommendations in the literature for the values of the corresponding parameters.

The interested reader is advised to carefully test the models before embarking on a long research programme, specifically looking at convergence rates and solution accuracy for the particular problem under investigation.

5.4. Comparison of models

To compare the performance of the various modifications of the yield stress models calculations were made of fully-developed laminar pipe flow of the Bingham plastic represented in Figure 9. The bulk velocity was 0.1 m/s and the pipe diameter was 10 mm. Calculations were carried out with the bi-viscosity, the modified bi-viscosity and Papanastasiou models, for values of the parameters encompassing the recomended ranges, in three different uniform meshes having 10, 20 and 40 cells in the radial direction. Due to its symmetry, the flow domain went from the axis to the wall.

This assessment is initiated by looking at the convergence criterion. It is worth mentioning that all the calculations reported in Section 4 were carried out until the normalised residual in all equations fell below 1×10^{-4}. This is a usual value for Newtonian and nonyield stress Generalised Newtonian fluid calculations and provides the kind of agreement and accuracy shown in Figure 6 and Table I. However, this picture is totally different for yield stress fluids.

In a series of calculations with the modified bi-viscosity model the effect of the convergence criterion on the results was assessed and the results are shown in the radial profiles of Figure 10 and in Table II. The results for a residual of 1×10^{-4} are poor and a normalised residual of at least 1×10^{-6} was required for an accurate result (differences below 1%). Even with this small residual, and also for the 1×10^{-7} case, the values of the velocity in the plug region are overpredicted by 0.6%, a difference in excess to that seen in Figure 6-a) for a power law fluid calculated with the same mesh. For the two lower residuals the velocity data in the central region of Figure 10 was underpredicted. The corresponding friction factors listed in Table II confirm the picture: for a normalised residual of 1×10^{-4} the error is 20% and to be within 0.5% of the theoretical value a residual of at least 1×10^{-6} had to be enforced.

Table II- Comparison between the theoretical and calculated friction factors of Figure 10.

Normalised residual	f	f-theoretical	error [%]
1E-4	7.007	5.840	20.0
1E-5	6.121	5.840	4.8
1E-6	5.846	5.840	0.1
1E-7	5.819	5.840	-0.36

This simple comparison shows the need for tight convergence criteria (at least 100 times stricter than for nonyield stress fluids) when performing numerical calculations with yield stress fluids.

Next, using a normalised residual of 1E-7, a mesh refinement investigation was carried out with the modified bi-viscosity model. The normalised radial velocity profiles are presented in Figure 11 and the corresponding friction factors are compared in Table III.

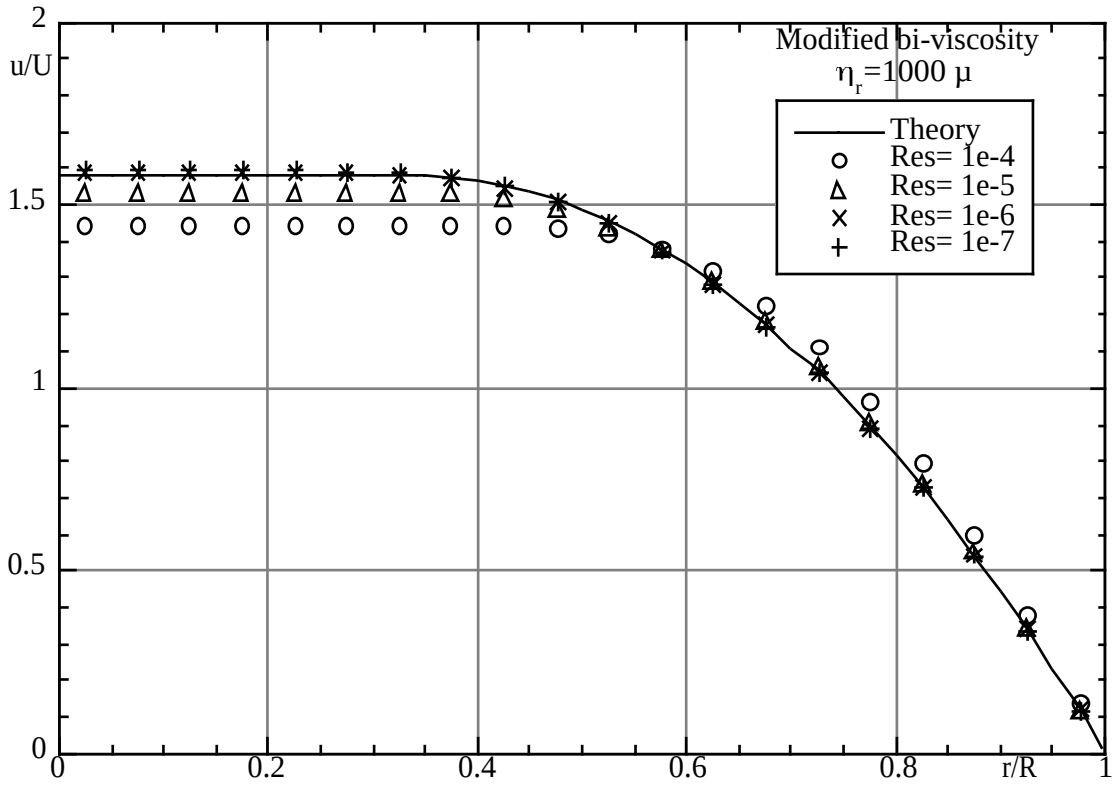


Figure 10- Radial velocity profiles in a pipe for a Bingham plastic. In all cases the mesh had 20 cells in the radial direction.

There are obvious improvements as the mesh is refined, but this effect is not as large as was seen with power law fluids in Section 4 andthis difference is dueto the smaller residual. Had we done the mesh refinements with a higher residual, the difference would have been larger. For instance, even for a mesh having 10 radial cells the predictions with a residual of 1E-7 are far

more accurate than with 20 cells and a residual of 1E-5. This clearly shows that for yield stress fluids it is at least as important to converge the calculations to a very small residual as it is to refine the mesh. For the other two models similar variations would be observed.

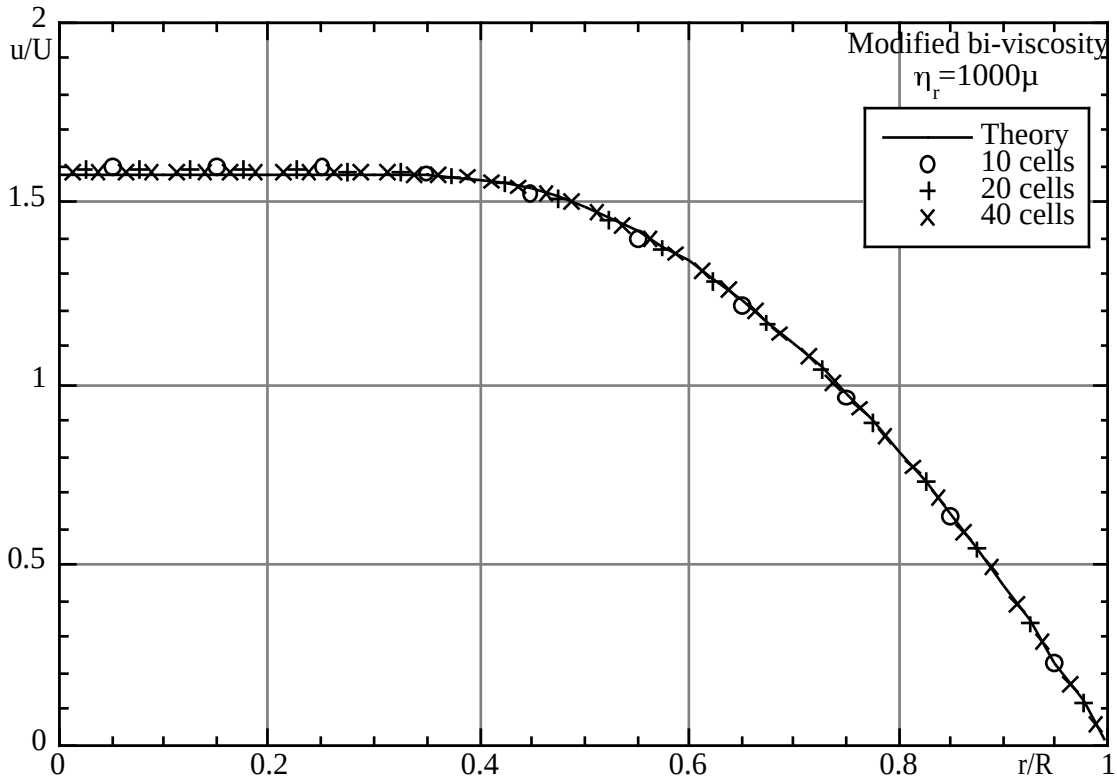


Figure 11- Radial velocity profiles in a pipe for a Bingham plastic. Mesh refinement effect. Normalised residual = 1E-7.

Table III- Comparison between the theoretical and calculated friction factors of Figure 11**.**

Mesh	f	f theoretical	error [%]
10 cells	5.805	5.840	-0.60
20 cells	5.819	5.840	-0.36
40 cells	5.822	5.840	-0.31

The two bi-viscosity models are now compared in Figures 12-a and -b) and in Table IV: in Figure 12-a) the high viscosity was $\eta_r = 1000\mu$, whereas in Figure 12-b) $\eta_r = 300\mu$. The residuals in all cases were 1×10^{-7} and differences in the velocity profile are not distinguishable. In terms of the friction factors listed in Table IV the differences between the two methods are very small less than 0.1%. In this case, using a criterion of $\eta_r / \mu = 1000$ provides only a marginally better prediction

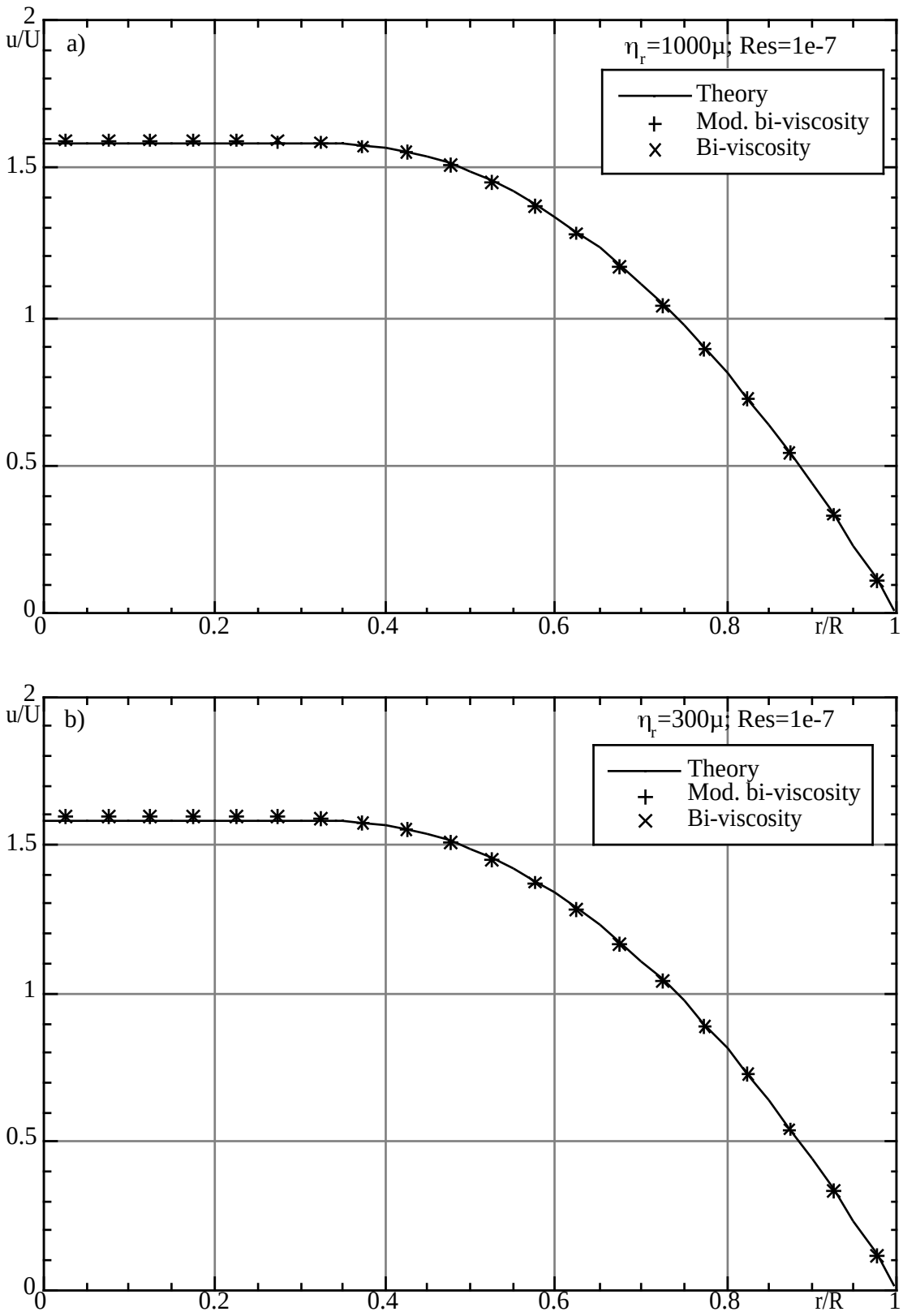


Figure 12- Radial velocity profiles in a pipe for a Bingham plastic. Mesh with 20 radial cells and normalised residual of 1E-7. Comparison between the Modified biviscosity and biviscosity models. **a)** $\eta_r / \mu = 1000$; **b)** $\eta_r / \mu = 300$.

Table IV- Comparison between the theoretical and calculated friction factors of Figure 12**.**

Model	f	f theoretical	error [%]
Biviscosity $\eta_0 / \mu = 1000$	5.814	5.840	-0.45
Modified Biviscosity	5.819	5.840	-0.36
Biviscosity $\eta_0 / \mu = 300$	5.808	5.840	-0.55
Modified Biviscosity	5.817	5.840	-0.39

Table V- Comparison between the theoretical and calculated friction factors of Figure 13.

Model	f	f theoretical	error [%]
Biviscosity	5.814	5.840	-0.45
Modified Biviscosity	5.819	5.840	-0.36
Papanastasiou	5.829	5.840	-0.19

Similar findings were reached by other authors: Hammad *et al* [22] used the bi-viscosity modification for laminar sudden expansion calculations and report similar difficulties and experiences when dealing with yield stress fluids.

Finally, in Figure 13 and Table V the Papanastasiou model is compared with the two biviscosity models. For the Papanastasiou model m was set to 100 and for the two biviscosity models $\eta_0 / \mu = 1000$ was used. For the Papanastasiou model both the velocity profile in the unyielded region and the friction factor are closer to the theoretical values than the data from the other two models, a result which is not surprising given the better approximation of the Bingham equation seen in Figure 9. However, the calculation with the Papanastasiou model took approximately twice as long as those with the two biviscosity models. This was expected given that the Papanastasiou model approaches better the original Bingham law, i.e., it results in higher viscosities in the unyielded region than the bi-viscosity equations, hence leading to stiffer matrices more difficult to converge. A decrease in the value of m could certainly provide a quicker convergence at the cost of lower accuracy. It is up to the user to decide what is more important in a particular solution and the concessions that can be made to the two conflicting parts of the problem: the accuracy of the solution versus the convergence.

To conclude this section, it is clear from these comparisons that the major factor affecting the quality of the predictions is the convergence criteria: the normalised residual must be set to a much lower value than is usual for Newtonian and nonyield stress Generalised Newtonian calculations. It is also clear from these comparisons that the quality of the predictions for yield stress fluids is not as good as those obtained for nonyield stress fluids because the viscosity model must be modified. Once their parameters have been well selected, the two bi-viscosity models presented for substituting the original Herschel-Bulkley equation provide similar results and accuracy. The Papanastasiou model tends to be marginally more accurate but its convergence is slower.

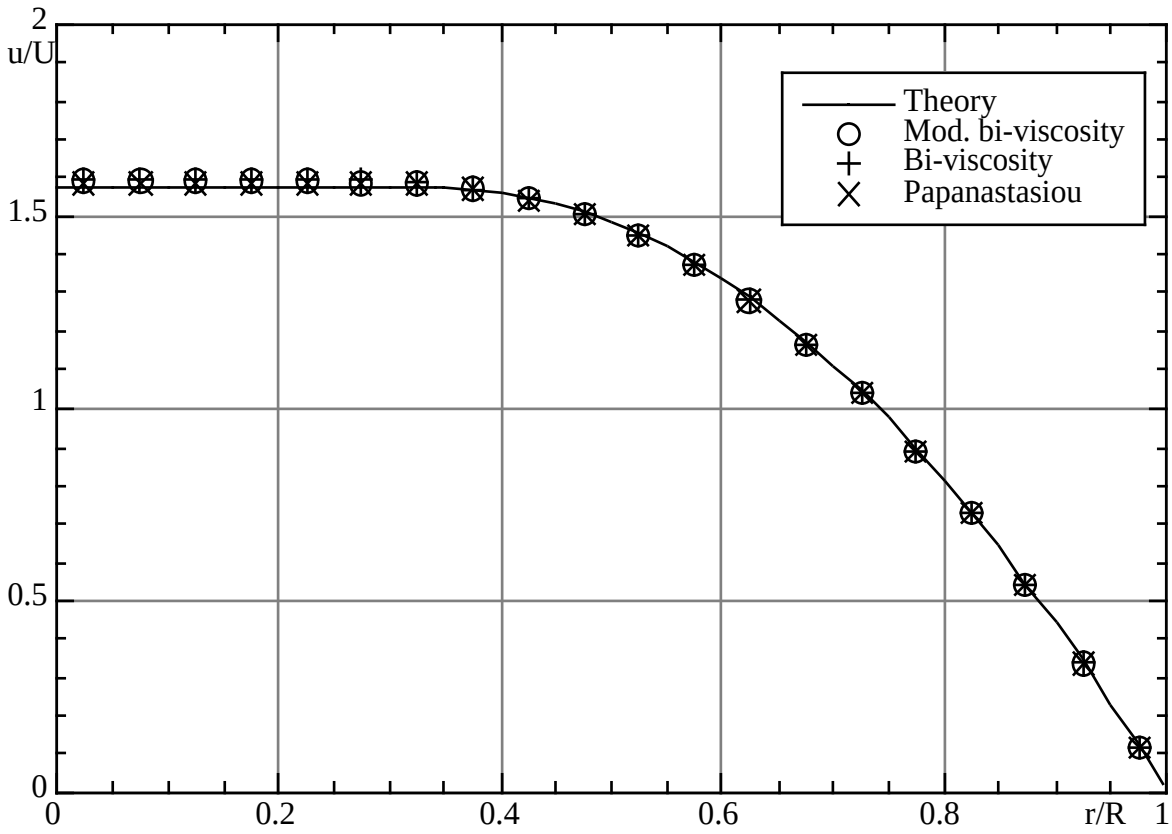


Figure 13- Radial velocity profiles in a pipe for a Bingham plastic. Mesh with 20 radial cells and normalised residual of 1E-7.20 cells. Comparison between the Modified biviscosity, biviscosity and Papanastasiou models: $\tau_y / \mu = 1000$ and $m = 100$.

Acknowledgements

The author acknowledges the helpfull discussions with Prof. P. J. Oliveira of Universidade da Beira Interior, Covilhã, Portugal.

References

[1] Patankar, SV 1980. *Numerical Heat Transfer and Fluid Flow.* McGraw-Hill, New York.

[2] Versteeg, HK and Malalasekera, W 1995. *An Introduction to Computational Fluid Dynamics. The Finite Volume Method.* Longman, Harlow.

[3] Ferziger, JH and Perić , M 2002. *Computational Methods for Fluid Dynamics*. Springer Verlag, 3rd edition, Berlin, Germany.

[4] Pinho, FT and Oliveira, PJ 2001. A metodologia dos volumes finitos aplicada à reologia computacional. I: Introdução. *e-rheo.pt*, 1, 1.

[5] Wapperom, P and Webster, MF 1999. Simulation for viscoelastic flow by a finite volume/element method. *Comput. Methods Appl. Mech. Eng.*, 180, 281-304.

[6] Spalding, DB 1972. A novel finite-difference formulation for differential expressions involving both first and secondderivatives. *Int. J. num. Methods Eng.*, 4, 551-559.

[7] Leonard, BP 1979. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Comput. Methods Appl. Mech. Eng.*, 19, 59-98.

[8] Harten, A 1983. High-resolution schemes for hyperbolic conservation laws. *J. Comput. phys.*, 49, 357-393.

[9] Alves, MA, Pinho, FT and Oliveira, PJ 2000. Effect of a high-resolution differencing scheme on finite-volume predictions of viscoelastic flows. *J. Non-Newt. Fluid Mech.*, 93, 287-314.

[10] Alves, MA, Pinho, FT and Oliveira, PJ 2001. The flow of viscoelastic fluids past a cylinder: finite-volume high-resolution methods. *J. Non-Newt. Fluid Mech.*, 97, 205-230.

[11] Patankar, SV and Spalding, DB 1972. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat and Mass Transfer*, 15, 1787.

[12] Harlow, FH and Welch, JE 1965. Numerical calculation of time dependent viscous incompressible flow of fluid with free surface. *Phys. Fluids*, 8, 2182-2189.

[13] Rhie, CM and Chow, WL 1983. Numerical study of the turbulent flow past an aerofoil with trailing edge separation. *AIAA J*, 21, 1525-1532.

[14] Van Doormal, JP and Raithby, GD 1984. Enhancements of the SIMPLE method for predicting incompressible fluid flows. *Num. Heat Transf.*, 7, 147-163.

[15] Escudier, MP, Oliveira, PJ and Pinho, FT 2001. Fully developed laminar flow of purely viscous non-Newtonian liquids through annuli, including the effects of eccentricity and inner-cylinder rotation. *Int. J. Heat and Fluid Flow*, in press.

[16] Beverly, CR and Tanner, RI 1992. Numerical analysis of three-dimensional Bingham plastic flow. *J. Non-Newt. Fluid Mech.*, 42, 85-115.

[17] Papanastasiou, TS 1987. Flows of materials with yield. *J. Rheology*, 31, 385-404.

[18] Soares, EJ, Naccache, MF and Mendes, PRS 1998. Heat transfer to Herschel-Bulkley materials in annular flows. Proceedings of VII ENCIT- Pontifícia Universidade Católica, Rio de Janeiro, RJ Brasil (November), 1146-1151.

[19] Vradis, GC and Ötügen, MV 1997. The axisymmetric sudden expansion flow of a non-Newtonian viscoplastic fluid. *J. Fluids Eng.*, 119, 193-200.

[20] Meuric, OFJ, Wakeham, RJ, Chiu, TW and Fisher, KA 1998. Numerical flow simulation of viscoplastic fluids in annuli. *Can J. Chem. Eng.*, 76, 27-41.

[21] João, AMR 2001. Annular flows of non-Newtonian fluids (in portuguese). MSc Thesis, University of Porto, Portugal.

[22] Hammad, KJ, Vradis, GC and Ötügen, MV 2001. Laminar flow of a Herschel-Bulkley fluid over an axisymmetric sudden expansion. *J. Fluids Eng.*,123, 588-594.