



HAL
open science

A Characteristics Strategy for Solving Advection Equations in 2D Steady Flows Containing Recirculating Areas

Francisco Chinesta, Guillaume Chaidron

► **To cite this version:**

Francisco Chinesta, Guillaume Chaidron. A Characteristics Strategy for Solving Advection Equations in 2D Steady Flows Containing Recirculating Areas. *Computer Methods in Applied Mechanics and Engineering*, 2003, 192 (37-38), pp.4217-4235. 10.1016/S0045-7825(03)00405-5 . hal-00020744

HAL Id: hal-00020744

<https://hal.science/hal-00020744>

Submitted on 16 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A characteristics strategy for solving advection equations in 2D steady flows containing recirculating areas

F. Chinesta ^{a,*}, G. Chaidron ^b

^a *Laboratoire de Mécanique des Systèmes et des Procédés, Ecole Nationale Supérieure d'Arts et Métiers (LMSP-ENSAM), 151 Boulevard de l'Hôpital, F-75013 Paris, France*

^b *Laboratoire de Rhéologie et Thermodynamique des Matériaux Macromoléculaires, Conservatoire National des Arts et Métiers, 292 rue Saint Martin, F-75141 Paris Cedex 03, France*

Numerical modelling of non-Newtonian flows usually involves the coupling between equations of motion characterized by an elliptic character, and the fluid constitutive equation, which defines an advection problem linked to the fluid history. There are different numerical techniques to treat the hyperbolic character of advection equations. In non-recirculating flows, Eulerian discretisations can give an accurate mesh size dependent solution within a short computing time. However, the existence of steady recirculating flow areas induces additional difficulties. Actually, in these flows neither boundary conditions nor initial conditions are known. In a former paper we have proved that in such flows Eulerian techniques lead to solutions with significant deviations from the exact one. These deviations obviously decrease as the mesh density increases. In other paper, the authors have proved that some linear advection equations modelling non-Newtonian fluid behaviors have only one solution in steady recirculating flows. This solution is found imposing the solution periodicity along the closed streamlines, where the equation is integrated by the method of characteristics. In this paper we propose a characteristics algorithm for solving advection equations in general steady flows, which may contain recirculating areas.

Keywords: Advection equation; Two-dimensional steady recirculating flows; Method of characteristics; Non-Newtonian fluid mechanics

1. Introduction

Numerical modelling of non-Newtonian fluid flows usually involves the coupling between motion equations, which leads to an elliptic problem, and the fluid constitutive equation, which introduces an advection problem related to the fluid history.

* Corresponding author.

E-mail address: francisco.chinesta@paris.ensam.fr (F. Chinesta).

For example, in short fiber suspensions, the fiber orientation is usually described from a second-order tensor $\underline{\underline{s}}$, which is symmetric and has a unit trace. This tensor can be written, under a certain number of assumptions [1–3], as $\underline{\underline{s}} = \frac{\underline{\underline{a}}}{\text{Tr}(\underline{\underline{a}})}$, with the evolution of tensor $\underline{\underline{a}}$ given by

$$\frac{d\underline{\underline{a}}}{dt} = \underline{\underline{\Omega}}\underline{\underline{a}} - \underline{\underline{a}}\underline{\underline{\Omega}} + k(\underline{\underline{D}}\underline{\underline{a}} + \underline{\underline{a}}\underline{\underline{D}}) - D_r \left(\underline{\underline{a}} - \frac{I}{3} \right), \quad (1)$$

where $\underline{\underline{a}}$ is a symmetric second-order tensor, which usually verifies $\text{Tr}(\underline{\underline{a}}) \neq 1$, $\underline{\underline{D}}$ represents the strain rate tensor, $\underline{\underline{\Omega}}$ is the vorticity tensor, k is a constant which depends on the fibers aspect ratio and D_r is a diffusion coefficient that takes into account the fibers interactions.

From a physical point of view, we can consider that the eigenvalues of the orientation tensor $\underline{\underline{s}}$ represent the probability of the fibers to be aligned in the direction of the corresponding eigenvectors.

Viscoelastic fluids can also be described by differential models [4,5]. In these models the evolution of the extra-stress tensor $\underline{\underline{\tau}}$ is given by an advection equation. For example, the upper convected Maxwell's model is given by

$$\lambda \frac{\delta \underline{\underline{\tau}}}{\delta t} + \underline{\underline{\tau}} = 2\mu \underline{\underline{D}}, \quad (2)$$

where $\delta/\delta t$ denotes the upper convected derivative

$$\frac{\delta \underline{\underline{\tau}}}{\delta t} = \frac{d\underline{\underline{\tau}}}{dt} - \text{Grad} \underline{\underline{v}} \underline{\underline{\tau}} - \underline{\underline{\tau}} (\text{Grad} \underline{\underline{v}})^T \quad (3)$$

and λ is a relaxation time.

Another example is the Pom–Pom viscoelastic model, described in [6], where the extra-stress tensor depends on both the molecular orientation $\underline{\underline{s}}$ and the molecular extension λ . The evolutions of both variables are governed by the following advection equations

$$\frac{d\underline{\underline{a}}}{dt} = \underline{\underline{\Omega}}\underline{\underline{a}} - \underline{\underline{a}}\underline{\underline{\Omega}} + \underline{\underline{D}}\underline{\underline{a}} + \underline{\underline{a}}\underline{\underline{D}} - \frac{1}{\tau_b} \left(\underline{\underline{a}} - \frac{I}{3} \right), \quad (4)$$

$$\underline{\underline{s}} = \frac{\underline{\underline{a}}}{\text{Tr}(\underline{\underline{a}})} \quad (5)$$

and

$$\frac{d\lambda}{dt} = \lambda \text{Tr}(\underline{\underline{s}}\underline{\underline{D}}) - \frac{1}{\tau_s} (\lambda - 1), \quad (6)$$

where τ_b and τ_s are two relaxation times.

All the previous equations can be written in the form

$$\frac{d\underline{\underline{a}}}{dt} = \underline{\underline{A}}(\underline{\underline{x}}(t))\underline{\underline{a}} + \underline{\underline{B}}(\underline{\underline{x}}(t)), \quad (7)$$

where $\underline{\underline{a}}$ is the vectorial representation of the unknown fields. In the case of non-linear models standard linearizations lead also to the previous linear equation.

The integration of Eq. (7) in steady non-recirculating flows does not introduce significant difficulties when an accurate numerical technique for treating hyperbolic advection equations is used. Among these techniques, the method of characteristics takes into account that the equation solution at any point depends only on the upstream solution. In spite of its low numerical diffusion and accuracy, this technique remains very time-consuming. Thus, in the original version of the method of characteristics, widely used in the context of non-Newtonian fluid flows simulation [7–13], for each point P where the solution is searched, we

must reconstruct the upstream path until reaching the inflow boundary (where a boundary condition is imposed) at point Q , and from the solution known at point Q the equation can be integrated, using first or higher order finite differences, along the streamline until returning to the departure point P .

To alleviate the computing time required by the original version of the method of characteristics, some semi-Lagrangian techniques have been proposed [14,15] and used in the framework of non-Newtonian flows simulations (see for example [16,17] among many others). In general, these techniques have been successfully applied for solving evolution problems. In these techniques, the solution at point \underline{x} at time $t + \Delta t$ is computed from the solution existing at time t in the position occupied by the considered point at time t . When the solution is computed in a certain number of points (nodes), their positions at previous times do not correspond necessarily with the nodes, and in consequence interpolations (with the associated numerical diffusion effects) are required in order to compute the solution at those points from the solution known at the nodes. Using these techniques, steady solutions are found from the steady state of the evolution problems.

In a former work [18] we have proposed a semi-Lagrangian technique able to compute directly steady solutions of the advection equation governing the fiber orientation in short fibers suspensions flows. This strategy, based on some of the ideas found in [4] and [19], combines an upstream integration along the characteristics associated to each node and a standard interpolation on the mesh skeleton. Thus, this technique can be considered as a standard semi-Lagrangian strategy if different time steps are considered for each node where the solution is searched. The steady solution is then reached after some iterations [18]. The consideration of different time steps is justified if one looks only for the steady solution.

Today, it is usual to find fully Eulerian discretisations of Eq. (7) using for example the discontinuous finite element method, the SUPG technique, the discontinuous finite volume method, . . . (see for example [19,20] and the references therein).

When we consider the integration of this type of equations in a steady recirculating flow, the problem seems a priori ill defined, because neither initial nor boundary conditions are given. As described in [12], some stabilized Eulerian techniques may induce, in some particular cases, a significant numerical diffusion when the mesh is not fine enough. These undesirable effects can be reduced by the use of finer meshes in the regions with high gradients or discontinuities in the solution. Moreover, the standard interpolation of orientation tensors introduce additional difficulties as described in [18]. Thus, the required mesh must sometimes be much finer than the mesh needed to integrate accurately the equation of motion in order to limit the numerical diffusion effects.

In this way, the integration of Eq. (7) using characteristics type methods seems to be simpler in this type of flows. A first possibility is the use of a semi-Lagrangian strategy. However, this strategy cannot avoid the numerical diffusion in the recirculating areas, nor reproduce the solution discontinuity between recirculating and non-recirculating flow regions. Thus, a fully Lagrangian technique seems to be the most accurate numerical procedure, but as previously commented, advection equations in steady recirculating flows seem to define an ill posed problem, because neither initial conditions nor boundary conditions are given. However, in [21], we proved that in fact some linear advection equations modelling non-Newtonian fluid behaviors define a well posed problem, and some results concerning the existence and unicity of steady solutions were given. These solutions can be found imposing the solution periodicity along the closed streamlines of the flow, where the advection equation is integrated by the method of characteristics. The non-linear character of constitutive equations was considered in [22]. In this case, standard linearization strategies were used to impose the solution periodicity of each one of the linearized problems resulting from the iteration procedures.

In this paper we describe the numerical algorithm used to compute steady solutions of linear advection equations (the non-linear case is reduced to the linear one by using standard linearizations) in general steady flows containing recirculating areas. The main particularity of this strategy is to generalize classical characteristic techniques for treating general flows containing recirculating areas.

The numerical strategy described in this work has some particularities: (i) the integration by characteristics is carried out independently in each element (then a parallel computation is possible); (ii) an accurate integration is performed in each element, and an exact expression of the solution can be obtained in some cases as described in [18]; (iii) the element solutions assembling is achieved by means of a fast iteration procedure and (iv) an accurate solution in steady recirculating flows is obtained by imposing the solution periodicity along the closed streamlines. The main contribution of this paper lies in the establishment of an original characteristics algorithm for treating advection equations defined in steady recirculating flows.

Many of the experimental and industrial flows show recirculating areas or recirculate themselves. As an example, in a lid-driven cavity problem, the flow recirculates under the influence of a moving plate. In a contraction or in an expansion flow (as encountered, for example, in extrusion processes), various recirculating areas are observed [23–25]. Actually, many rheometric devices involve this type of flows. Most of these phenomena are associated with a steady state of 2D flows. By this reason, we consider in this paper 2D steady flows which take place, obviously, in 3D geometries. In spite of the fact that the flow is assumed 2D, the advection equation defined in those flows involve variables defined in the 3D space (for example the fiber orientation).

Notation:

- We denote a vector by an underline, e.g. \underline{a} , with components a_i .
- In the same way, the components of a matrix $\underline{\underline{a}}$ are expressed by a_{ij} .

2. Steady solution of linear advection equations in steady non-recirculating flows

We consider the general form of linear advection equations given by Eq. (7), where we assume that vector \underline{a} contains n scalar unknowns, i.e. $\underline{a} \in \mathbb{R}^n$

$$\underline{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}. \quad (8)$$

The matrix $\underline{\underline{A}}$ and the vector \underline{B} may depend on the spatial coordinates (the flow kinematics depends on the spatial coordinates).

The streamlines result from the integration of

$$\frac{d\underline{x}}{dt} = \underline{v}(\underline{x}(t)), \quad (9)$$

which represents the characteristics of Eq. (7).

We describe now the main idea of the numerical algorithm proposed in this paper. For this purpose, we consider two points \mathcal{A} and \mathcal{B} , both located on the same flow streamline. \mathcal{B} is also assumed to be in a downstream position with respect to the position of \mathcal{A} .

We define the \underline{I}^k vector, verifying $I_i^k = \delta_{ik}$ (where δ refers to the Kronecker's function). Since Eq. (7) is linear, its solution can be written as the addition of the general solution \underline{a}_h of the homogeneous differential equation

$$\frac{d\underline{a}_h}{dt} = \underline{\underline{A}}(\underline{x}(t))\underline{a}_h \quad (10)$$

and a particular solution of the non-homogeneous differential equation (Eq. (7)).

As a linear combination of solutions is also a solution of the homogeneous equation, we can integrate by the method of characteristics, from \mathcal{A} to \mathcal{B} , the following n advection equations

$$\frac{d\underline{a}_h^k}{dt} = \underline{A}(\underline{x}(t))\underline{a}_h^k \quad \forall k \in [1, \dots, n] \quad (11)$$

with the initial conditions given by

$$\underline{a}_h^k(t=0) = \underline{a}_h^k(\mathcal{A}) = \begin{pmatrix} a_1^k(\mathcal{A}) \\ a_2^k(\mathcal{A}) \\ \vdots \\ a_{k-1}^k(\mathcal{A}) \\ a_k^k(\mathcal{A}) \\ a_{k+1}^k(\mathcal{A}) \\ \vdots \\ a_n^k(\mathcal{A}) \end{pmatrix} = \underline{I}^k = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \quad (12)$$

The solutions of these equations are noted by \underline{a}_h^k , where the index 'h' indicates that these solutions are related to the homogeneous equation. Now, we can express the general solution \underline{a}_h of the homogeneous equation, from a linear combination, i.e.

$$\underline{a}_h = \sum_{k=1}^{k=n} \alpha_k \underline{a}_h^k, \quad (13)$$

where α_k , $k = [1, \dots, n]$, represent the n scalar unknowns.

To obtain a particular solution \underline{a}_p of Eq. (7), we integrate this equation from an arbitrary initial condition, for example $\underline{a}_p(t=0) = \underline{a}_p(\mathcal{A}) = \underline{0}$.

The general solution of the linear advection equation \underline{a} results

$$\underline{a} = \underline{a}_h + \underline{a}_p = \underline{a}_p + \sum_{k=1}^{k=n} \alpha_k \underline{a}_h^k. \quad (14)$$

Now, if we are looking for the solution at the point \mathcal{B} we can write

$$\underline{a}(\mathcal{B}) = \underline{a}_h(\mathcal{B}) + \underline{a}_p(\mathcal{B}) = \underline{a}_p(\mathcal{B}) + \sum_{k=1}^{k=n} \alpha_k \underline{a}_h^k(\mathcal{B}) = \underline{C}^{\mathcal{A}\mathcal{B}} + \underline{D}^{\mathcal{A}\mathcal{B}} \underline{\alpha}, \quad (15)$$

where $\underline{C}^{\mathcal{A}\mathcal{B}} = \underline{a}_p(\mathcal{B})$, the k th-column of $\underline{D}^{\mathcal{A}\mathcal{B}}$ corresponds to the solution \underline{a}_h^k at the point \mathcal{B} , and the vector $\underline{\alpha}$ is defined by

$$\underline{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}. \quad (16)$$

Obviously, the solution at the point \mathcal{A} is given by

$$\underline{a}(\mathcal{A}) = \underline{a}_h(\mathcal{A}) + \underline{a}_p(\mathcal{A}) = \underline{a}_p(\mathcal{A}) + \sum_{k=1}^{k=n} \alpha_k \underline{a}_h^k(\mathcal{A}) = \underline{0} + \sum_{k=1}^{k=n} \alpha_k \underline{I}^k = \underline{\alpha}. \quad (17)$$

And finally we can write

$$\underline{a}(\mathcal{B}) = \underline{C}^{\mathcal{A}\mathcal{B}} + \underline{D}^{\mathcal{A}\mathcal{B}} \underline{a}(\mathcal{A}). \quad (18)$$

We can notice that, as expected, the solution at the point \mathcal{B} depends only on the solution at the point \mathcal{A} .

As the solution at a point \mathcal{B} only depends on the solution in an arbitrary point upstream \mathcal{A} , every streamline can be divided in several intervals and the solution of Eq. (18) can be found in all these intervals simultaneously and independently.

Thus, we consider m points on a streamline P_i , $i \in [1, \dots, m]$, which define $m - 1$ intervals in non-circulating flows. We assume that these points are ordered in the flow direction, i.e. P_i is located upstream with respect to the position of P_{i+1} , although the explicit knowledge of the connectivity will not be required in the algorithm proposed later in this work.

If we apply Eq. (18) in each interval, we obtain

$$\underline{a}(P_i) = \underline{C}^{P_{i-1}P_i} + \underline{D}^{P_{i-1}P_i} \underline{a}(P_{i-1}) \quad \forall i \in [2, \dots, m], \quad (19)$$

where $\underline{a}(P_1)$ is defined by the inflow boundary condition whose knowledge is required due to the hyperbolic character of the problem.

Eq. (19) can be written in the matrix form

$$\begin{pmatrix} \underline{I} & \underline{0} & \underline{0} & \cdots & \underline{0} & \underline{0} \\ -\underline{D}^{P_1P_2} & \underline{I} & \underline{0} & \cdots & \underline{0} & \underline{0} \\ \underline{0} & -\underline{D}^{P_2P_3} & \underline{I} & \cdots & \underline{0} & \underline{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \underline{0} & \underline{0} & \underline{0} & \cdots & -\underline{D}^{P_{m-1}P_m} & \underline{I} \end{pmatrix} \begin{pmatrix} \underline{a}(P_1) \\ \underline{a}(P_2) \\ \underline{a}(P_3) \\ \vdots \\ \underline{a}(P_m) \end{pmatrix} = \begin{pmatrix} \underline{a}(P_1) \\ \underline{C}^{P_1P_2} \\ \underline{C}^{P_2P_3} \\ \vdots \\ \underline{C}^{P_{m-1}P_m} \end{pmatrix}, \quad (20)$$

where \underline{I} is the unit matrix. In many situations exact expressions of $\underline{C}^{P_{i-1}P_i}$ and $\underline{D}^{P_{i-1}P_i}$ can be obtained [18].

The following relation is also verified

$$\begin{aligned} \underline{a}(P_m) &= \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} \underline{a}(P_{m-1}) = \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} (\underline{C}^{P_{m-2}P_{m-1}} + \underline{D}^{P_{m-2}P_{m-1}} \underline{a}(P_{m-2})) \\ &= \dots = \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} \underline{C}^{P_{m-2}P_{m-1}} + \dots + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_2P_3} \underline{C}^{P_1P_2} + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_1P_2} \underline{a}(P_1) \\ &= \underline{C}^{P_1P_m} + \underline{D}^{P_1P_m} \underline{a}(P_1), \end{aligned} \quad (21)$$

which derives directly from Eq. (19) and where

$$\underline{C}^{P_1P_m} = \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} \underline{C}^{P_{m-2}P_{m-1}} + \dots + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_2P_3} \underline{C}^{P_1P_2} \quad (22)$$

and

$$\underline{D}^{P_1P_m} = \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_1P_2}. \quad (23)$$

This expression reproduces the integration by the method of characteristics along the streamline from the point P_1 until reaching the point P_m .

Eq. (21) will not be used in the numerical algorithm because it requires the knowledge of the points connectivity. However, this expression will be useful in the next section.

2.1. Linear solver

In order to solve Eq. (20) we propose an iterative solver which converges at maximum in $m - 1$ iterations, and which does not require the explicit construction of that linear system of equations.

The iteration process proceeds from the following solution estimation

$$\begin{cases} \underline{\mathbf{a}}^0(P_1) = \underline{\mathbf{a}}(P_1), \\ \underline{\mathbf{a}}^0(P_2) = \underline{\mathbf{0}}, \\ \underline{\mathbf{a}}^0(P_3) = \underline{\mathbf{0}}, \\ \vdots \\ \underline{\mathbf{a}}^0(P_m) = \underline{\mathbf{0}}, \end{cases} \quad (24)$$

where the superscript “0” refers to the first iteration.

From this solution we can compute its first updating from Eq. (19)

$$\underline{\mathbf{a}}^1(P_i) = \underline{\mathbf{C}}^{P_{i-1}P_i} + \underline{\mathbf{D}}^{P_{i-1}P_i} \underline{\mathbf{a}}^0(P_{i-1}) \quad \forall i \in [2, \dots, m] \quad (25)$$

and the j th-updating is given by

$$\underline{\mathbf{a}}^j(P_i) = \underline{\mathbf{C}}^{P_{i-1}P_i} + \underline{\mathbf{D}}^{P_{i-1}P_i} \underline{\mathbf{a}}^{j-1}(P_{i-1}) \quad \forall i \in [2, \dots, m]. \quad (26)$$

In the first iteration, the exact solution is, at least, already computed in the point directly connected with the point located on the inflow boundary. In the iteration i , the exact solution will be necessarily found in the point P_{i+1} , using the exact solution found in the point P_i in the previous iteration. Thus, in $m - 1$ iterations the system will be completely solved. We can notice, that in the case of an ordered list of points in the flow direction, the solution is obtained in only one iteration, i.e. $\underline{\mathbf{a}}^1(P_i) = \underline{\mathbf{a}}(P_i) \quad \forall i \in [2, \dots, m]$.

From an algebraic point of view, this solver corresponds to a particular form of the Jacobi’s method. This strategy can be improved considering the Gauss–Seidel’s solver

$$\underline{\mathbf{a}}^j(P_i) = \underline{\mathbf{C}}^{P_{i-1}P_i} + \underline{\mathbf{D}}^{P_{i-1}P_i} \underline{\mathbf{a}}^j(P_{i-1}) \quad \forall i \in [2, \dots, m]. \quad (27)$$

We can also notice that the Gauss–Seidel solver corresponds to a direct integration by the method of characteristics when the connectivity of the points is known.

2.2. Discussion in the context of a finite element discretisation of the equations of motion

Actually, the algorithm just described does not represent a significant improvement with respect to the numerical techniques based on the method of characteristics, for integrating advection equations in steady non-recirculating flows. However, two particularities can be noticed in the technique just described when it is considered in the context of finite elements discretisations of the equations of motion which define the flow kinematics. Usually, the flow velocity is obtained from the solution of the motion equations by using a mixed formulation of finite elements verifying the LBB conditions. Then, the list of m points previously considered could be defined by the intersections of a particular streamline and the skeleton of the finite element mesh. In this case, the advection equation solution, as previously described, can be computed simultaneously in each element crossed by the streamline. This is a first advantage if one uses parallel computation platforms. The second particularity is related to the fact that the contributions of all elements are automatically taken into account and in many cases the integration in each element can be carried out using exact solution expressions as described in [18].

Furthermore, this technique can be used also for the treatment of steady recirculating flows as we describe in the next section. Thus, the same technique can be applied to treat general flows, where recirculating and non-recirculating flow regions can exist simultaneously.

3. Steady solution of linear advection equations in steady recirculating flows

In the case of steady recirculating flows, the streamlines are closed. In order to solve linear advection equations along closed streamlines we must impose the solution periodicity as explained in [21]. The

technique introduced in the previous section can be extended to obtain, within a short computing-time, an accurate solution of advection equations in steady recirculating flows.

For this purpose, we consider that a particular closed streamline intersects the finite elements mesh skeleton in $m - 1$ points. At first, we assume that the list of points is ordered. Later on, a more general numerical algorithm, which does not need ordered points, will be proposed.

In the present case we have $m - 1$ intervals associated with the $m - 1$ points. If we proceed in each element as explained in the previous section, we have

$$\underline{a}(P_i) = \underline{C}^{P_{i-1}P_i} + \underline{D}^{P_{i-1}P_i} \underline{a}(P_{i-1}) \quad \forall i \in [2, \dots, m] \quad (28)$$

with $P_m \equiv P_1$, implying $\underline{a}(P_m) = \underline{a}(P_1)$, which describes the periodicity condition. Now, we introduce the periodicity in Eq. (21)

$$\begin{aligned} \underline{a}(P_1) &= \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} \underline{C}^{P_{m-2}P_{m-1}} + \dots + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_2P_3} \underline{C}^{P_1P_2} + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_1P_2} \underline{a}(P_1) \\ &= \underline{C}^{P_1P_m} + \underline{D}^{P_1P_m} \underline{a}(P_1). \end{aligned} \quad (29)$$

This equation can be rewritten in the form

$$(\underline{I} - \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_1P_2}) \underline{a}(P_1) = \underline{C}^{P_{m-1}P_m} + \underline{D}^{P_{m-1}P_m} \underline{C}^{P_{m-2}P_{m-1}} + \dots + \underline{D}^{P_{m-1}P_m} \dots \underline{D}^{P_2P_3} \underline{C}^{P_1P_2}, \quad (30)$$

which allows us to compute the solution at the point P_1 verifying the periodicity constraint.

To find the solution using the Eq. (30) directly, the connectivity information is required. To avoid the explicit construction of the connectivity table, an iterative algorithm is proposed.

Thus, in that case, we can compute the solution related to an arbitrary choice of the solution at P_1

$$\underline{a}^0(P_1) = \underline{a}^\star. \quad (31)$$

The solution at the points P_2, \dots, P_m is obtained at maximum in $m - 1$ iterations, by using the iteration algorithm previously described

$$\underline{a}^j(P_i) = \underline{C}^{P_{i-1}P_i} + \underline{D}^{P_{i-1}P_i} \underline{a}^j(P_{i-1}) \quad \forall i \in [2, \dots, m]. \quad (32)$$

Obviously, the solution obtained at the point P_m is different from the initial one \underline{a}^\star arbitrarily chosen. We can define the residual associated with the initial estimation $\underline{\mathcal{R}}(\underline{a}(P_1))$

$$\underline{\mathcal{R}}(\underline{a}(P_1)) = \underline{a}(P_m) - \underline{a}(P_1) = \underline{a}(P_m) - \underline{a}^\star. \quad (33)$$

Now a Newton's technique is used to update $\underline{a}(P_1)$ in order to obtain a null residual

$$\underline{0} = \underline{\mathcal{R}}(\underline{a}(P_1) + \underline{\Delta a}(P_1)) = \underline{\mathcal{R}}(\underline{a}(P_1)) + \frac{\partial \underline{\mathcal{R}}}{\partial \underline{a}(P_1)} \underline{\Delta a}(P_1). \quad (34)$$

The linearity of the correction defined by Eq. (34) results from the linear dependence of the residual on $\underline{a}(P_1)$. To prove this linear dependence we substitute in the residual definition (Eq. (33)) the relation between the solutions at P_m and P_1 (Eq. (21))

$$\underline{\mathcal{R}}(\underline{a}(P_1)) = \underline{C}^{P_1P_m} + \underline{D}^{P_1P_m} \underline{a}(P_1) - \underline{a}(P_1), \quad (35)$$

from which the tangent matrix results

$$\underline{K}_\Gamma = \frac{\partial \underline{\mathcal{R}}}{\partial \underline{a}(P_1)} = \underline{D}^{P_1P_m} - \underline{I}. \quad (36)$$

Thus, the solution correction at the point P_1 results from Eqs. (33), (34) and (36)

$$\underline{\Delta a}(P_1) = -\left(\frac{\partial \mathcal{R}}{\partial \underline{a}(P_1)}\right)^{-1} \mathcal{R}(\underline{a}(P_1)) = -(\underline{D}^{P_1 P_m} - \underline{I})^{-1}(\underline{a}(P_m) - \underline{a}^*). \quad (37)$$

However, if the list of points is not ordered, the expression of $\underline{D}^{P_1 P_m}$ cannot be obtained. To avoid this difficulty, we can obtain the tangent matrix numerically in the following way: we compute, using the integration algorithm previously described, which converges at maximum in $m - 1$ iterations, the n solutions at P_m associated with the following n initial values

$$\underline{a}(P_1) = \underline{a}^* + \epsilon \underline{I}^j \quad \forall j \in [1, \dots, n], \quad (38)$$

where ϵ is a parameter small enough. The n solutions at P_m are expressed by $\underline{a}^j(P_m)$, whereas $\underline{a}(P_m)$ refers to the solution related to $\underline{a}(P_1) = \underline{a}^*$.

It is easy to verify that the j th column of \underline{K}_T contains

$$\frac{\{\underline{a}^j(P_m) - (\underline{a}^* + \epsilon \underline{I}^j)\} - \{\underline{a}(P_m) - \underline{a}^*\}}{\epsilon}. \quad (39)$$

Thus, the solution of $n + 1$ linear problems (related to the solutions $\underline{a}(P_m)$ and $\underline{a}^j(P_m)$, $j \in [1, \dots, n]$) allows us to compute the steady solution of linear advection equations defined along the closed streamlines in the recirculating flows, verifying the periodicity condition. n represents the fluid model variables, that rarely exceeds tens.

4. General 2D flows: localization of steady recirculating flow areas

We will prove in this section that as soon as the stream function is known, steady recirculating flow areas can be easily located, as described below, in 2D flows. If a vorticity-stream function formulation is applied to define the flow kinematics, then the stream function is directly obtained. However, if a velocity–pressure mixed formulation is used, then the stream function must be computed from the velocity field, taking into account that in 2D incompressible flows the velocity derives from the stream function ψ

$$\begin{cases} u = \frac{\partial \psi}{\partial y}, \\ v = -\frac{\partial \psi}{\partial x}, \end{cases} \quad (40)$$

where u and v are the velocity vector components. A direct consequence of Eq. (40) is

$$\Delta \psi = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} = \text{curl } \underline{v} \cdot \underline{e}_z, \quad (41)$$

where \underline{e}_z is the unit vector defining the out-plane direction. Thus, if the velocity field is known, the velocity derivatives can be computed, and the solution of Eq. (41) leads to the stream function solution. The solution of Eq. (41) can be computed easily using a standard finite element Galerkin formulation. Appropriate boundary conditions must be imposed in order to verify the prescribed velocities.

For irrotational flows $\text{curl } \underline{v} = \underline{0}$, and the stream function verifies the Laplace's equation

$$\Delta \psi = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0 \quad (42)$$

and applying the principle of the maximum, the highest and lowest values of ψ are located on the flow domain boundary. In consequence, each value of ψ found inside the fluid domain, corresponds with a value

of ψ found on the inflow and outflow boundaries. Thus, each point inside the flow domain is on a streamline which goes from the inflow boundary to the flow exit.

However, when $\text{curl } \underline{v} \neq 0$ the principle of maximum cannot be applied and the highest and lowest stream function values can be located inside the flow domain, in whose neighborhood we find values of the stream function which are not found on the inflow boundary. Then, the points with a value of the stream function which does not correspond with the values existing on the inflow boundary are located on closed streamlines. In this way, the steady recirculating flow areas can be identified easily.

5. Numerical tests

In this section, two examples concerning very simple flows and advection equations will be analyzed in order to conclude about the accuracy of the numerical strategy proposed in the previous sections. The first one concerns a non-recirculating flow whereas the second one is a recirculating flow. For both problems the exact solutions are known which allows us to quantify the numerical error and the accuracy. In the next section a short fiber suspension flow in a 4:1 planar contraction containing a steady recirculating area, is analyzed.

5.1. A rigid body uniform motion

The first flow is defined in the domain $[0, 1] \times [0, 0.5]$ by the following velocity field

$$\begin{cases} u = 1, \\ v = 0, \end{cases} \quad (43)$$

where the following scalar advection equation is considered

$$\frac{da}{dt} = -Aa + B, \quad (44)$$

A and B are assumed constants, $A = B = 1$, and the condition $a(x = 0, y) = 0$ is imposed on the inflow boundary $x = 0$.

The homogeneous equation is

$$\frac{da_h}{dt} = -Aa_h, \quad (45)$$

whose general solution a_h is

$$a_h = \alpha e^{-At}. \quad (46)$$

The particular solution a_p of Eq. (44) associated with a null initial condition, i.e. $a_p(t = 0) = 0$, is

$$a_p = \frac{B}{A}(1 - e^{-At}). \quad (47)$$

Thus, the general solution of Eq. (44) is

$$a = a_h + a_p = \alpha e^{-At} + \frac{B}{A}(1 - e^{-At}), \quad (48)$$

where α results to be the initial condition imposed on the inflow boundary.

A first finite element mesh containing 470 elements (see Fig. 1) is considered in order to compute the stream function associated with the velocity field defined by Eq. (43). In this case the stream function is $\psi = y$, and the streamlines are defined by the family of lines $y = cte \in [0, 0.5]$.

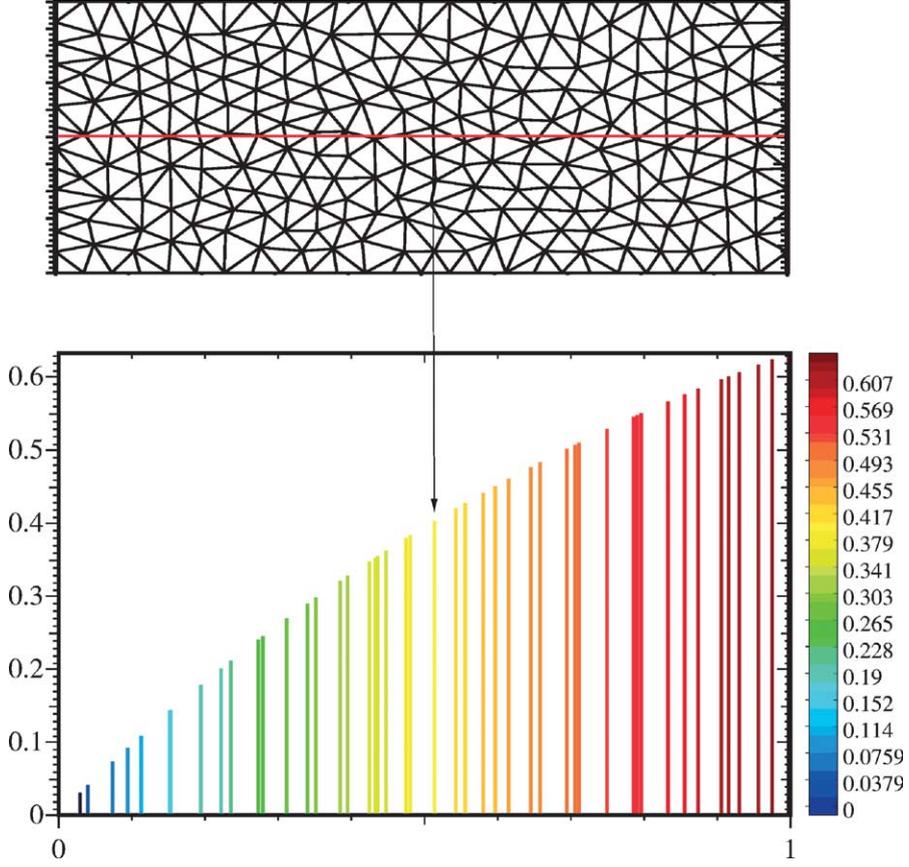


Fig. 1. Solution of the equation on the streamline $y = 0.25$.

If we consider two points \mathcal{P} and \mathcal{Q} on the streamline $y = k$, whose coordinates are (x_p, k) and (x_Q, k) ($x_Q > x_p$) respectively, the time t_{PQ} taken by an imaginary particle to cover the distance between these two points, taking into account the uniform velocity field considered, is given by

$$t_{PQ} = \frac{x_Q - x_p}{u}. \quad (49)$$

Thus, from Eq. (48) we can write

$$a_Q = a_p e^{-At_{PQ}} + \frac{B}{A}(1 - e^{-At_{PQ}}) \quad (50)$$

or using the notation introduced in the previous sections

$$a_Q = C^{PQ} + D^{PQ} a_p, \quad (51)$$

where $C^{PQ} = \frac{B}{A}(1 - e^{-At_{PQ}})$ and $D^{PQ} = e^{-At_{PQ}}$.

Now, we consider the streamline related to $\psi = 0.25$, which corresponds to the line $y = 0.25$. This line intersects the finite element mesh, shown in Fig. 1, in 48 points ($m = 48$). If we look for the solution using the numerical strategy proposed in this paper, we obtain the solution represented in Fig. 1. In this figure, we represent the solution of the equation in each intersection point between the mesh skeleton and the

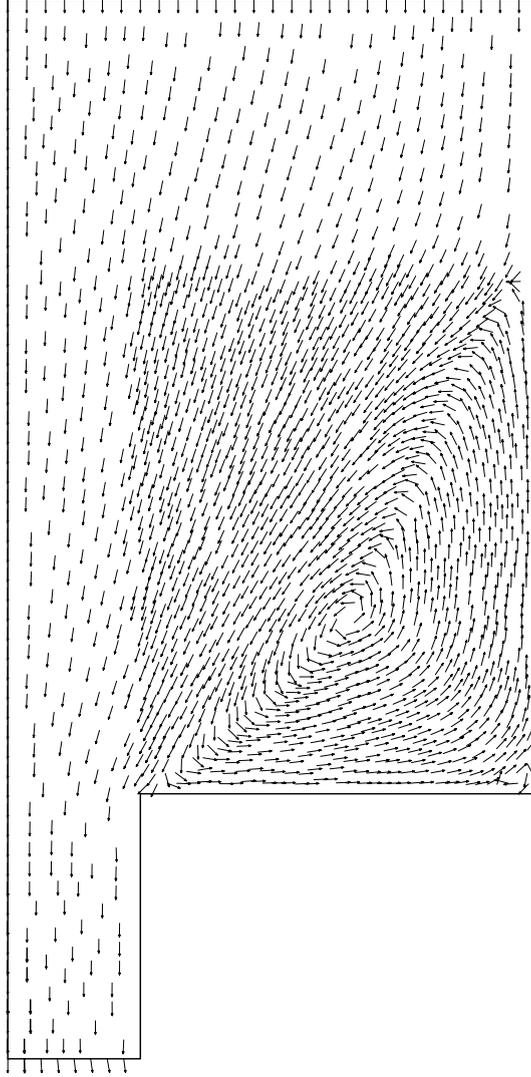


Fig. 2. Contraction problem: normalized velocity of the fluid.

considered streamline ($\psi = 0.25$). Obviously, as we have used the exact expressions of $C^{P_{i-1}P_i}$ and $D^{P_{i-1}P_i}$, the solution obtained coincides with the exact solution of Eq. (44):

$$a = \frac{B}{A}(1 - e^{-At}) \quad (52)$$

with $t = x/u$.

If $C^{P_{i-1}P_i}$ and $D^{P_{i-1}P_i}$ are computed numerically using the method of characteristics, some numerical errors are introduced, and the accuracy depends on the order of the finite difference formula considered in the integration along the characteristics (a fourth-order Runge–Kutta technique with step control to guarantee the numerical stability is used in the numerical applications presented in this work).

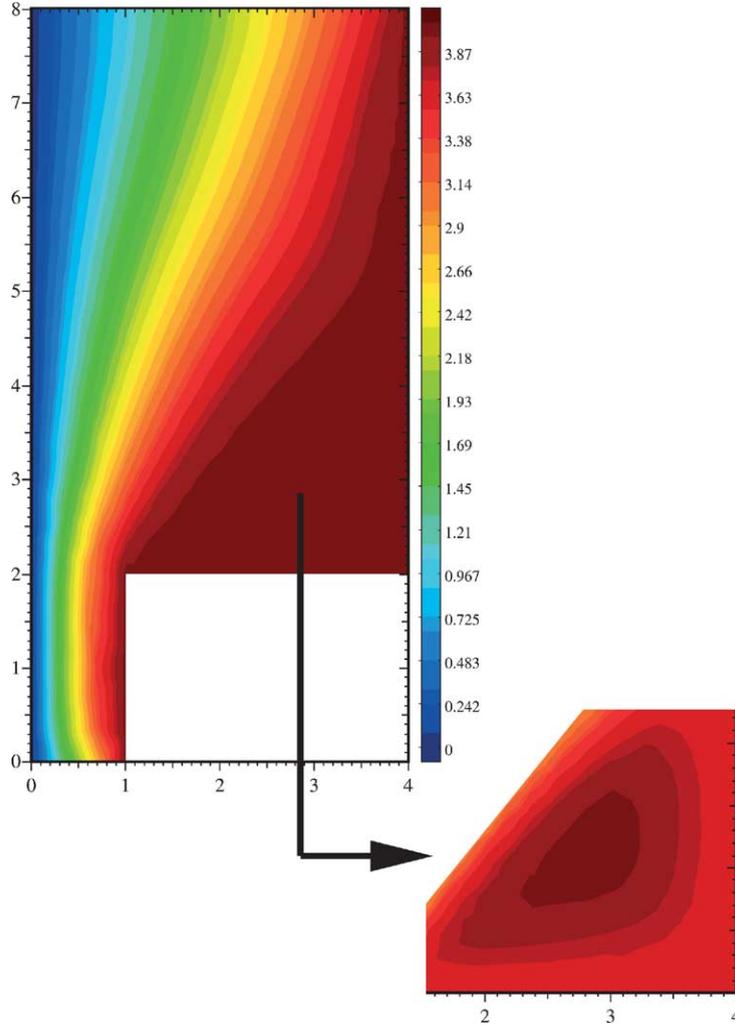


Fig. 3. Contraction problem: reconstructed stream function.

5.2. A rigid body rotation motion

In this case the same scalar advection equation is considered, Eq. (44), but now it is defined in the re-circulating flow defined by

$$\begin{cases} u = -(y - 0.5), \\ v = (x - 0.5) \end{cases} \quad (53)$$

with $x, y \in [0, 1]$.

The stream function related to this velocity field is $\psi = -\frac{1}{2}((x - 0.5)^2 + (y - 0.5)^2)$. Thus, the streamlines are the family of circles defined by $\psi = cte \in [-0.25, 0]$. For $|\psi| < \frac{0.25}{2}$ the streamlines do not intersect the domain boundary. Thus, these streamlines are closed and the only condition available to

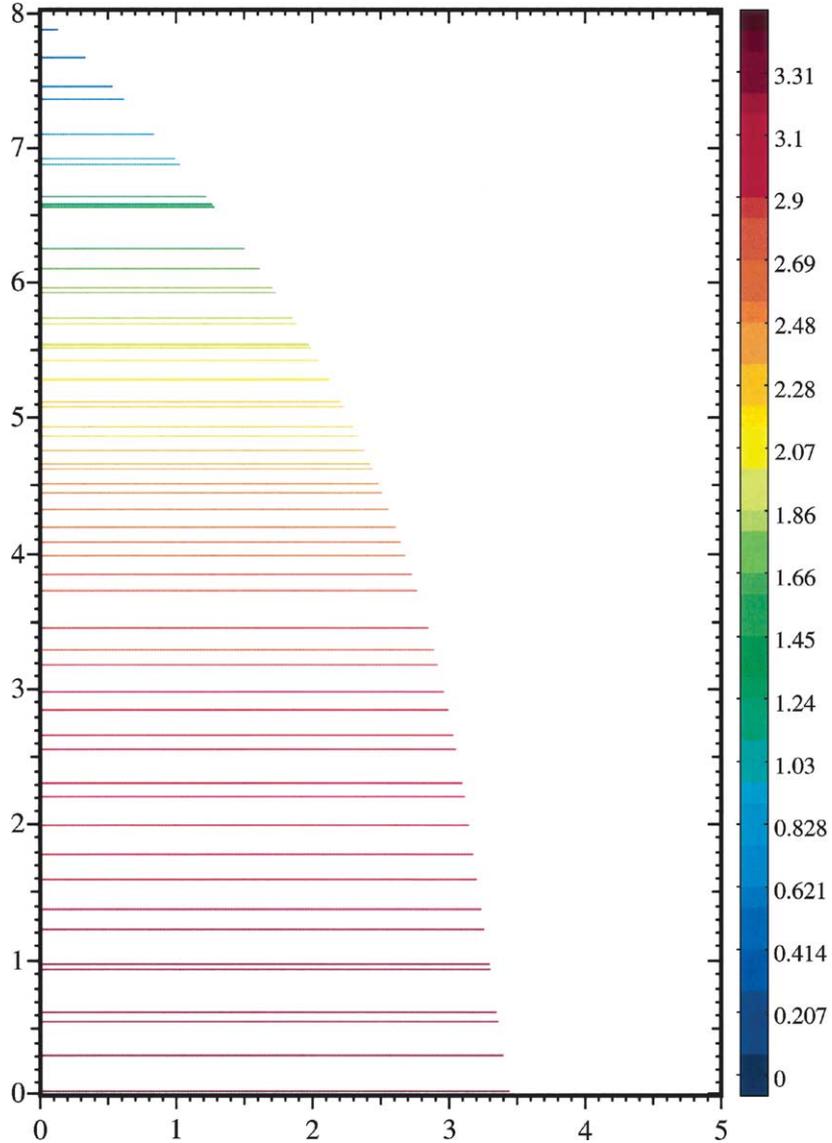


Fig. 4. Solution of the equation along the streamline $\psi = 2$ for $A = 0.1$ and $B = 1$.

their integration is the periodicity of the solution, imposed by the steady flow. For $|\psi| > \frac{0.25}{2}$ the streamlines intersect the flow domain boundaries, and the integration can be carried out as in the previous example.

Now, if we apply the strategy proposed previously for steady recirculating flows, the solution $a = 1$ is obtained at each intersection point between the closed streamline and the mesh skeleton. This solution corresponds to the steady solution of Eq. (44)

$$a(t \rightarrow \infty) = \frac{B}{A} = 1. \quad (54)$$

6. A rheological application: short fibers suspension flow in a 4:1 planar contraction

Fig. 2 represents the normalized velocity field $\left(\frac{\underline{v}}{\|\underline{v}\|}\right)$ in a symmetric 4:1 planar contraction involving a short fibers suspension flow (see [12] for more details). The inflow velocity is $\underline{v}^T(x, y = 8) = (0, -1)$ and a null traction condition is assumed on the outflow boundary $y = 0$. We can notice that the numerical simulation predicts a steady recirculating area in the contraction corner, in good agreement with the experimental visualizations [26]. Fig. 3 depicts the stream function computed from the solution of Eq. (41). We can notice that the values of the stream function which do not belong to the inflow boundary, $\psi > 4$, identify precisely the recirculating flow area.

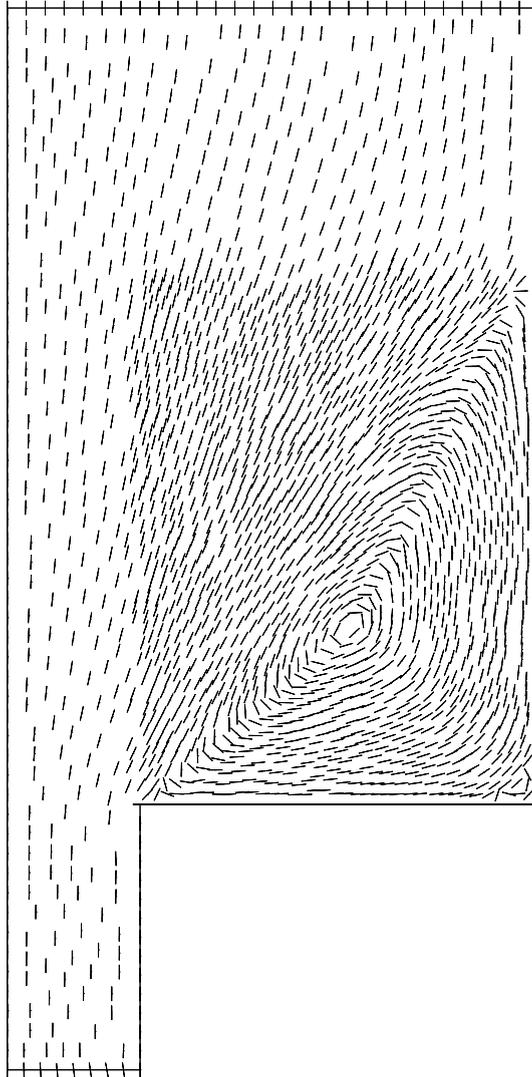


Fig. 5. Fiber orientation prediction obtained using the generalized characteristics technique.

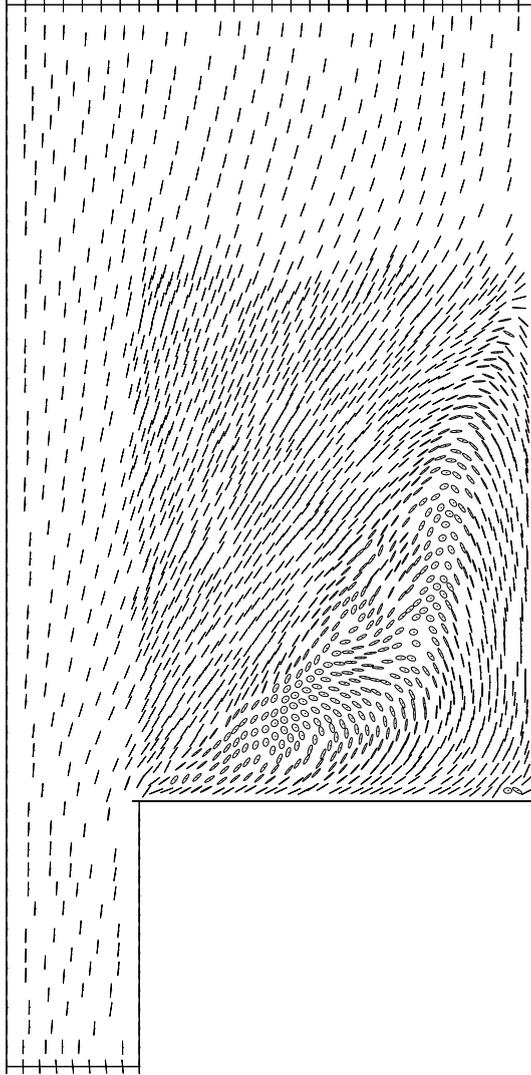


Fig. 6. Fiber orientation prediction obtained using the stabilized finite element method.

Firstly, we consider the advection equation (44) with the initial condition given by $a(x, y = 8) = 0$. In Fig. 4, the solution of Eq. (44) for $A = 0.1$ and $B = 1$ is depicted at the intersection points between the mesh skeleton and the streamline defined by $\psi = 2.0$. In the recirculating area, the solution $a = 10$ is found, which obviously corresponds with the steady solution.

Now, we consider a more practical application involving a short fibers suspension flow in a 4:1 contraction, whose kinematics is depicted in Fig. 2. We assume that the fibers are locally aligned in the flow direction on the inflow boundary, i.e.

$$\underline{a}(x, y = 8) = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \quad (55)$$

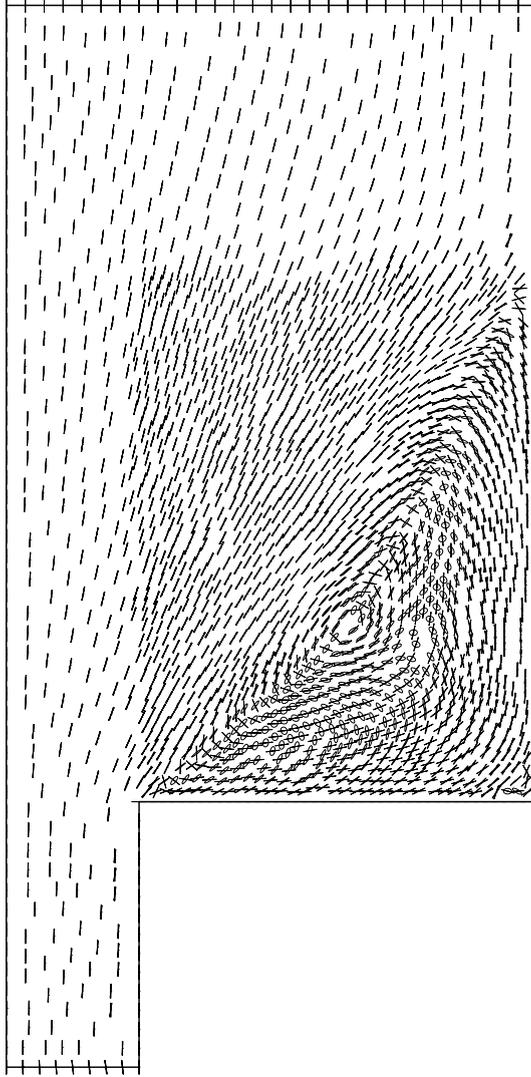


Fig. 7. Fiber orientation prediction obtained using the stabilized finite element method versus the exact solution (local alignment of the fibers with the flow).

and we compute the solution of the advection equation governing the fiber orientation (Eq. (1) with $D_r = 0$ and $k = 1$). In this case the orientation solution results to be the local alignment of the fibers with the flow as proved in [12]. Fig. 5 depicts the solution obtained at each node of the mesh using the strategy proposed in the previous sections. At each node the eigenvector associated with the highest eigenvalue (which is approximately 1 in every point) is depicted. The numerical orientation prediction is in perfect agreement with the exact solution which predicts a local alignment of the fibers on the flow direction. Now, if we compute the fiber orientation using the SUPG finite element method [27], then the solution obtained (see Fig. 6) shows significant deviations with respect to the exact orientation, mainly in the recirculating flow area. In this figure, we represent the orientation tensor by means of an ellipse. The ellipse axis directions correspond to the eigenvectors of the orientation tensor and the semi-axis sizes correspond to the

eigenvalues. The ellipse thickness indicates a fiber misalignment with respect to the most probable orientation direction. Fig. 7 depicts the superposition of Figs. 5 and 6, from which significant deviations in the prediction of the most probable fiber orientation direction using the SUPG technique are noticed.

Even if the CPU time required by the characteristics type strategies is higher than the computation time required by stabilized Eulerian techniques, the accuracy of the first ones is in general better if we proceed with a coarse mesh. The only possibility to improve the accuracy of Eulerian techniques is based on local remeshing, which requires appropriate remeshing criteria. From our experience, if one want to compute an accurate enough solution of the fiber orientation, very fine meshes are needed, and in that case the computing time becomes similar to the one required by a Lagrangian simulation.

In spite of these results, this work does not attempt to compare characteristics and stabilized finite element strategies for solving this kind of problems, for which a deeper analysis will be required.

Remark. In order to couple the flow kinematics resolution with the solution of the advection equation governing the fiber orientation evolution (coupled models), a fixed point strategy is used. Thus, we proceed until convergence, solving the advection equation from the flow kinematics computed at the previous iteration, updating the flow kinematics from the fiber orientation just computed. In the resolution of the flow kinematics, the fiber orientation must be evaluated at the integration points used to evaluate the variational formulation of the equations of motion. A first possibility is to compute directly the fiber orientation at these points, solving the advection equation governing its evolution along the streamlines passing through those points [9]. Other possibility is the evaluation of the fiber orientation at the mesh nodes, using the same strategy, from which the orientation at the integration points can be interpolated [7,8,10,12]. A last possibility is based on the computation of the solution along a certain number of characteristics, from which the solution can be interpolated anywhere [11,13]. The strategy described in this paper can be applied to compute the solution of the orientation equation at the nodes, at the integration points or along certain characteristics, in general flows containing recirculating zones. In any case, the aspects related to the flow kinematics resolution or the coupling between the flow kinematics and the fiber orientation are not the purpose of the present paper. However, we would like to notice that the use of streamline integrations coupled with fully Eulerian discretisation of the equations of motion is the base of the micro–macro modelling recently proposed by the Keunings’s team [28–30].

7. Conclusions

In this paper a general characteristics algorithm for computing steady solutions of advection equations in general steady flows has been described and tested. The following particularities can be pointed out: (i) the integration by characteristics is carried out independently in each element (then a parallel computation is possible); (ii) an accurate integration is performed in each element, and an exact expression of the solution can be used in particular cases; (iii) the element solutions assembly is achieved by means of a fast iteration procedure and (iv) an accurate solution in steady recirculating flows is obtained by imposing the solution periodicity along the closed streamlines.

This technique can be applied for treating non-linear models (after linearization) and it allows to compute accurate solutions of advection equations in general flows in a reasonable computing time.

References

- [1] G.L. Hand, A theory of anisotropic fluids, *J. Fluid Mech.* 13 (1962) 33–46.
- [2] S.G. Advani, Ch.L. Tucker III, Closure approximations for three-dimensional structure tensors, *J. Rheol.* 34 (1990) 367–386.
- [3] C.L. Tucker III, Flow regimes for fiber suspensions in narrow gap, *J. Non-Newtonian Fluid Mech.* 39 (1991) 239–268.

- [4] R. Keunings, Viscoelastic flows, in: S.G. Advani (Ed.), *Flow and Rheology in Polymer Composites Manufacturing*, Elsevier, 1994.
- [5] F.P.T. Baaijens, *Experimental and Numerical Analysis of Viscoelastic Flows*, PPS-15, Polymer Processing Society, Netherlands, 1999.
- [6] T.C.B. McLeish, R.G. Larson, Molecular constitutive equations for a class of branched polymers: the Pom-Pom polymer, *J. Rheol.* 42 (1) (1998) 81–110.
- [7] C. Givler, M.J. Crochet, R.B. Pipes, *Numerically Predicted Fiber Orientation in Dilute Suspensions*, NUMIFORM, Pineridge Press, 1982, pp. 559–575.
- [8] J. Rosenberg, M. Den, R. Keunings, Simulation of non-recirculating flows of dilute fiber suspensions, *J. Non-Newtonian Fluid Mech.* 37 (1990) 317–345.
- [9] M.C. Altan, S.I. Güçeri, R.B. Pipes, Anisotropic channel flow of fiber suspensions, *J. Non-Newtonian Fluid Mech.* 42 (1992) 65–83.
- [10] G. Ausias, J.F. Agassant, M. Vincent, Flow and fiber orientation calculations in reinforced thermoplastic extruded tubes, *Int. Polym. Process.* IX (1994) 51–59.
- [11] K. Chiba, K. Nakamura, Numerical solution of fiber suspensions flow through a complex channel, *J. Non-Newtonian Fluid Mech.* 78 (1998) 167–185.
- [12] A. Poitou, F. Chinesta, R. Torres, Numerical simulation of the steady recirculating flows of fibers suspensions, *J. Non-Newtonian Fluid Mech.* 90 (2000) 65–80.
- [13] K. Chiba, K. Yasuda, K. Nakamura, Numerical solution of fiber suspensions flow through a parallel channel by coupling flow fields with fiber orientation distribution, *J. Non-Newtonian Fluid Mech.* 99 (2001) 145–157.
- [14] J.P. Benqué, J. Ronat, Quelques Difficultés des Modèles Numériques en Hydraulique, in: R. Glowinski, J.L. Lions (Eds.), *Computer Methods in Applied Mechanics and Engineering*, North Holland, 1982, pp. 471–494.
- [15] O. Pironneau, On the transport-diffusion algorithm and its application to the Navier–Stokes equation, *Numer. Math.* 38 (1982) 309–332.
- [16] M. Fortin, D. Esselaoui, A finite element procedure for viscoelastic flows, *Int. J. Numer. Methods Fluids* 7 (1987) 1035–1052.
- [17] R.M. Phillips, T.N. Phillips, Flow past a cylinder using a semi-Lagrangian spectral element method, *Appl. Numer. Math.* 33 (2000) 251–257.
- [18] F. Chinesta, A. Poitou, R. Torres, A semi-Lagrangian strategy to predict the fiber orientation in the steady flows of reinforced thermoplastics, *Comput. Methods Appl. Mech. Eng.* 189 (2000) 233–247.
- [19] O. Pironneau, *Finite Element Methods for Fluids*, John Wiley & Sons, Masson, 1989.
- [20] R.E. Erwing, H. Wang, A summary of numerical methods for time-dependent advection-dominated partial differential equations, *J. Comput. Appl. Math.* 128 (2001) 423–445.
- [21] F. Chinesta, G. Chaidron, On the steady solution of linear advection problems in steady recirculating flows, *J. Non-Newtonian Fluid Mech.* 98 (2001) 65–80.
- [22] G. Chaidron, F. Chinesta, On the steady solution of non-linear advection equations in steady recirculating flows, *Comput. Methods Appl. Mech. Eng.* 191 (2002) 1159–1172.
- [23] K. Walters, Overview of macroscopic viscoelastic flows, *Viscoelast. Rheol.* (1985) 47–79.
- [24] R.E. Evans, K. Walters, Further remarks on the lip-vortex mechanism of vortex enhancement in planar-contraction flows, *J. Non-Newtonian Fluid Mech.* 32 (1989) 95–105.
- [25] P. Townsend, K. Walters, Expansion flows of non-Newtonian liquids, *Chem. Eng. Sci.* 49 (5) (1993) 749–763.
- [26] G.G. Lipscomb, M.M. Denn, D.U. Hur, D.V. Boger, The flow of fiber suspensions in complex geometries, *J. Non-Newtonian Fluid Mech.* 26 (1988) 297–325.
- [27] F. Chinesta, A. Poitou, R. Torres, Numerical prediction of the fiber orientation in steady flows, *Rev. Eur. Eléments Finis* 8 (4) (1999) 355–374.
- [28] P. Halin, G. Lielens, R. Keunings, V. Legat, The Lagrangian particle method for macroscopic and micro–macro viscoelastic flow computations, *J. Non-Newtonian Fluid Mech.* 79 (1998) 387–403.
- [29] X. Gallez, P. Halin, G. Lielens, R. Keunings, V. Legat, The adaptative Lagrangian particle method for macroscopic and micro–macro computations of time-dependent viscoelastic flows, *Comput. Methods Appl. Mech. Eng.* 180 (1999) 345–364.
- [30] P. Wapperom, R. Keunings, V. Legat, The backward-tracking Lagrangian particle method for transient viscoelastic flows, computations, *J. Non-Newtonian Fluid Mech.* 81 (2–3) (2000) 273–295.