Preprint of the paper

"On the resolution of the viscous incompressible flow for various SUPG finite element formulations"

P. Vellando, J. Puertas, J. Bonillo, J. Fe (2000)

CD Proceedings of the ECCOMAS 2000 conference, Barcelona 11-14 September 2000 (ISBN 84-89925-70-40).

European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS 2000
Barcelona, 11-14 September 2000

ON THE RESOLUTION OF THE VISCOUS INCOMPRESSIBLE FLOW FOR VARIOUS SUPG FINITE ELEMENT FORMULATIONS

*P. Vellando, **J. Puertas, **J. Bonillo, *J. Fe.
ETS. de Ingenieros de Caminos, Canales y Puertos.
*Dpto. de Métodos Matemáticos y de Representación.
**Dpto. de Tecnología de la Construcción.
Universidad de La Coruña. Campus de Elviña. 15192 La Coruña, Spain.
Tel: (34) 981 167000, fax: (34) 981 167170. e-mail: vellando@iccp.udc.es

Key words: Viscous incompressible flow, FEM, Navier Stokes, SUPG.

Abstract. A Finite Element based program has been released to solve the steady 2D Navier-Stokes equations. The mixed-variable algorithm is used as a first approach to solve the differential problem. In order to reduce the number of equations, both a penalty and segregated formulation are implemented to give solution to the viscous incompressible flow and their results are compared with the former formulation. The program makes use of a SUPG type algorithm as a stabilisation procedure, in order to eliminate the numerical oscillations, which may appear when the boundary conditions force a sudden change in the solution, without necessarily refining the mesh. The three different algorithms are checked making use of the benchmark problems of the flow in a square cavity, the backward step and the flow past a cylinder. Finally the code is used to solve the flow in some practical problems and their results are commented.

1 INTRODUCTION

The finite element method was developed to solve problems of the rigid body, fluids instead, are of an 'odd' nature compared to solids and the FEM cannot be applied so straightforward. In the FE resolution of the structural analysis problems the position of the particles is not considered to be a function of time. The solution of the rigid body problem is stated in terms of the nodes displacements and instead in fluids, we have a mixed formulation in which both velocity and pressure are needed in order to solve the flow. The resulting governing equation in fluids is a non-linear differential system and moreover, the so-called 'stiffness' matrix associated to this system, inherits from structural analysis its nomenclature but not its symmetry.

For many of the linear elasticity problems solved using the FEM, the mere requirement of the ownership of the functions to a certain set of mathematical spaces ensures the obtaining of a stable and meaningful approximation, but this is not again so straightforward in fluids. This aspect would require a further discussion that is beyond the scope of this paper. Anyway in what follows, the basic finite element functions used will lead to meaningful solutions even for basic elements that do not verify strictly the LBB condition, the main theoretical restriction when referring to viscous flow problems.

The viscous incompressible flow is governed by the Navier-Stokes equations. Although some other simplified formulae can be adopted to solve flows were the frictional forces can be neglected (potential flow) or 'creeping' flows (stokes simplification) for instance, the Navier-Stokes equations are the only ones to solve the general problem. The unknowns for the N-S equations are both velocity and pressure and the most intuitive way of solving the problem is simply to transform the constitutive equation into a system of equations in which the unknowns were both variables. This straightforward procedure although intuitive and simple, is however quite expensive computationally speaking. In order to improve the memory requirements, a penalty and segregated algorithm will be used, and their results compared for direct and iterative solvers, which have been implemented on them. The SUPG (Streamline Upwinding/Petrov-Galerkin) method has been used as a stabilisation procedure so as to being able to solve the flow without using very refined meshes. Once the algorithm has been checked on some benchmark problems with good results, the program is used to solve some practical problems in the field of the wastewater treatment.

2 GOVERNING EQUATIONS

The Navier-Stokes equations, can be written as:

$$u_{i,t} + u_j u_{i,j} = -\frac{1}{\rho} p_{,i} + \nu u_{i,jj} + f_i$$
$$u_{i,i} = 0 \quad \text{in } \Omega \qquad (1)$$

together with the initial and boundary conditions:

$$u_i\rfloor_{\Gamma_1} = b_i \quad \text{(Dirichlet)} \qquad \sigma_{ij} n_j\rfloor_{\Gamma_2} = t_i \quad \text{(Newman)}$$
$$u_i(x_j, 0) = u_{i0}(x_j) \qquad \text{with} \quad u_{i0,i} = 0 \tag{2}$$

where u is velocity, p is pressure, f is the body force per unit mass, ρ is density, ν is the cinematic viscosity, Γ_1 and Γ_2 are two non overlapping subsets of the piecewise smooth domain boundary Γ, b_i is the velocity vector prescribed in Γ_1, t_i is the traction vector prescribed on Γ_2, n_j is the outward unit vector normal to Γ_2, and the derivatives are made with respect to indices after commas.

3 FINITE ELEMENT FORMULATION

The weighted residuals method is applied in order to transform our differential problem into an integral equation over the domain Ω. The differential equation is multiplied by a set of weighting functions w_i, q, and integrated over the domain Ω:

$$\int_\Omega w_i\left(u_{i,t} + u_j u_{i,j} + \frac{1}{\rho} p_{,i} - \nu u_{i,jj} - f_i\right) d\Omega = 0 \qquad \int_\Omega q u_{i,i} d\Omega = 0 \tag{3}$$

So as to introduce a Galerkin type FE formulation we are going to define some function spaces in which our variables will be included. Let $L^2(\Omega)$ be the space of functions that are square integrable over the domain Ω, and the Sovolev space $H^k(\Omega)$, the subspace of $L^2(\Omega)$, in which the derivatives of order up to k belong also to the space $L^2(\Omega)$. The subspace $L_0^2(\Omega)$ is defined as the subspace of $L^2(\Omega)$ with the constraint of having a zero mean over the domain Ω; this subspace could be used in connection with the pressure unknown or be replaced by the constraint of fixing the pressure at a point. The subspace H_0^1 is formed by functions that belong to H^1 and vanish on the boundary Γ.

Next we are going to apply the gauss theorem to find out the weak version of the former equations, so as to reduce the order of the derivatives involved and together with it, the derivability requirements of the functions. Our problem is therefore reduced to that of finding $u_i, p \in H^1$, such that:

$$\int_\Omega w_i\left(u_{i,t} + u_j u_{i,j} - f_i\right) + \nu\int_\Omega w_{i,j} u_{i,j} d\Omega - \frac{1}{\rho}\int_\Omega w_{i,i} p d\Omega - \int_{\Gamma_2} t_i w_i d\Gamma_2 = 0 \quad \int_\Omega q u_{i,i} d\Omega = 0$$

$$\forall w_i \in H^1 \quad \forall q \in H^1, \quad \text{with} \quad w_i\rfloor_{\Gamma_1} = 0 \quad u_i\rfloor_{\Gamma_1} = b_i \quad u_i(x_j, 0) = u_{i0}(x_j) \tag{4}$$

Next step in the resolution of our partial differential equation by the FEM will be the splitting of our arbitrarily shaped domain Ω, into a set of basic elements approximating the shape of Ω. The solution to be obtained afterwards will be an approximation to the exact solution of the differential equation that we are not able to solve analytically and which will be given only in a few points of the domain, specifically in the vertices of the basic elements. We are going to obtain our approximate solution once we have determined u_i^h, p^h belonging to some subspaces $V_0^h \in H_0^1(\Omega)$ and $S_0^h \in L_0^2(\Omega)$, where h is a parameter related to the size of the grid in which the domain Ω is subdivided.

Velocity and pressure can then be expressed in terms of this discretization as:

$$u_i^h = \sum_{j=1}^{N} \alpha_i^j v^j \qquad \text{and} \qquad p^h = \sum_{j=1}^{N} \beta^j q^j \tag{5}$$

where v and q are known as the trial functions. As a first guess we are going to use a Galerkin-type FEM formulation and therefore weighting functions will be equal to trial functions. Introducing the approximation into equation (4) the following expression is obtained:

$$\int_\Omega w_i^h \left(u_{i,t}^h + u_j^h u_{i,j}^h - f_i\right) + \nu \int_\Omega w_{i,j}^h u_{i,j}^h d\Omega - \frac{1}{\rho}\int_\Omega w_{i,i}^h p d\Omega - \int_{\Gamma_2} t_i^h w_i^h d\Gamma_2 = 0 \qquad \int_\Omega q^h u_{i,i}^h d\Omega = 0$$

$$\forall w_i^h \in H^h \;\; \forall q^h \in H^h, \text{ with } \quad w_i^h\big|_{\Gamma_1} = 0 \quad u_i^h\big|_{\Gamma_1} = b_i \quad u_i^h(x_j, 0) = u_{i0}^h(x_j) \tag{6}$$

Once the weighted residuals method has been applied and the approximation has been introduced, we should proceed to the integration of the elementary matrices and the resolution of the system of equations that is consequently obtained. Nonetheless the use of a Galerkin formulation, that takes weighting functions equal to trial functions, may lead to some problems in the obtaining of the flow solution by the FEM. The finite element method was applied when first released to structural problems and this solution thus obtained had the 'best approximation' property, that is, the difference between the finite element solution and the exact solution was reduced with respect to a certain norm. The stiffness matrix resulting from structural problems solved by the FEM is symmetric, instead the 'stiffness' matrix obtained for fluids is only symmetric if we consider the Stokes simplification, that is if we neglect the non-linear convective term $u_j u_{i,j}$. This simplification can only be made for the so-called creeping flow, or in other words sufficiently slow flows with scant depth. In any other case the coefficient matrix of the resulting system of equations is going to be non-symmetric and as a result, the 'best approximation' property is lost. The faster the flow turns, the more non-symmetric the coefficient matrix becomes. In practice this is featured by the appearance of some spurious node-to-node oscillations also known as 'wiggles', when a downstream boundary condition forces a fast change in the velocity field solution. One way of avoiding these oscillations is a refinement of the mesh such that convection no longer dominates on an element level.

The SUPG (Streamline/Upwinding Petrov-Galerkin) method succeeds in eliminating the spurious velocity field without carrying out the refinement of the mesh. The Galerkin FE formulation involving piecewise linear interpolations results in a central-difference type approximation of the derivatives. The finite difference solutions of fluid flow problems are also affected by these oscillations. In these approaches the use of upwinding differencing was discovered to be useful in the obtaining of stable velocity field solutions

The consideration of an upwind approximation of the derivatives brings about the addition of the proper amount of artificial diffusion that is needed in order to correct the underdiffusivity of the central difference approaching method. It should not be forgotten that this coefficient affects the way in which the derivatives are interpolated and not the actual physical magnitudes involved in the equations.

The multi-dimensional generalisation of the upwind treatment of the advection diffusion equation brings an additional problem and this is the appearance of an excessive diffusion normal to the flow. The streamline upwind method eliminates this spurious crosswind diffusion by considering and 'artificial' diffusivity that acts only in the direction of the flow (see Brooks[1] for details).

The streamline upwind Petrov-Galerkin weighting functions to be considered are of the form:

$$\tilde{w}_i = w_i + \tilde{p}_i \tag{7}$$

and therefore an extra term should be included in the equation (6) to yield:

$$\int_\Omega w_i^h \left(u_{i,t}^h + u_j^h u_{i,j}^h - f_i\right) + \nu \int_\Omega w_{i,j}^h u_{i,j}^h \, d\Omega - \frac{1}{\rho}\int_\Omega w_{i,i}^h p \, d\Omega - \int_{\Gamma_2} t_i^h w_i^h \, d\Gamma_2 +$$

$$+ \sum_e^N \int_{\Omega_e} \tilde{p}_i^h \left(u_{i,t}^h + u_j^h u_{i,j}^h - \nu u_{i,jj}^h + \frac{1}{\rho} p_{,i}^h - f_i \right) d\Omega = 0 \qquad \int_\Omega u_{i,i}^h q^h \, d\Omega = 0 \, ,$$

$$\forall w_i^h \in H^h \;\; \forall q^h \in H^h, \text{ with } \quad w_i^h \big|_{\Gamma_1} = 0 \quad u_i^h \big|_{\Gamma_1} = b_i \quad u_i^h\left(x_j, 0\right) = u_{i0}^h\left(x_j\right) \tag{8}$$

where $\tilde{p}_i^h$, is the discretized streamline upwind contribution to the weighting function defined as:

$$p_i^h = \frac{\bar{k} u_j^h w_{i,j}^h}{\left\| u^h \right\|^2} \tag{9}$$

where the multi-dimensional definition of $\bar{k}$ is:

$$\bar{k} = \frac{\bar{\xi} u_x^h h_x + \bar{\eta} u_h^h h_h}{2}$$

$$\bar{\xi} = \left(\coth \alpha_x - \frac{1}{\alpha_x} \right) \qquad \bar{\eta} = \left(\coth \alpha_h - \frac{1}{\alpha_h} \right)$$

$$\alpha_\xi = \frac{u_\xi^h h_\xi}{2\nu} \qquad \alpha_\eta = \frac{u_\eta^h h_\eta}{2\nu}$$
$$u_\xi^h = e_{\xi i} u_{ei}^h \qquad u_\eta^h = e_{\eta i} u_{ei}^h \tag{10}$$

where h_ξ, h_η and $e_{\xi i}$, $e_{\eta i}$ are the characteristic basic element lengths and unit vectors in the direction of ξ and η (see figure 1), α_ξ and α_η are the directional Reynolds numbers of the basic element, u_{ei}^h is the velocity in the interior of the element and ν is the cinematic viscosity of the fluid.

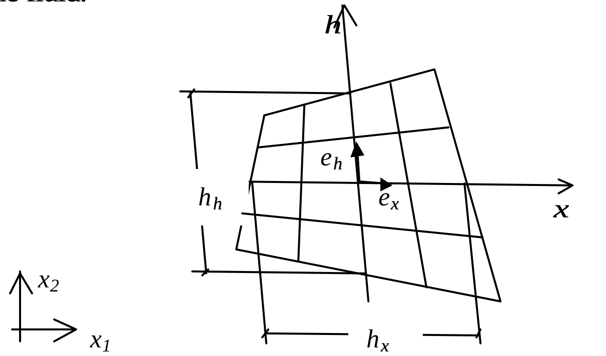

Fig. 1 Basic element.

The implementation of an SUPG-type stabilisation algorithm on the convective terms allows for good results on not very refined meshes and flows featured by high Reynolds numbers. The use of a very dense mesh involves high computational costs and consequently large amounts of memory requirements and vast CPU times. Consequently the SUPG formulation yields as a result, a better computational efficiency.

3.1 Mixed formulation.

Once the modified weighting functions have been introduced and after interpolating u_i, p and w_i let's go on with the resolution of the steady Navier-Stokes equations:

$$\int_\Omega \overline{w}_i^h \left(u_j^h u_{i,j}^h - f_i\right) + \nu \int_\Omega \overline{w}_{i,j}^h u_{i,j}^h d\Omega - \frac{1}{\rho}\int_\Omega \overline{w}_{i,i}^h p d\Omega - \int_{\Gamma_2} t_i^h w_i^h d\Gamma_2 = 0 \qquad \int_\Omega u_{i,i}^h q^h d\Omega = 0 \ ,$$

$$\forall w_i^h \in H^h \ \ \forall q^h \in H^h \text{, with } \quad w_i^h \big]_{\Gamma_1} = 0 \quad u_i^h \big]_{\Gamma_1} = b_i \quad u_i^h\left(x_j, 0\right) = u_{i0}^h\left(x_j\right) \tag{11}$$

The most intuitive way of solving the above equations is simply to transform the integral equation into a system of equations in which the unknowns were both velocity and pressure. This straightforward procedure although intuitive and simple is however quite expensive computationally speaking. The associated coefficient matrix of the resulting system for a Q1/P0 basic element, is 2M+N dimensional, where M and N are the number of velocity and pressure unknowns respectively, and therefore for refined meshes the memory requirements for the computational resolution may became very large.

The matrix notation for the steady problem could be:

$$\mathbf{C}(\mathbf{u})\mathbf{u} + \mathbf{A}\mathbf{u} - \mathbf{B}\mathbf{p} = \mathbf{f}$$
$$\mathbf{B}^T\mathbf{u} = \mathbf{0} \tag{12}$$

Where the matrices involved result from the assembling of the elementary matrices which can be written as:

$$A_{ij} = \nu\int_{\Omega_e} \frac{\partial w_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial w_i}{\partial y}\frac{\partial N_j}{\partial y}\, d\Omega$$
$$B_{ij}^x = \frac{1}{\rho}\int_{\Omega_e} \frac{\partial w_i}{\partial x}\chi_j\, d\Omega \qquad B_{ij}^y = \int_{\Omega_e} \frac{\partial w_i}{\partial y}\chi_j\, d\Omega$$
$$F_i^x = \int_{\Omega_e} w_i f_x\, d\Omega \qquad F_i^y = \int_{\Omega_e} w_i f_y\, d\Omega \tag{13}$$

The connective term $\mathbf{C}(\mathbf{u})\mathbf{u}$ is not a product of matrices but a non-linear velocity dependent function. This term should be avoided in order to transform the system into a linear system of equations. The method used for this linearization will be the, so-called, successive approximation method, because of its simplicity and the good results achieved for problems with Reynolds numbers of moderate order up to 10^3. In this method the convective term is iteratively obtained as a function of the previously determined values of the velocity field. The non-linear velocity dependent function for the n-th iteration $\mathbf{C}(\mathbf{u}^n)$ is taken as the product of the constant coefficient matrix C, times the column vector **u**, unknown of the problem at the present iteration.

$$\mathbf{C}(\mathbf{u}^n) \approx \mathbf{C}^n(\mathbf{u}^{n-1})\mathbf{u}^n \text{ with } n=1,2,...$$
$$C_{ij}^n = \int_{\Omega_e} w_i\left(N_k u_k^{n-1}\frac{\partial N_j}{\partial x} + N_k v_k^{n-1}\frac{\partial N_j}{\partial y}\right)d\Omega \tag{14}$$

The matrix **C** is not anymore a function of the present unknowns but depends on the previous values of the vector field, and is taken as zero as a first-iteration guess. The solution is usually achieved within some tens of iterations and depends on the Reynolds number of the flow, or in other words on the amount of convection we have to deal with. The compressed and expanded matrix forms can be expressed as:

$$\mathbf{C}^n(\mathbf{u}^{n-1})\mathbf{u}^n + \mathbf{A}\mathbf{u}^n - \mathbf{B}\mathbf{p}^n = \mathbf{f} \qquad \mathbf{B}^T\mathbf{u}^n = \mathbf{0}$$

$$\begin{bmatrix} \mathbf{A} + \mathbf{C}^{n-1} & \mathbf{0} & \mathbf{B}^x \\ \mathbf{0} & \mathbf{A} + \mathbf{C}^{n-1} & \mathbf{B}^x \\ (\mathbf{B}^x)^T & (\mathbf{B}^y)^T & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{u}^n \\ \mathbf{v}^n \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_x \\ \mathbf{f}_y \\ \mathbf{0} \end{bmatrix} \quad n=1,2,.... \tag{15}$$

set of iterative constant-coeficient linear system of equations, which gives solution to the viscous incompressible flow for given boundary conditions.

This formulation not only results in a large dimensioned system but also generates a stiffness matrix radically different to the narrow-band type of matrix, which is required for the direct resolution of the system of equations. To overcome these shortcomings, some way of avoiding the resolution of these large dimensioned systems is to be developed and following, the penalty and segregated methods are presented. The basic element used for these calculations will be the Q1/P0 (bilinear velocity-constant pressure), that, although does not verify the LBB conditions, achieves a good approximation for the problems considered.

3.2 Penalty formulation

This alternative formulation gives the possibility of imposing the incompressibility constraint without solving an auxiliary pressure equation by replacing the continuity equation by

$$u_{i,i} = -eP \tag{16}$$

where the so-called penalty parameter e is a number that tends to zero. This equation is incorporated into the dynamic equation and therefore a system of equations that depends on both velocity and pressure is transformed into a velocity-dependant single equation that converges to the fully incompressible problem as e approaches to zero.

The variational Lagrange multipliers technique gives solution to the problem of finding the stationary values of an integral expression I(u) (the dynamic equation), constrained by an additional equality J(u) = 0 (the incompressibility condition), transforming the problem into the obtaining of the stationary values of the modified expression

$$I(\mathbf{u}, \boldsymbol{l}) = I(\mathbf{u}) + \frac{1}{e} J(\mathbf{u})\,. \tag{17}$$

It may be proved that as e approaches to zero, the solution given by the minimisation of (17) converges to the solution of the problem posed in (1). For practical purposes the value of e must be balanced between a sufficiently small value in order to achieve a solution closer to the real one and a value large enough so as not to promote the ill conditioning of the stiffness matrix. By applying this method, we have achieved the splitting of the unknowns involved and therefore the velocities can be obtained in advance for a sufficiently small value of e by solving the discrete system of equations

$$\int_{\Omega} \boldsymbol{n} u_{i,i}^{h} w_{i,i}^{h} - f_i w_i^{h} + \frac{1}{e} u_{i,i}^{h} w_{i,i}^{h} \, d\Omega = 0$$

the pressure field can be post-processed using

$$p^{h} = -\frac{1}{e} u_{i,i}^{h}$$

The solution to these equations will approximate to that of the initial problem as ε tends to zero, provided that a consistency condition analogous to that of the mixed problem hold now for the penalty formulation, this approach may lead to the obtaining of a singular coefficient matrix associated to the penalty term

$$\frac{1}{\varepsilon}\int_{\Omega} u^h_{i,i} w^h_{i,i}\, d\Omega \tag{20}$$

As ε tends to zero, this term happens to dominate the system of equations and therefore the problem is over-constrained and the only possible solution may be the trivial one. This is a problem totally analogous to the one obtained when an equal order interpolation for both velocity and pressure is chosen in the mixed formulation. This problem can be avoided by making a so-called selective reduced integration of the elementary matrices involved in the resolution of the problem. A reduced numerical integration consists in using a quadrature rule that is not exact for the polynomials considered. The use of this reduced integration rule for the penalty term transforms the associated 'penalty' matrix into a rank deficient matrix and consequently 'unlocks' the obtaining of a non-trivial solution. For instance, if a bilinear interpolation of the velocity field is used, a Newton-Cotes interpolation that can integrate only constant functions exactly may be used.

The full steady convective-term-including penalised equations would be now

$$\int_{\Omega_h} \nu u^h_{i,i}\overline{w}^h_{i,i} + \frac{1}{\varepsilon} u^h_{i,i}\overline{w}^h_{i,i}\, d\Omega = \int_{\Omega_h} f_i \overline{w}^h_i\, d\Omega \tag{21}$$

Once the basic element has been chosen and the approximation for u_i

$$u^h_i = \sum_{j=1}^{N} u^j_i N^j \tag{22}$$

is introduced, we can carry out the integration and assembling of the elementary matrices to obtain the 'single matrix equation

$$\mathbf{C}(\mathbf{u})\mathbf{u} + \mathbf{A}\mathbf{u} + \frac{1}{\varepsilon}\overline{\mathbf{B}}\mathbf{p} = \mathbf{f} \tag{23}$$

Where

$$A_{ij} = \nu \int_{\Omega_e} \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y}\, d\Omega \qquad \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B}^x & \mathbf{D} \\ \mathbf{D}^T & \mathbf{B}^y \end{pmatrix}$$

$$C^n_{ij} = \int_{\Omega_e} \overline{w}_i \left(N_k u^{n-1}_k \frac{\partial N_j}{\partial x} + N_k v^{n-1}_k \frac{\partial N_j}{\partial y} \right) d\Omega$$

$$B^x_{ij} = \frac{1}{\rho}\int_{\Omega_e} \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\, d\Omega \qquad B^y_{ij} = \frac{1}{\rho}\int_{\Omega_e} \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y}\, d\Omega \qquad D^x_{ij} = \frac{1}{\rho}\int_{\Omega_e} \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial y}\, d\Omega$$

$$F^x_i = \int_{\Omega_e} N_i f_x\, d\Omega \qquad F^y_i = \int_{\Omega_e} N_i f_y\, d\Omega \tag{24}$$

3.3 Segregated formulation:

The penalty method succeeds in solving the Navier-Stokes equations with a large reduction in the execution time and great memory savings, thanks to the smaller number of equations to be solved. Anyhow, this is an approximate method that depends on the election of the parameter e, which for very small values produces an ill conditioning of the stiffness matrix and for too large values, may prevent the system from converging. Most finite difference and finite volume approaches to the problem of viscous incompressible flow employ some form of segregated method such as the SIMPLE and SIMPLEST algorithms in which the velocity and pressure unknowns are not obtained simultaneously. The segregated method calculates velocities and pressures in an alternative iterative sequence, requiring much less storing requirements than the conventional mixed method. Moreover, achieves a greater reduction in the number of equations compared to the penalty formulation that is reduced to the number of nodes, and avoids the use of the sometimes inconvenient penalty parameter.

Another gain of these segregated algorithms is that a mixed-order interpolation can be used. As it has already been said, the mixed and penalty methods required a velocity approximation different from that of the pressure. The easier-to-implement discretization of the domain in terms of the same basic functions for both velocity and pressure, leads to oscillation-free solutions even when the div-stability condition is not verified, and the tendency to produce the checkerboard pressure distribution is therefore eliminated.

The momentum equations are treated by the weighted residuals finite element method in the former cases, but this time the pressure term $\int_\Omega w_i p_{,i} \, d\Omega$ is not considered as an unknown anymore but included in the right hand of the system. For the first iteration the pressures are taken as zero as a first guess and for the following, this zero value will be properly corrected. With this, we do not only get rid of the 'unwanted' pressure unknown, but also accomplish that, due to the independence of the x-component dynamic equation with respect to v and the y-component with respect to u, the system to be solved is of n dimension. Following Rice[8] the dynamic equation can be written:

$$\mathbf{C}(\mathbf{u},\mathbf{v}) + n\mathbf{A}\mathbf{u} = \mathbf{K}^u\mathbf{u} = \mathbf{f}^u - \int_\Omega w_i \frac{\partial N_j}{\partial x} p_j \, d\Omega$$
$$\mathbf{C}(\mathbf{u},\mathbf{v}) + n\mathbf{A}\mathbf{v} = \mathbf{K}^v\mathbf{v} = \mathbf{f}^v - \int_\Omega w_i \frac{\partial N_j}{\partial y} p_j \, d\Omega \qquad (25)$$

This equation could be rewritten as:

$$u_i = \tilde{u}_i - \left(K^u \frac{\partial p}{\partial x} \right)_i$$
$$v_i = \tilde{v}_i - \left(K^v \frac{\partial p}{\partial y} \right)_i \qquad (26)$$

where $\tilde{u}_i$ and $\tilde{v}_i$ are termed pseudo-velocities and defined as :

$$\tilde{u}_i = \frac{1}{k_{ii}^u}\left(-k_{ij}^u u_j + f_i^u\right)_{i\neq j}$$
$$\tilde{v}_i = \frac{1}{k_{ii}^v}\left(-k_{ij}^v v_j + f_i^v\right)_{i\neq j} \qquad (27)$$

and the pressure-velocity coupling coefficients K_i^u, K_i^v are equal to:

$$K_i^u = \frac{1}{k_{ii}^u}\int_\Omega w_i\, d\Omega$$
$$K_i^v = \frac{1}{k_{ii}^v}\int_\Omega w_i\, d\Omega \qquad (28)$$

The approximation (26) will be the required relation between velocity and pressures. Although it is only comparable to the use of a secant approximation in Newton´s method, it is enough for the algorithm to converge. Since an equal order bilinear approximation is used for the pressure, the continuity residual is formed by the product of the same weighting functions as those used in the dynamic equations:

$$\int_{\Omega_h} w_i u_{j,j}\, d\Omega = 0\,,$$

using the divergence theorem, the following expression is obtained

$$\int_{\Omega_h} w_{i,j} u_j\, d\Omega_h - \int_\Gamma w_i u_j n_j\, d\Gamma = 0 \qquad (29)$$

where n_j is the normal outward vector with respect to the boundary Γ, or in expanded form

$$\int_\Omega \frac{\partial w_i}{\partial x} w_j u_j + \frac{\partial w_i}{\partial y} w_j v_j\; d\Omega = \int_\Gamma w_i\left(w_j u_j n_x + w_j v_j n_y\right)\, d\Gamma \qquad (30)$$

Substituting (26) into (30) we obtain the discretized integral expression

$$\mathbf{K}^p\mathbf{p} = \mathbf{f}^p \qquad (31)$$

$$k_{ij}^p = \int_{\Omega_h} \frac{\partial N_i}{\partial x} N_k K_k^u \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} N_k K_k^v \frac{\partial N_j}{\partial y}\, d\Omega \qquad (32)$$

$$f_i^p = \int_{\Omega_h} \frac{\partial N_i}{\partial x} N_j \tilde{u}_j + \frac{\partial N_i}{\partial y} N_j \tilde{v}_j\; d\Omega - \int_{\Gamma_h} N_i\left(N_j u_j n_x + N_j v_j n_y\right) d\Gamma \qquad (33)$$

As it can be observed, the coefficient matrix for the pressure equation is similar to that obtained for the diffusive term in the conventional finite element formulation if the viscosity is replaced by *K*. This equation results consequently in a robust system with a symmetric and positive definite associated matrix. To solve this pressure equation we should take into account not only the prescribed nodal pressure values, that usually are given at the outlets and at least are given a point, but also the implicitly prescribed pressures on the nodes where the velocity is given. For this type of implicitly imposed pressure, the pseudo-velocities are set equal to the prescribed nodal velocities and therefore the values of K_i^u and K_i^v are set equal to zero.

Once we have solved the pressure system, velocities are updated using

$$u_i = \tilde{u}_i - \frac{1}{k_{ii}^u}\int_\Omega w_i \frac{\partial N_j}{\partial x} p_j \; d\Omega ; \qquad v_i = \tilde{v}_i - \frac{1}{k_{ii}^v}\int_\Omega w_i \frac{\partial N_j}{\partial y} p_j \; d\Omega , \tag{34}$$

to ensure continuity.

The iterative process is based in assuming a zero velocity as a first guess for the resolution of the dynamic equation. Once the pseudo-velocities and the pressure-velocity coefficients have been calculated, the pressure continuity system is assembled and solved. Finally the velocities are updated, making use of the newly determined pressure field and with both new velocities and pressures the dynamic equations are reassembled, the dynamic system is again solved and the same procedure is repeated until convergence is achieved.

When using a segregated algorithm, the use of uncoupled velocity and pressure fields may lead to the divergence of the whole process. To avoid this problem, an under-relaxation of the unknowns is to be introduced in order of the algorithm to converge. The linear relaxation formulae to be used is

$$f^n = f^{n-1} + a\left(f^n - f^{n-1}\right), \tag{35}$$

where f^n is the value of the unknowns (either velocity or pressure) at the present iteration.

4 Numerical Results

The FEM results in a system of linear equations containing a big amount of equations and unknowns. For instance when a mixed formulation together with a Q1/P0 (Bilinear velocity-Constant pressure) quadrilateral element the number of equations and unknowns is settled as twice the number of nodes plus the number of basic elements and the storage of such a big amount of information requires a clever data-keeping strategy. When using a direct numerical method for the resolution of the system of equations, an alternative way of data storing is the so-called skyline or column profile storage. Instead of storing every single matrix-element, we could think of storing only the first non-zero element of each column and the following elements in that column up to the diagonal. Due to the fact that we are dealing with convective-term including formulations the coefficient matrix associated to the system is going to be non-symmetric, and another vector-valued variable is required for the lower triangular matrix. Together with this vector valued variable *v*, an additional pointer vector *p* has to be defined so as to indicate the position of the elements. Nevertheless when either the mesh is progressively refined or very large domains are going to be considered, the memory requirements became extraordinarily large, in order to avoid this problem an alternative and more efficient storing schedule should be used.

The 'cheapest' storing mechanism is to keep exclusively those elements different from zero. This is a much better procedure that avoids wasting memory resources in storing mid-height zeros, which can be more numerous than the number of non-zero elements even when the mesh is re-numbered so as to reduce the band width to a minimum and specially when a mixed method is used. Provided that the sparse storage cannot be used in combination with a direct solver because some elements could be 'thrown out' of the sparse stencil, when this type of storage is used, some other algorithm should be implemented to solve the system of equations. For the present calculations a Preconditioned Biconjugate Gradient Method (PBCG) type of solver will be implemented in order to solve the resulting system.

The skyline storing together with a Crout solver, will be also used in those meshes with a reasonable number of elements, therefore obtaining exact and one-step solutions. For more refined meshes and especially when a mixed formulation is implemented, a PBCG-type algorithm is used.

We will assume we have reached convection convergence once

$$\max_{i=1,..N}\left|f_i^n - f_i^{n-1}\right| < 10^{-4} \tag{36}$$

for each of the unknowns.

The problem of the flow in a square cavity has been considered to check the algorithms by comparison of their results with those of other authors. The velocity is settled as one on the topside and the no-slip condition is considered on the other sides. The pressure is fixed as zero in the centre of the lower side of the cavity. The domain has been interpolated in terms of a 31x31 node non-regular mesh with Q1/P0 basic elements.

The results for the penalty algorithm for Reynolds numbers of 100, 1000, 5000 and 10000 are shown bellow. In all the cases considered, the penalty parameter has been taken as 10^{-4}. The solution has been obtained using a PBCG iterative method for solving the system of equations.

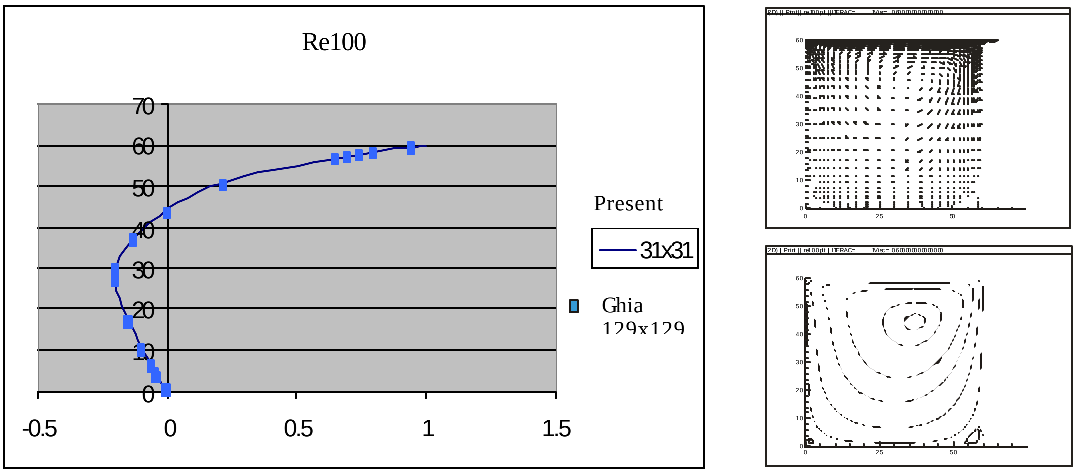

Fig 2. - Horizontal velocities along a central vertical line compared with those of Ghia [2] for a Reynolds number of 100. Velocity field and streamlines.

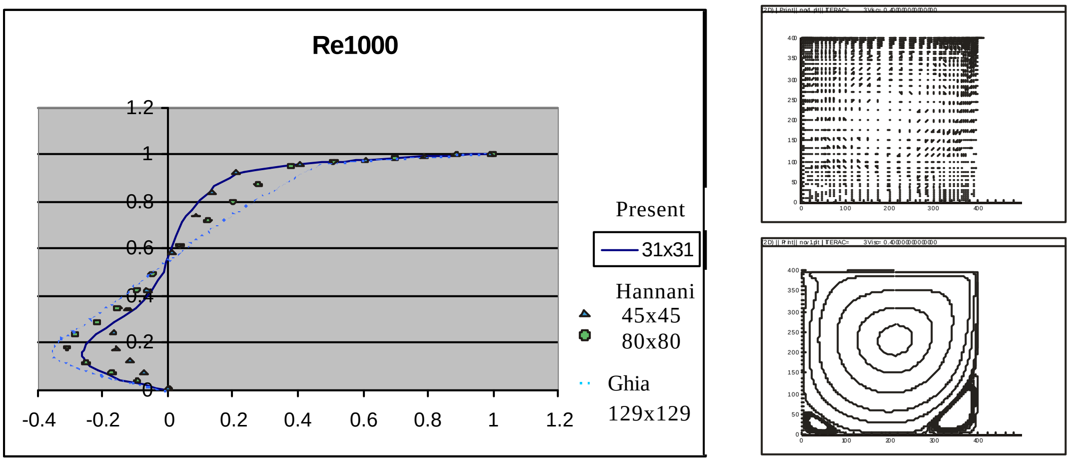

Fig 3. - Horizontal velocities along a central vertical line compared with those of Hannani [3] and Ghia [2] for a Reynolds number of 1000. Velocity field and streamlines.

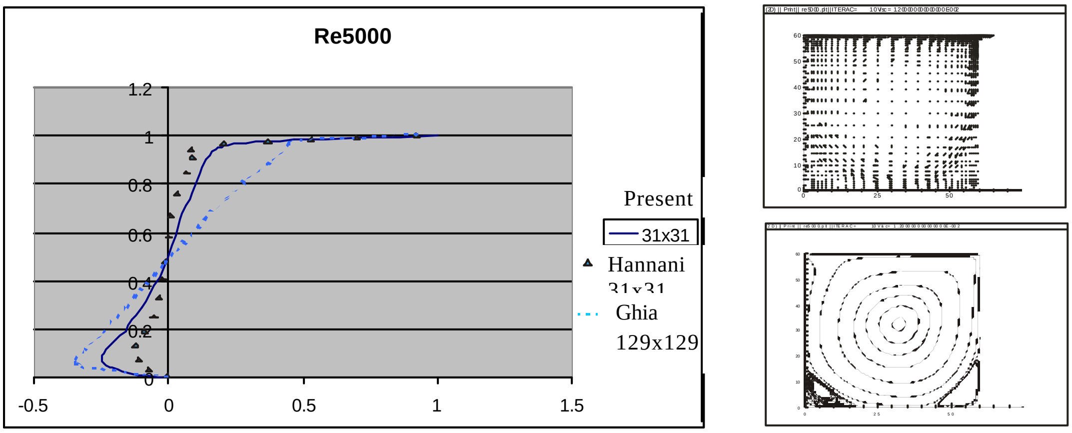

Fig 4. - Horizontal velocities along a central vertical line compared with those of Hannani [3] and Ghia [2] for a Reynolds number of 5000. Velocity field and streamlines.

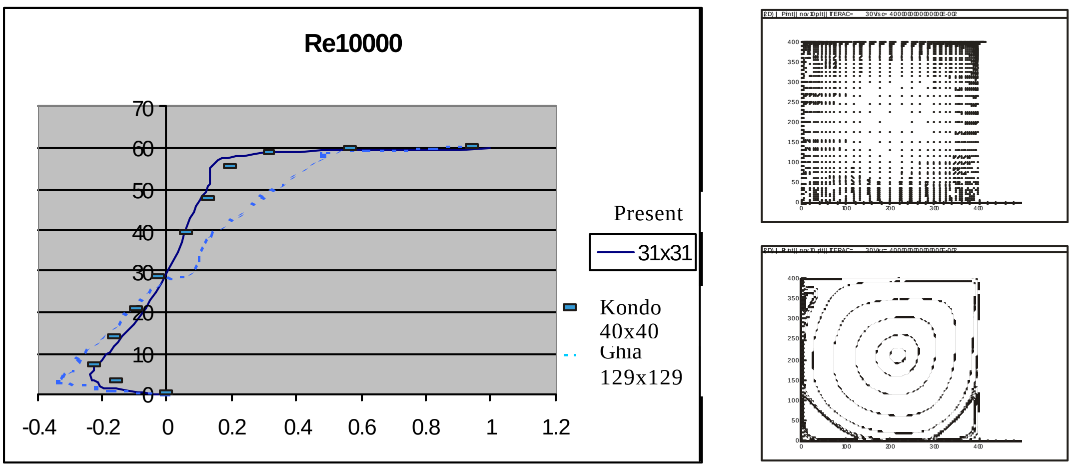

Fig 5.- Horizontal velocities along a central vertical line compared with those of Kondo [4] Ghia [2] for a Reynolds number of 10000. Velocity field and streamlines.

For the mixed formulation a PBCG algorithm has been used to solve the system of equations with a column profile storing procedure. The results for the pressure and velocity for a Reynolds number of 10000, with a 1681-node, 1600-element mesh are shown bellow.

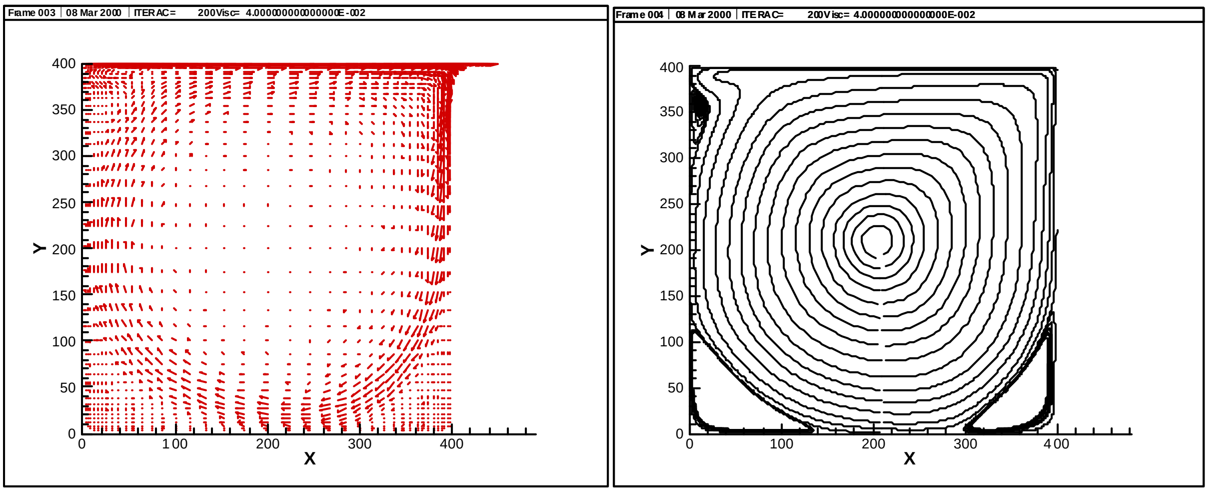

Fig 5 Vector field and streamlines (41x41 non-regular mesh, mixed).

When the segregated algorithm is used, an under-relaxation of the unknowns has to be introduced in order of the algorithm to converge. The relaxation parameters used were taken as $a_u = 0.7$ and $a_p = 0.2$. The results for pressures and velocities obtained for the segregated formulation when a Reynolds number of 400 is used are shown bellow.

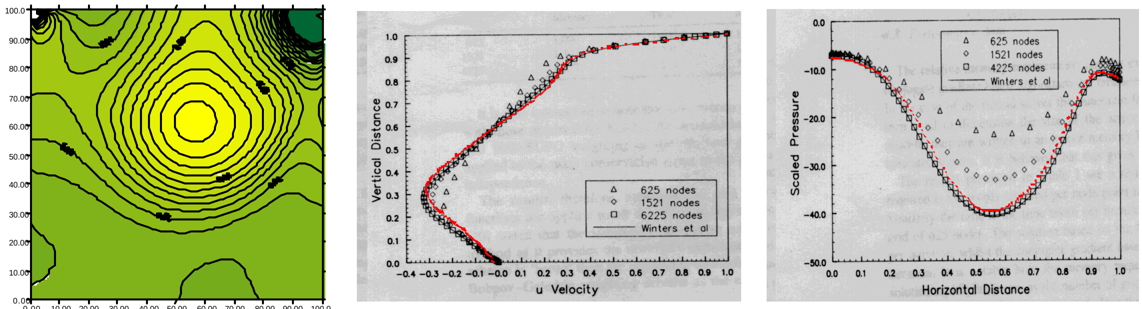

Fig 6. - Pressure field, Horizontal velocities along a central vertical line and Pressures along a horizontal central line compared with those of Winters and du Toit [5].

The flow over a backward step of width 30 l.u. and length 440 l.u. has been also calculated, making use of a mixed formulation with a PBCG solver. The results for a Reynolds number of 100 and 1000 and a regular mesh of 3381 nodes and 3150 elements are shown bellow.

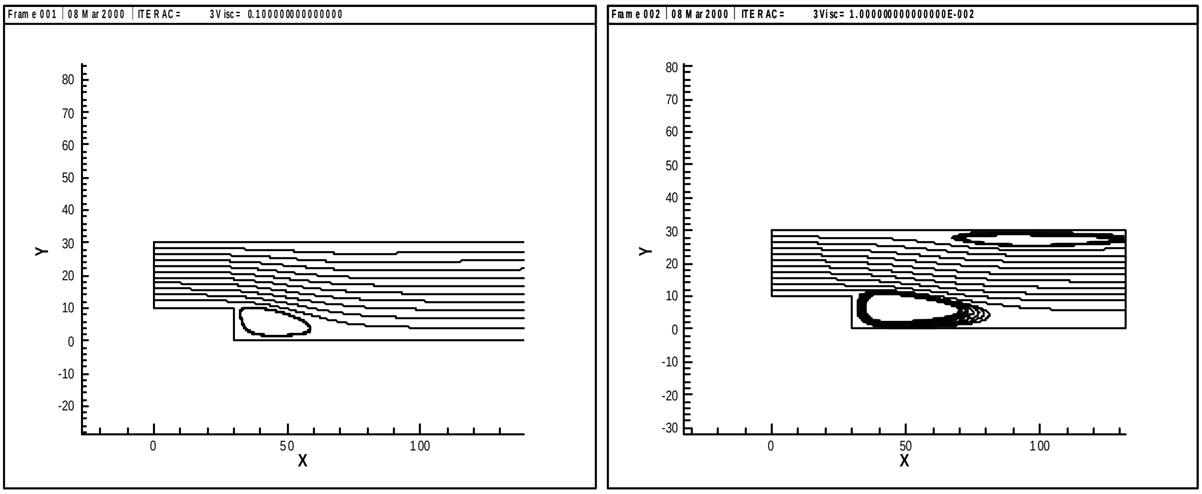

Fig 6 Streamlines for the backward step problem (Re 100 and 1000).

The steady flow around a circular cylinder has also been calculated for a Reynolds number of 500 on a 732-node mesh, with the expected results.

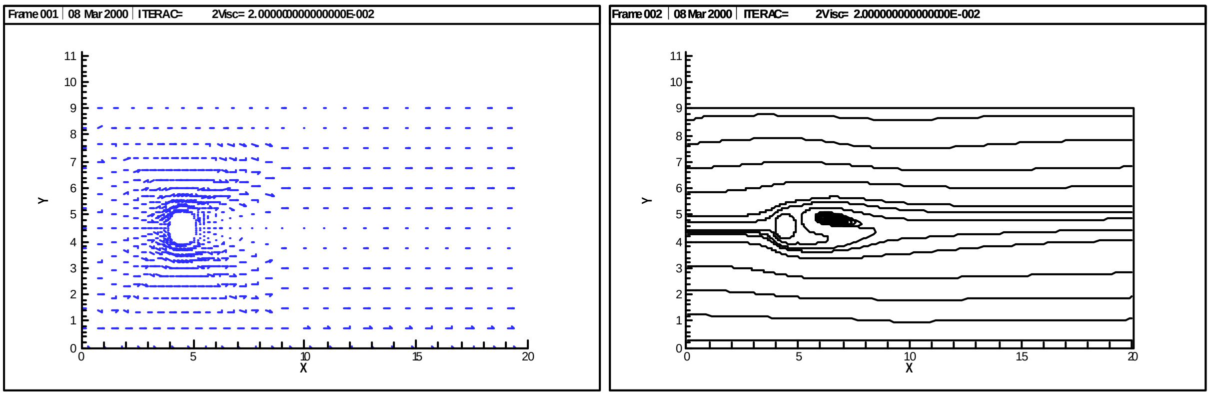

Fig 7 Velocity field and streamlines for the flow around a cylinder problem.

The program has also been used to calculate the flow in some real cases, helping in the obtaining of the optimal shape of some water treatment plants structures. The results for the wastewater flow in a biological reactor are presented as an example. A scaled flow of 4 l/h is re-circulated on a cavity in which a biological film purifies the water flow with a resulting Reynolds number of 100. The domain shown in figure 8 represents the cross section of the biological reactor, which has been split into 1916 nodes. The flow has been solved using a 2D flat Navier-Stokes flow with mixed formulation and PBCG solver. A spillway boundary condition has been considered for the top border. The resulting streamlines and vector field is sketched bellow.

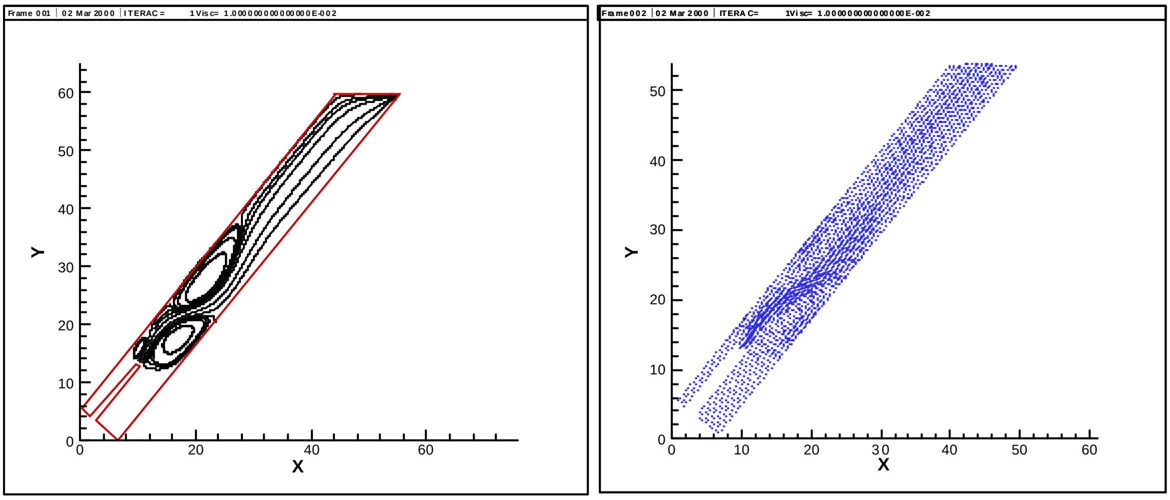

Fig 8 Streamlines and velocity field for the biological reactor problem.

5 Conclusions

The program seems to achieve good results for the three formulations as can be seen in the plots, compared with results from Winters, Hannani, Ghia, Kondo and others. The results from the present study seem to adjust to those of the others, with even a less refined mesh. When a mixed formulation is used, the matrices involved in the resolution of the Navier-Stokes equations became large and this implies that very big meshes can not be used, therefore small vortices are not detected. However the iteration process is reduced to the achievement of the convection effect, so a few iterations are needed, and therefore the CPU time involved is less than one hour in a conventional PC. When a mixed or segregated algorithm is used, the iterative process becomes much longer. The program has been run in a Digital AlphaServer 1000A computer, taking CPU times of one or two hours for the 31x31 mesh, depending on the Reynolds number. With respect to the basic elements, when a Q1/P0 quadrilateral is used, the pressure results for the mixed algorithm are polluted by a checker board pressure mode that anyway, can be removed by a proper smoothing of the results. This unwanted distortion does not appear when an equal order four-node basic element is used for the segregated procedure.

REFERENCES

[1] A. N. Brooks, J. ,R. Hughes. Streamline Upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Comput. Methods Appl. Mech. Engrg 32 (1982) 199-259.

[2]. U. Ghia, K. N. Ghia and C.T. Shin. High- Re solutions for incompressible flow using the Navier-Stokes equation and the multigrid method. Journal of Computational Physics 48, 387-411 (1982).

[3] S. K. Hannani, M. Stanislas and P. Dupont. Incompressible Navier-Stokes computations with SUPG and GLS formulations- A comparison study. Comput. Methods Appl. Mech. Engrg 124 (1995) 153-170.
[4] N. Kondo. Third-Order upwind finite element formulations for incompressible viscous flow problems. Comput. Methods Appl. Mech. Engrg. 93 (1991) 169-187.
[5] C. G. du Toit. Finite element solution of the Navier-Stokes equations for incompressible flow using a segregated algorithm. Comput. Methods Appl. Mech. Engrg. 151 (1998) 131-141.
[6] N. Kondo. Third-Order upwind finite element formulations for incompressible viscous flow problems. Comput. Methods Appl. Mech. Engrg. 93 (1991) 169-187.
[7] H.G. Choi and J.Y. Yoo. Streamline upwind scheme for the segregated formulation of the Navier-Stokes equation. Numerical Heat Transfer, PartB, 25:145-161, 1994.
[8] J. G. Rice and R. J. Schnipke. An equal-order velocity pressure formulation that does not exhibit spurious velocity modes. Comput. Methods Appl. Mech. Engrg. 58 (1986) 135-149.
[9] U. Ghia, K. N. Ghia and C.T. Shin. High- Re solutions for incompressible flow using the Navier-Stokes equation and the multigrid method. Journal of Computational Physics 48, 387-411 (1982).
[10] H. G. Choi, H. Choi, J.Y. Yoo. A fractional four-step finite element formulation of the unsteady incompressible Navier-Stokes equations using SUPG and linear equal-order element methods. Comput. Methods Appl. Mech. Engrg. 143 (1997) 333-348.
[11] C. G. du Toit. Finite element solution of the Navier-Stokes equations for incompressible flow using a segregated algorithm. Comput. Methods Appl. Mech. Engrg. 151 (1998) 131-141.
[12] S. K. Hannani, M. Stanislas and P. Dupont. Incompressible Navier-Stokes computations with SUPG and GLS formulations- A comparison study. Comput. Methods Appl. Mech. Engrg 124 (1995) 153-170.
[13] A. N. Brooks, J. ,R. Hughes. Streamline Upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Comput. Methods Appl. Mech. Engrg 32 (1982) 199-259.
[14] S. W. Kim. A fine grid finite element computation of two-dimensional high reynolds numbers flows. Computers and Fluids Vol 16 no 4 pp 429-444 1988.
[15] L. P. Franca. S. L. Frey. Stabilized finite element methods: II. The incompressible Navier-Stokes equations. Comput. Methods Appl. Mech. Engrg 99 (1992) 209-233.
[16] I. E. Barton. Comparisson of Simple- and Piso-type algorithms for transient flows. International Journal for numerical methods in fluids, vol 26, 459-483 (1998)
[17] A. Pascau, C. Pérez F. J. Serón. A comparison of segregated and coupled methods for the solution of the incompressible Navier-Stokes equations. Comunications in numerical methods in engineering, vol 12, 617-630 (1996).
[18] C. C. Yu and J. C. Heinrich. Petrov- Galerkin method for multidimensional, time-dependent, convective-diffusion equations. International Journal for numerical methods in engineering, vol. 24, 2201-2215 (1987)

[19] O. C. Zienkiewicz and J. Wu. Incompressibility without tears- How to avoid restrictions of mixed formulation. International Journal for numerical methods in engineering, vol. 32 1189-1203 (1991).
[20] P. A. B. De Sampaio. A Petrov-Galerkin formulation for the incompressible Navier-Stokes equations using equal order interpolation for velocity and pressure. International Journal for numerical methods in engineering vol. 31, 1135-1149 (1991).
[21] C.T. Shaw. Using a segregated finite element scheme to solve the incompressible Navier-Stokes equations. International Journal for numerical methods in fluids, vol. 12, 81-92 (1991).
[22] T. W. H. Sheu and S.-M. Lee. A segregated solution algorithm for incompressible flows in general co-ordinates. International Journal for numerical methods in fluids, vol 22, 515-448 (1996)
[23] B. M. Mohammadi, O Pironneau. Unsteady separated turbulent flows computation with wall-laws and k-ε model. Comput. Methods Appl. Mech. Engrg. 148 (1997) 393-405.
[24] A. C. Benim and W. Zinser. A segregated formulation of Navier-Stokes equations with finite elements. Comput. Methods Appl. Mech. Engrg. 57 (1986) 223-237.
[25] V. Haroutunian, M. S. Engelman and I. Hasbani. Segregated finite element algorithms for the numerical solution of large scale incompressible flow problems. International Journal for numerical methods in fluids vol. 17 323-348 (1993).