Equal-order interpolations: a unified approach to stabilize the incompressible and advective effects

M. Storti, N. Nigro, S. Idelsohn*

Grupo de Tecnología Mecánica del INTEC, Universidad Nacional del Litoral and CONICET, Güemes 3450, 3000, Santa Fe, Argentina

Abstract

In this paper we present a new SUPG formulation for compressible and near incompressible Navier–Stokes equations [5]. It introduces an extension of the exact solution for one-dimensional systems to the multidimensional case, in a similar way to that arising in the scalar problem. It is important to note that this formulation satisfies both the one-dimensional advective–diffusive system limit case and the advection-dominated multidimensional system case presented by Mallet et al. Another interesting feature of this formulation is that it introduces naturally a stabilizing term for the incompressibility condition, in a similar way to that found by other authors [1–4]. However, in our formulation the stabilization is introduced to the whole system of equations, while other authors introduce a term to stabilize the incompressibility condition and another one for the advective term.

In Section 1 we present Navier–Stokes equations for compressible flow and, then, we pass to detail several topics related to the numerical discretization of such advective–diffusive multidimensional systems of PDEs, in the Petrov–Galerkin context. The method is applicable and described for the general Re > 0 laminar flow, but the nature of the stabilizing effect of the artificial diffusion matrix introduced is discussed in depth for the simpler Stokes (Re = 0) flow.

Several numerical results are shown in Section 5, taking the well-known test problem of the square-cavity and a variant of this, namely a multiply connected square-cavity, as a validation for this code.

0. Notation and symbols

x, X	vectors in lower case, matrices in upper case
^{,i}	$\partial X / \partial x_i$
X^T	transpose of X
a	velocity in advection–diffusion 1D, node index
a	advection velocity
A_i	i advective matrix in i coordinate direction
b	right-hand side of the system
B	idem
B_j^e	right-hand side nodal contribution
c	sound velocity
e	internal energy, element index
F_j^e	nodal contribution of the internal fluxes

---

* Corresponding author. e-mail: mstorti@venus.unl.edu.ar

F^e_{lj}	nodal contribution of the surface fluxes
F^i_a	fluxes vector, advective part
F^i_d	fluxes vector, diffusive part
Fo	Fourier number
	mesh size
h	fluxes vector on Neumann boundary conditions
	diffusivity
k	wave number vector
k^{num}	numerical diffusivity
K^{num}	numerical diffusivity matrix
K_{ij}	diffusivity matrix, spatial components ij
M	Mach number
M_j	mass matrix corresponding to j node
N_{el}	number of elements
$\hat{n}$	normal unit vector
N_a	interpolation function of node a
p	pressure
Pe	Pèclet number
P_a	perturbation function
q	heat flux vector
S	basis transformation matrix
t	time
u	velocity vector
u	(1) property to be advected in 1D
	(2) x component of u
U	state vector
V	auxiliary state vector
W_a	weight function
x	mesh size
Λ	eigenvalues matrix
τ	(1) intrinsic time matrix
	(2) deviatoric stress tensor
ϵ	perturbation parameter
$\tilde{\epsilon}$	non-dimensionalized perturbation parameter
Γ_h	domain boundary
Γ_{int}	inter-element contour
μ	dynamic viscosity
∇	gradient operator
ν	kinematic viscosity
ϕ	second magic function
ψ	magic function
τ	scalar intrinsic time
Δ	Laplace operator
Δx	grid spacing
κ	thermal conductivity
λ	second coefficient of viscosity
λ_m	eigenvalue
θ	temperature
ρ	density
Ω	volume
Ω_e	elementary volume

1. Mathematical model—conservative form of the Navier–Stokes equations

Navier–Stokes equations written in conservative form are

$$F^i_{a,i} = F^i_{d,i} + b \tag{1}$$

Implicitly, we assume $(\ )_{,i} = \partial(\ )/\partial x_i$. $U \in \mathbb{R}^5$ is the fluid local state vector, where $U = [\rho, \rho u^T, \rho e]^T$, with ρ, u, e as the density, velocity and total energy of the fluid. $F_a, F_d \in \mathbb{R}^{5 \times 3}$ are the advective and diffusive fluxes, respectively, that depend on the state vector and its gradient as

$$F_a(U) = \begin{bmatrix} \rho u^T \\ \rho u u^T + p I_{3\times3} \\ (\rho e + p) u^T \end{bmatrix} \qquad F_d(U, \nabla U) = \begin{bmatrix} 0 \\ \tau \\ (\tau \cdot u + q)^T \end{bmatrix} \tag{2}$$

where p, τ, q are the thermodynamic pressure, the stress deviatoric tensor and the heat flux vector, respectively, with

$$\tau_{jk} = \mu(u_{j,k} + u_{k,j}) + \lambda u_{l,l} \delta_{jk} \tag{3.1}$$

$$q_j = \kappa \theta_{,j} \tag{3.2}$$

where μ is the dynamic viscosity, λ is the second viscosity coefficient, κ the thermal conductivity and θ the temperature.

We finish the description of the mathematical model, introducing the state equation of the fluid and the relation between the energy and two of the thermodynamic variables of the fluid

$$\theta = \theta(p, \rho) , \qquad e = e(p, \rho) \tag{4}$$

This kind of advective–diffusive system is incompletely parabolic and not completely parabolic because the continuity equation has not diffusive term.

2. Numerical spatial discretization—a Petrov–Galerkin formulation overview

2.1. Introduction

The numerical formulation is based on Petrov–Galerkin weighted residual method which allows test functions that can be different from the interpolation ones and not necessarily continuous. This method introduces the numerical dissipation needed to stabilize the system in advection-dominated problems, keeping the consistency with the continuum problem, and is one of the most referenced in the CFD area by the finite element method [6–8]. We suppose that

$$F^i_a(U) = A_i U , \qquad F^i_d(U, \nabla U) = K_{ij} U_{,j} \tag{5}$$

where A_i and K_{ij} are constant matrices. For each node a there is an interpolation function N_a (hat type in 1D, bilinear in 2D, and multi-linear in general) and a test function $W_a = N_a + P_a$, where P_a is called the perturbation function. The standard Galerkin method is recovered when we impose $P_a \equiv 0$. The W_a (and, of course, P_a) are not necessarily continuous through the inter-element boundaries. Now we describe the variational formulation employed and, in the following sections, we explain how to get the perturbation functions.

$$\int_\Omega (N^T_a A_i U_{,i} + N^T_{a,i} K_{ij} U_{,j}) \, d\Omega + \sum_{e=1}^{N_{el}} \int_{\Omega_e} P^{eT}_a (A_i U_{,i} + K_{ij} U_{ij} - b) \, d\Omega = \int_\Omega N^T_a b + \int_\Gamma N^T_a h \, d\Gamma \tag{6}$$

h is the diffusive flux imposed on the boundary Γ_h. The Euler–Lagrange form is obtained through classical integration by parts:

$$\sum_{e=1}^{N_{el}} \int_{\Omega_e} W_a^{\mathrm{T}}(A_i U_{,i} - K_{ij} U_{,ij} - b) \, d\Omega + \int_{\Gamma_{int}} N_a^{\mathrm{T}} [n_i K_{ij} U_{,j}] \, d\Gamma + \int_{\Gamma_h} N_a^{\mathrm{T}}(n_i K_{ij} U_{,j} - h) \, d\Gamma = 0 \tag{7}$$

where

$$[n_i K_{ij} U_{,j}](x) = n_i(x)\{(K_{ij} U_{,j})(x^+) - (K_{ij} U_{,j})(x^-)\} \tag{8}$$

is the jump in the diffusive flux throughout the inter-element boundary, x is a point which lies there, $x^{\pm}$ are points belonging to each side of the boundary and Γ_{int} is the inter-element contour. Consistency is warranted because the continuum solution is also solution of this variational formulation.

In the following sections we present the type of perturbation function that we have used. We start remembering the simple scalar one-dimensional problem and, then, we will extend these results to multidimensional systems.

2.2. One-dimensional scalar advection–diffusion equation

Here

$$au_{,x} = ku_{,xx} \tag{9}$$

is the equation to be solved where a is the advection velocity of the fluid, k is the diffusivity and u the dependent variable. By a centered finite difference scheme or, similarly, by a finite element (Galerkin method) formulation, we obtain the following linear system on a homogeneous grid of spacing Δx:

$$a\frac{u_{i+1} - u_{i-1}}{2\,\Delta x} = k\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}. \tag{10}$$

The truncation error is $O(\Delta x^2)$ but it produces wiggles for advection dominated problems characterized by a Péclet number $Pe = u\,\Delta x / k \gg 1$. In a finite difference context it is well known that the previous defect is solved by the introduction of artificial diffusion or by an upwinding technique.

On the other hand, exact solution to the 1D advection–diffusion problem with a constant mesh size and Dirichlet boundary conditions could be obtained if the k (physical diffusion) of the discrete problem were changed by

$$k' = k + k^{\mathrm{num}}, \qquad k^{\mathrm{num}} = \frac{a\,\Delta x}{2}\,\psi(\mathrm{Pe}) \tag{11}$$

where $\psi(\mathrm{Pe})$ is the so-called magic function defined as (see Fig. 1):

$$\psi(\mathrm{Pe}) = \coth\left(\frac{\mathrm{Pe}}{2}\right) - \frac{2}{\mathrm{Pe}} \tag{12}$$

This expression shows up when we insert the exact nodal values within the discrete scheme. For $\mathrm{Pe} \to \pm\infty$ the scheme tends to the well-known full-upwind finite difference method.
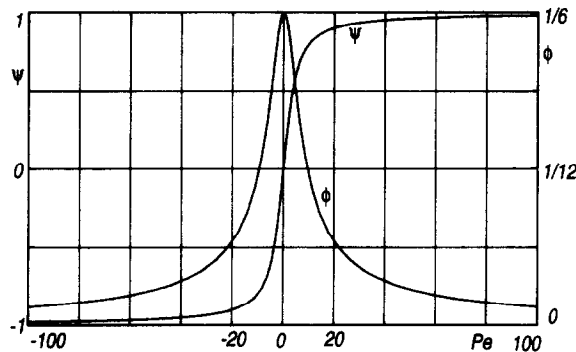

Fig. 1. Magic function $\psi(\mathrm{Pe})$ and its associated ϕ.

The discrete system involved is

$$a \frac{u_{i+1} - u_{i-1}}{2 \Delta x} = (k + k^{\text{num}}) \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} \tag{13}$$

This scheme can be cast as a Petrov–Galerkin formulation if we take the perturbation function for the node i, as

$$P_i = \tau a N_{i,x}, \qquad \tau = \frac{\Delta x}{2|a|} \psi(\text{Pe}) \tag{14}$$

τ is called, in SUPG terminology, an *intrinsic time scale*, where SUPG stands for the 'Streamline Upwind Petrov–Galerkin' method. This kind of test functions produces a conservative scheme.

2.3. One-dimensional advective–diffusive systems

We focus on the version of (1) without source term

$$AU_{,x} = KU_{,xx}, \qquad U \in \mathbb{R}^m, \qquad A, K \in \mathbb{R}^{m \times m} \tag{15}$$

where m is the number of equations of the system. The generalization from the scalar to the system case consists in finding a similarity transformation to carry the system to a set of m uncoupled advective–diffusive scalar equations. We define a new variable vector $U = SV$, where S is a non-singular change of basis matrix to be determined, and multiply the whole system by $S^{-1}K^{-1}$:

$$S^{-1}K^{-1}ASV_{,x} = V_{,xx} \tag{16}$$

(We suppose that K is non-singular. The case of Navier–Stokes equations, where this matrix is singular, will be treated later on.) Then, the problem is reduced to find a matrix S for which

$$S^{-1}K^{-1}AS = \Lambda \tag{17}$$

with Λ diagonal matrix. For symmetrizable parabolic systems it is always possible to find the above matrix. This is so because in this case the eigenvalue problem is equivalent to diagonalize the matrix $K'^{(-1/2)}A'K'^{(-1/2)}$, where A', K' are those matrices arising from the transformation of A, K to the symmetric basis, respectively. This last matrix is symmetric and, therefore, diagonalizable. Then, the kth equation in the diagonal basis reads:

$$\lambda_k v^k_{,x} = v^k_{,xx} \tag{18}$$

and, as we have seen above (see Eqs. (9) and (10)) it can be solved without oscillations with the scheme

$$\lambda_k \frac{v^k_{i+1} - v^k_{i-1}}{2 \Delta x} = \left\{ 1 + \frac{\lambda_k \Delta x}{2} \psi(\lambda_k \Delta x) \right\} \frac{v^k_{i+1} - 2v^k_i + v^k_{i-1}}{\Delta x^2} \tag{19}$$

The matricial form is

$$\Lambda \frac{V_{i+1} - V_{i-1}}{2 \Delta x} = \left\{ I + \frac{\Lambda \Delta x}{2} \psi(\Lambda \Delta x) \right\} \frac{V_{i+1} - 2V_i + V_{i-1}}{\Delta x^2} \tag{20}$$

Multiplying by kS and transforming back to the U variable:

$$KSAS^{-1} \frac{U_{i+1} - U_{i-1}}{2 \Delta x} = KS \left\{ I + \frac{\Lambda \Delta x}{2} \psi(\Lambda \Delta x) \right\} S^{-1} \frac{U_{i+1} - 2U_i + U_{i-1}}{\Delta x^2} \tag{21}$$

and taking into account that

$$\psi(\Lambda \Delta x) = \psi(S^{-1}K^{-1}AS \Delta x) = S^{-1}\psi(K^{-1}A \Delta x)S \tag{22}$$

we arrive to the following expression:

$$A \frac{U_{i+1} - U_{i-1}}{2\Delta x} = \left\{ K + A\psi(K^{-1}A\,\Delta x)\frac{\Delta x}{2} \right\} \frac{U_{i+1} - 2U_i + U_{i-1}}{\Delta x^2} \tag{23}$$

This kind of generalization gets exact nodal solutions in a similar way to the scalar case. We identify the numerical diffusive matrix as

$$K^{\mathrm{num}} = A \frac{\Delta x}{2} \psi(K^{-1}A\,\Delta x) \tag{24}$$

similar to that obtained in the scalar case (see Eq. (11)). $K^{-1}A\,\Delta x$ acts as a matrix of Pèclet numbers. In the SUPG context, the numerical diffusive matrix is written as

$$K^{\mathrm{num}} = A\tau A , \quad \text{with} \quad \tau = \frac{\Delta x^2}{2}\,\phi(K^{-1}A\,\Delta x)K^{-1} \tag{25}$$

where the ϕ function is associated to the ψ one and defined as

$$\phi(x) = \psi(x)/x \tag{26}$$

(see Fig. 1). Note that, as $\psi(0) = 0$, ϕ is a regular function in $x = 0$. We conclude showing that the weight function is

$$W_i^{\mathrm{T}} = N_i^{\mathrm{T}} + P_i^{\mathrm{T}} = N_i^{\mathrm{T}} + N_{i,x}^{\mathrm{T}}A\tau \tag{27}$$

Note that the contraction of the perturbation term with the advective one produces a diffusive term of the type $N_{i,x}^{\mathrm{T}}A\tau A N_{i,x}$.

2.4. Multidimensional advective–diffusive scalar equation

In this subsection we show the extension of (11) to several spatial dimensions. A usual extension to the multi-dimensional scalar case:

$$a \cdot \nabla u = k\,\Delta u \tag{28}$$

is to take

$$k' = kI + \tau a a^{\mathrm{T}} \tag{29}$$

with

$$\tau = (\Delta x/2|a|)\psi(\Delta x|a|/k) \tag{30}$$

assuming a uniform and orthogonal grid. (The extension to a general grid includes a transformation to the master element.) The numerical features involved are well known and we will not repeat the details here [7].

2.5. Extension to multidimensional advective–diffusive system

Expression (30) can be put for the advection-dominated limit ($\mathrm{Pe} \to \infty$) as

$$\tau^{-2} = \tau_x^{-2} + \tau_y^{-2} + \tau_z^{-2} , \quad \tau_x = \Delta x/2|a_x| \tag{31}$$

which can be regarded as a composition of critical times in stability analysis. Then, the extension to multi-dimensional advective–diffusive systems we propose is

$$K_{ij}^{\mathrm{num}} = A_i \tau A_j \tag{32}$$

with

$$\tau^{-p} = \tau_x^{-p} + \tau_y^{-p} + \tau_z^{-p} \tag{33}$$

where

$$\tau_i = (\Delta x_i^2/2)\phi(K_{ii}^{-1}A_i \, \Delta x_i)K_{ii}^{-1} \qquad i = 1, 2, 3 \tag{34}$$

and p is a positive number (see Section 4 with respect to our particular choice of p).

It can be shown that, in the advection-dominated limit, Eq. (34) reduces to the scheme proposed by Mallet [9], with the only difference that the absolute value of the Jacobian matrices must be interpreted in the matrix of K_{ii}. Both schemes are equivalent in the following sense: they give a set of m uncoupled, full-upwinded equations for the eigen-components, but on a different basis.

2.6. Solution of the eigenvalue problem for the incompressible Navier–Stokes equations

In the most general case, the eigenvalue decomposition implicit in (34) must be performed via standard numerical routines. However, in order to reduce the computational cost, it is highly recommended to resort to analytical expressions, when it is possible. In this section we show that this is actually the case for the 3D incompressible Navier–Stokes equations, where the eigenvalue problem reduces to a quadratic equation. For the 3D compressible Navier–Stokes equations, the problem reduces to a cubic and can be solved analytically via the Cardano solution, but the expressions will not be given here.

The incompressible Navier–Stokes equations are

$$\begin{aligned}
(u \cdot \nabla)u &= -\nabla p + \nu \, \Delta u + f \\
\nabla \cdot u &= 0
\end{aligned} \tag{35}$$

where u is the velocity vector, f a body force source term and p the pressure (scaled by the constant density ρ, so that p has units of $[p] = m^2/s^2$). This can be cast in the form of a multi-dimensional advective diffusive system as in Eqs. (1) and (5) if we put

$$U = \begin{bmatrix} u \\ p \end{bmatrix}, \qquad A_j k_j = \begin{bmatrix} (u \cdot k)I & k \\ k^{\mathrm{T}} & 0 \end{bmatrix}, \qquad K_{jl} = \delta_{jl}\begin{bmatrix} \nu I & 0 \\ 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} f \\ 0 \end{bmatrix} \tag{36}$$

k is an arbitrary *wave number* vector.

As we have mentioned (after Eq. (16)), the method described so far is not applicable to the Navier–Stokes system since the corresponding K_{ii} matrices are singular in this case. For this reason the eigenvalue problem (17) is degenerated and the limit value of its eigenvalues is undetermined. The system is regularized adding a small positive quantity ϵ to the null diagonal element of the K_{ii} matrices. We stress the fact that this parameter ϵ is only introduced to compute the weight functions for the SUPG method, leaving the equations unaltered. The influence of this parameter on the precision and stability of the scheme will be analyzed later, for the particular case of the Stokes equations.

Taking $i = 1$ the argument of the ϕ function in (34) is

$$\Delta x \, \tilde{K}_{xx}^{-1}A_x = \Delta x \begin{bmatrix} 0 & 1/\epsilon & 0 & 0 \\ 1/\nu & u/\nu & 0 & 0 \\ 0 & 0 & u/\nu & 0 \\ 0 & 0 & 0 & u/\nu \end{bmatrix} \tag{37}$$

and its eigenvalues are simply

$$\lambda_{1,2} = \mathrm{Re}_{\Delta x}, \quad \lambda_{3,4} = 1/2(\mathrm{Re}_{\Delta x} \pm \sqrt{\mathrm{Re}_{\Delta x}^2 + 4 \, \Delta x^2/\nu\epsilon}) \tag{38}$$

where $\mathrm{Re}_{\Delta x} = u \, \Delta x/\nu$ is a Reynolds number based on the x component of the velocity and the size of the element, and $\tilde{K}_{ij}$ stands for the regularized diffusive Jacobian:

$$\tilde{K}_{jl} = \delta_{jl}\begin{bmatrix} \nu I & 0 \\ 0 & \epsilon \end{bmatrix}, \tag{39}$$

The eigenvalue decomposition for the other directions is similar.

3. Analysis of the stabilizing terms for the incompressibility condition in the Stokes equations

In what follows, we will show what kind of scheme is obtained for the Stokes equations, and we will compare it to other types of stabilization techniques. The regularization via the ϵ parameter as described in the first paragraph of Section 2.6 applies here also. The Stokes equations and their advective–diffusive-system form are the same as in (35) and (36), but setting $u = 0$ in the advective term $(u \cdot \nabla)u \equiv 0$ in (35) and in the advective Jacobians in (36).

The eigen-system decomposition, for the x direction, is

$$K_{xx}^{-1}A_x = S_x A_x S_x^{-1} \tag{40}$$

with

$$S_x = \begin{bmatrix} -\sqrt{\epsilon/\nu} & 0 & 0 & \sqrt{\epsilon/\nu} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \quad A_x = \begin{bmatrix} -1/\sqrt{\epsilon\nu} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\sqrt{\epsilon\nu} \end{bmatrix} \tag{41}$$

τ_x is computed from (34), and the final expression is

$$\tau_x = \mathrm{diag}\left\{ \frac{h\xi}{2}\sqrt{\frac{\epsilon}{\nu}}; \frac{h^2}{12\nu}; \frac{h^2}{12\nu}; \frac{h\xi}{2}\sqrt{\frac{\nu}{\epsilon}} \right\} \tag{42}$$

where $\xi = \psi(h/\sqrt{\epsilon\nu})$ and $\mathrm{diag}\{a, b, c \ldots\}$ is a diagonal matrix with diagonal entries $a, b, c \ldots$, while, for the other directions, a similar calculation gives

$$\tau_y = \mathrm{diag}\left\{ \frac{h^2}{12\nu}; \frac{h\xi}{2}\sqrt{\frac{\epsilon}{\nu}}; \frac{h^2}{12\nu}; \frac{h\xi}{2}\sqrt{\frac{\nu}{\epsilon}} \right\}$$

$$\tau_z = \mathrm{diag}\left\{ \frac{h^2}{12\nu}; \frac{h^2}{12\nu}; \frac{h\xi}{2}\sqrt{\frac{\epsilon}{\nu}}; \frac{h\xi}{2}\sqrt{\frac{\nu}{\epsilon}} \right\} \tag{43}$$

The expression for τ is obtained from (33) (assuming $p = 1$ for simplicity):

$$\tau = \mathrm{diag}\{\alpha, \alpha, \alpha, \beta\}, \quad \alpha = \frac{\sqrt{\epsilon}h^2\xi}{2h\sqrt{\nu} + 24\nu\sqrt{\epsilon}\xi}, \quad \beta = \frac{h\xi}{6}\sqrt{\frac{\nu}{\epsilon}} \tag{44}$$

and the K_{ij}^{num}, from expression (32), are

$$K_{ij}^{\mathrm{num}}k_i k_j = \begin{bmatrix} \beta k k^{\mathrm{T}} & 0 \\ 0 & \alpha k^2 \end{bmatrix} \tag{45}$$

Note that in the limit $\epsilon \to 0$ we have $\xi \to 1$, $\alpha \to h/2\sqrt{\epsilon/\nu}$ and $\beta \to h/6\sqrt{\nu/\epsilon}$. Finally, the discretized system is

$$\begin{bmatrix} \nu K + \beta G & Q \\ -Q^{\mathrm{T}} & \alpha H \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} F \\ \alpha S \end{bmatrix} \tag{46}$$

where the standard finite element matrices and interpolations are assumed:

$$p(x) \approx \sum_\mu p_\mu N_{p\mu}(x)$$

$$u_i(x) \approx \sum_\mu u_{i\mu} N_{u,\mu}(x)$$

$$K_{i\mu,j\nu} = \delta_{ij}\left\{ \int_\Omega N_{u\mu,k}N_{u\nu,k} \right\} d\Omega$$

$$Q_{\mu,j\nu} = \int_\Omega N_{p\mu}N_{u\nu,j} \, d\Omega$$

$$H_{\mu,\nu} = \int_\Omega N_{p\mu,k} N_{p\nu,k} \, d\Omega$$

$$G_{i\mu,j\nu} = \int_\Omega N_{u\mu,i} N_{u\nu,j} \, d\Omega$$

$$F_{i\mu} = \int_\Omega f_i N_{u,\mu} \, d\Omega$$

$$S_\mu = \int_\Omega f_i N_{p\mu,i} \, d\Omega \tag{47}$$

These expressions include the possibility of different interpolation spaces $\mathcal{V}_p = \mathrm{span}\{N_{p\mu}\}$, $\mathcal{V}_u = \mathrm{span}\{N_{u\mu}\}$ for pressure and velocity but, in fact, we will always work with $\mathcal{V}_p = \mathcal{V}_u$ since we are interested in equal-order interpolations.

K is a viscous-rigidity matrix and Q, $-Q^T$ are centered discrete approximations to the divergence and gradient operators. For the Galerkin formulation the G, H operators are not present and, due to the null diagonal block in the matrix, spurious oscillations are present if the pair $(\mathcal{V}_p, \mathcal{V}_u)$ fails to satisfy the Brezzi–Babúska condition. In practice, it has been found that a necessary condition to the Brezzi–Babúska one is that the order of interpolation for u must be one degree higher than that of p. However, some a priori natural combinations like $Q2/Q1$ (biquadratic velocities/discontinuous bilinear pressures) are not stable. It has been found that reducing the pressure space has a stabilizing effect, for instance eliminating the xy term in the bilinear approximation for pressure for the $Q2/Q1$ element gives the $Q2/P1$ element (biquadratic velocities/discontinuous linear pressures) which is stable. But this procedure is far to be general, and it is very difficult to find stable combinations, particularly in 3D. On the other hand, enriching the velocity space has also a stabilizing effect, and some combinations can be stabilized by the introduction of bubble functions (for instance the $P2^+/P1$ element, (triangle with biquadratic velocities plus bubble, and discontinuous linear pressures). Recently, a great deal of research work [1–3] has been carried out to circumvent this restriction with equal order combinations, via the addition of stabilizing terms like those used to stabilize advective–diffusive problems. Finally, it has been shown that not only the stabilization via bubble addition is equivalent to the addition of a stabilizing term, but also the converse is true: Under certain restrictions any stabilizing term is equivalent to the addition of certain (a priori not known) 'virtual' bubbles [10].

On the other hand, it has been shown [4] that all the stabilized formulations are based on the introduction (with different justifications varying from one author to another) of a stabilizing term proportional to the discrete version of Δp in the continuity equation, like the αH term in our formulation, see Eq. (46). Furthermore, Sampaio's formulation is the closest to ours, since it introduces a term equivalent to a discrete $\nabla(\nabla \cdot u)$ in the momentum equation, as the βG in ours. This term represents physically a bulk viscosity term. It can also be found by a straightforward application of the Taylor–Galerkin method [4].

Most of the stabilized methods are basically of the form (46). The case $\alpha = \beta = 0$ corresponds to the Galerkin non-stabilized case. The case $\beta = 0$, $\alpha = \alpha' h^2 / 2\nu$ corresponds to the stabilized method of Hughes et al. [1]. Sampaio's scheme [2] (in the Stokes regime and for linear elements) is obtained with $\alpha = 2mh^2/\nu$, $\beta = \lambda m u^2 h^2/4\nu$, and λ, m are O(1) constants, defined in his work. System (46) corresponds to the centered Galerkin discretization of the PDE system:

$$-\nu\,\Delta u - \beta\nabla(\nabla\cdot u) + \nabla p = f$$
$$\nabla\cdot u - \alpha\,\Delta p = -\alpha(\nabla\cdot f) \tag{48}$$

and it can be obtained from the original Stokes system by adding $-\beta$ times the gradient of the continuity equation to the momentum equation, and $-\alpha$ times the divergence of the momentum equation to the continuity equation. Since the systems are equivalent, the resulting stabilized scheme has the same precision (in the sense of order of truncation error) independently from α and β.

With respect to the influence of α and β on stability, we have made a deep discrete Fourier analysis

which will not be presented in detail here, but its relevant results. As usual, all quantities are decomposed in plane waves like $p(x) = \hat{p}\, e^{ik \cdot x}$, and response functions of the form:

$$\frac{\partial \hat{p}}{\partial \hat{f_i}} = G_{p,f_i}(k, \alpha, \beta, \ldots) \tag{49}$$

are computed. This last response function is very important since it is the one which is singular for non-stable formulations at particular wave-number vectors corresponding to 'checkerboard modes'. For the instance of the $Q1/Q1$ (bilinear velocities, bilinear continuous pressures interpolation) the checkerboard modes are

$$k_{\text{check}} = \{\epsilon_x, \epsilon_y, \epsilon_z\}\pi/h\,, \quad \epsilon_i = 0, 1 \tag{50}$$

For these modes the variations of the quantities in one direction are of the form $\{\ldots, 1, -1, 1, -1, \ldots\}$.

Taking $k = (k, 0, 0)$, for simplicity, the resulting 1D discrete system is

$$- \alpha\, \frac{p_{i-1} - 2p_i + p_{i-1}}{h^2} + \frac{u_{i+1} - u_{i-1}}{2h} = -\alpha\, \frac{f_{i+1} - f_{i-1}}{2h}$$

$$\frac{p_{i+1} - p_{i-1}}{2h} - (\beta + \nu)\, \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = \frac{f_{i+1} + 4f_i + f_{i-1}}{6} \tag{51}$$

and the response function is

$$(1/h)G_{p,f} = \frac{-i\, \sin(kh)\{(1/3)[2 + \cos(kh)] + (\gamma/4)\sin^2(kh/2)\}}{\gamma \sin^4(kh/2) + \sin^2(kh)} \tag{52}$$

where

$$\gamma = 16\alpha(\nu + \beta)/h^2 = \frac{16\alpha\nu}{h^2} + \frac{16\alpha\beta}{h^2} = I + II \tag{53}$$

is a global stability parameter.

In Fig. 2 we can see the effect of γ on the response function. For $\gamma > 0$ the singularity is removed at $k = \pi$. However, for too small $\gamma(\gamma \ll 1)$ an undesirable peak in the response function occurs near $k = \pi$. It is seen that for $\gamma \approx 1$ the peak is completely removed and a monotone response curve is obtained. Numerical results from Hughes et al. [1] show that for the lid-driven cavity flow benchmark with a global stability parameter γ ranging from 0.8 to 8, oscillations are absent, whereas for $\gamma < 0.08$ oscillations exist. These results are in perfect agreement with the previous discrete analysis.

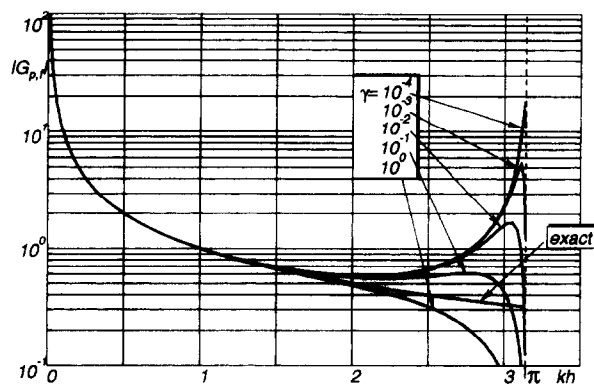As regards the proposed method, the global stability parameter is

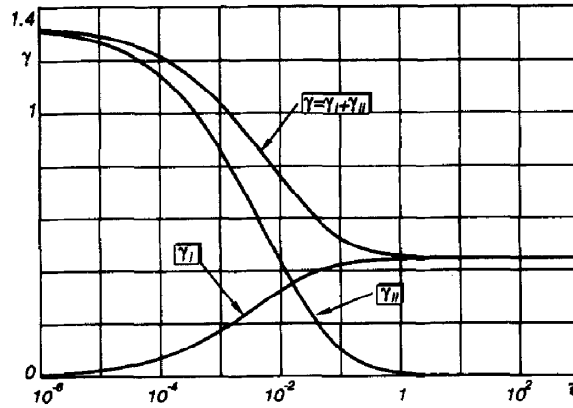Fig. 2. Response curves for several global stability parameters γ.

Fig. 3. Global stability parameter for the proposed method.

$$\gamma = \frac{4}{3}\,\xi\,\frac{\xi + 6\sqrt{\tilde{\epsilon}}}{1 + 12\sqrt{\tilde{\epsilon}}\,\xi} \tag{54}$$

where

$$\tilde{\epsilon} = \epsilon\nu/h^2 \,, \qquad \xi = \psi(1/\sqrt{\tilde{\epsilon}}) \tag{55}$$

$\tilde{\epsilon}$ is a new non-dimensional stability parameter. The relationship between γ and $\tilde{\epsilon}$ is depicted in Fig. 3 and also each of the terms in (53). We can see that, no matter the choice of $\tilde{\epsilon}$, the algorithm is stable ($\gamma > 0.4$). Note that for very small $\tilde{\epsilon}$ the first term $16\alpha\nu/h^2$ in (53) is not enough to stabilize, but the second term $16\alpha\beta/h^2$ is. So that, in this case, the bulk-viscosity stabilization term is indispensable. Keeping the α term small is a nice feature since the term $-\alpha\,\Delta p$ changes the nature of the system, for instance: the number of boundary conditions to be specified. Bad boundary conditions are absorbed in boundary layers whose thickness grows with α and, so, a small α guarantees a good behavior of the scheme even if additional boundary conditions are not good. Note that $\beta = \nu/(6\sqrt{\tilde{\epsilon}}) = O(1)$ as a function of the mesh size, but, as it was previously shown, it does not affect precision.

In the numerical examples $\tilde{\epsilon}$ has been always chosen as $\tilde{\epsilon} = 1$.

4. Application to the near-incompressible Navier–Stokes equations

One of the aims of stabilizing equal order interpolations is to obtain algorithms that work simultaneously in compressible and incompressible regime. We have applied the concepts developed in the previous sections to the Navier–Stokes compressible equations (1)–(4), and, now, we focus on the behavior of the resulting code in the near-compressible regime, i.e. $M \to 0$.

In this case the expression for the numerical diffusion matrix is much more complicated and will not be shown here. However, it is important to mention that the 1D-eigenvalue problem for each τ_i (Eq. (34)) can be solved analytically. Once the τ_x, τ_y, τ_z matrices are obtained, the global intrinsic time matrix can be calculated easily from Eq. (33) if the choice $p = 1$ is assumed, otherwise an eigenvalue decomposition for τ must be done which, as far as we know, cannot be obtained analytically. Since numerical decomposition routines would be too expensive we have always chosen $p = 1$. In non-orthogonal grids, the transformation to the master element at each Gauss point yields the correct direction and mesh size.

The resulting non-linear system is solved via pseudo-temporal explicit integration. In order to accelerate the convergence to the steady state, a local time step, selected according to stability analysis, is used. We will not give the details of these issues here, but they are similar to the techniques we presented in previous works. In general, this scheme becomes more competitive for large 3D problems, and we are working now in several ways to improve the convergence, via local preconditioning [11,12].

In the following sections, several examples are shown for internal flows with $M < 0.05$ and Reynolds number up to 250. These results are to be compared to the respective incompressible solutions. From the examples, it can be concluded that the algorithm stabilizers correctly the pressure under near-incompressible conditions. Furthermore, it is worth to note that one of the Pèclet numbers in each eigen-system is a local Reynolds number based on the local projection of the velocity and the element size. This shows that the scheme stabilizes the advective part of the Navier–Stokes equation automatically. Results in this direction will be shown in a future work. Results for lower Mach numbers have not been obtained due to bad conditioning of the pseudo-temporal scheme.

5. Numerical examples

The first example is the famous test of the square, lid-driven cavity. Here, we present results for Re = 1, 40, 100. In Fig. 4 we show the problem description. The mesh for the cases for Re = 1, 40 is the one from Fig. 5, it has 20 × 20 elements refined near the solid walls, while the mesh for the Re = 100 case has the same amount of elements, but without refinement (see Fig. 6).
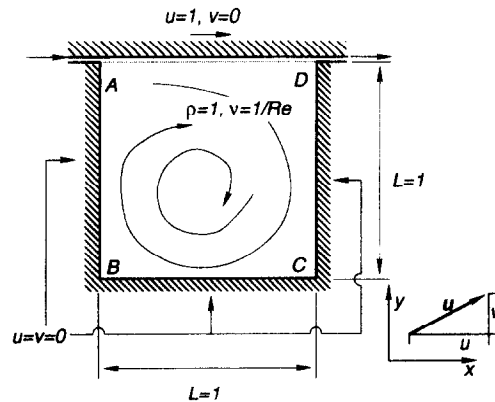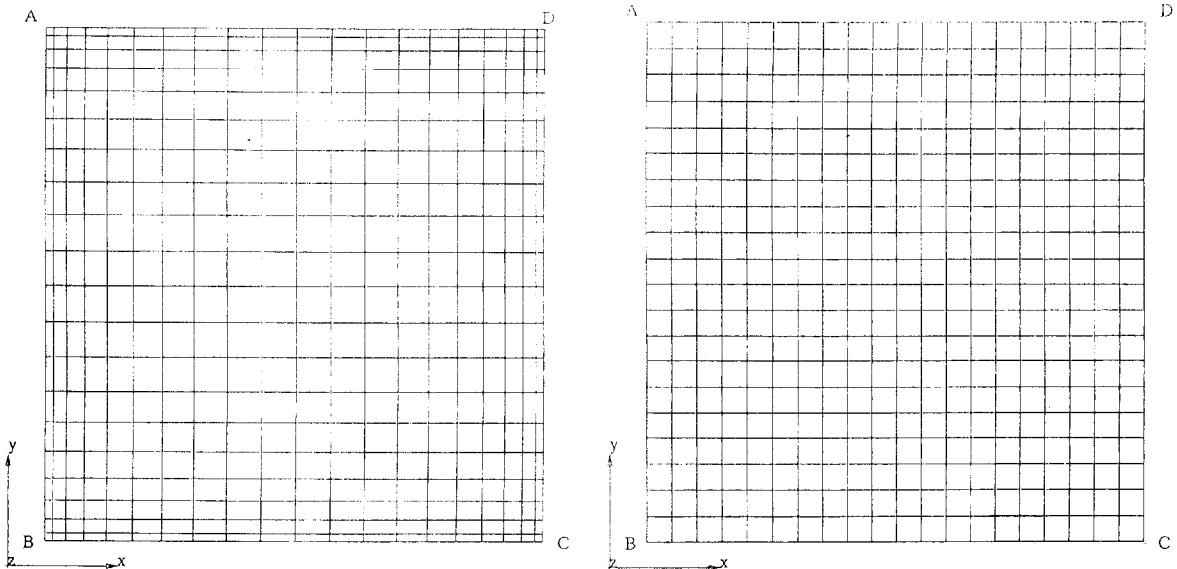

Fig. 4. Lid-driven square-cavity—Problem description.

Fig. 5. Lid-driven square-cavity—20 × 20 non-homogeneous mesh.

Fig. 6. Lid-driven square-cavity—20 × 20 homogeneous mesh.

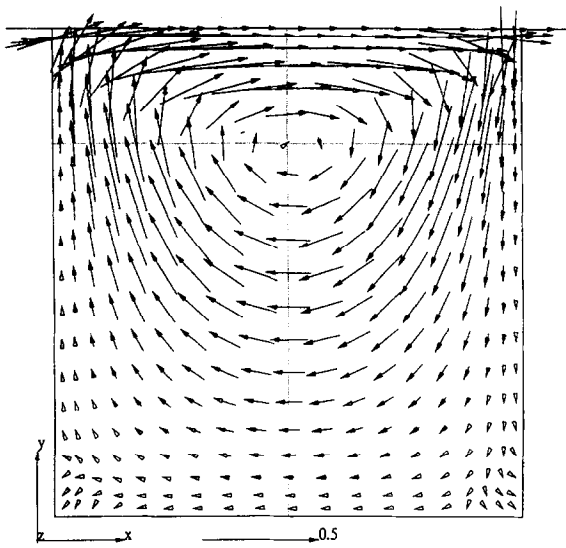Fig. 7. Lid-driven square-cavity—Velocity field for Re = 1.
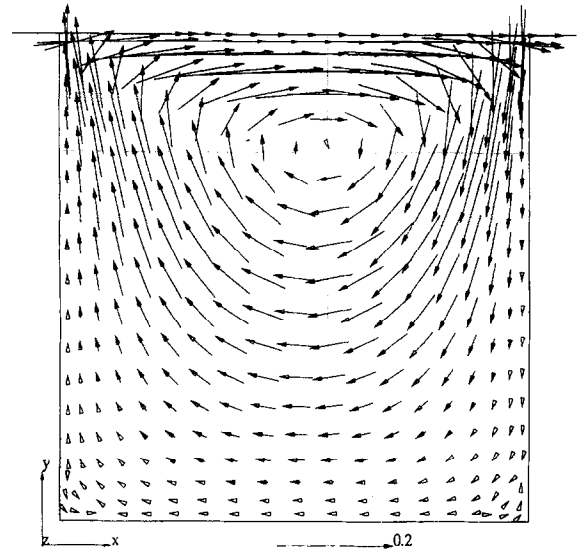
Fig. 8. Lid-driven square-cavity—Velocity field for Re = 40.

In Figs. 7–9 we show the velocity field for each Reynolds number and we include in it the reference position of the principal vortex center obtained from Schreiber et al. [13]. The results we present here are in good agreement with the referenced ones. We do not present results for secondary vortices, but they can be solved with finer meshes.

Moreover, we solved another test-problem similar to the square-cavity one, consisting in a square-cavity with two internal obstacles. This problem is referenced as the multiply-connected lid-driven square-cavity [14]. In Fig. 10 we show the geometry of the problem, and we can see the position of the solid obstacles. They are centered at $(x, y) = (\frac{1}{3}, \frac{1}{2})$ for the first one, and $(\frac{2}{3}, \frac{1}{2})$ for the other one, with a thickness of $\frac{1}{15}$ and a height of $\frac{1}{3}$. We solve for Re = 0, 50, 250 and we show in Figs. 11–13 the streamlines we got from the numerical results. They were compared to published results by Lippke et
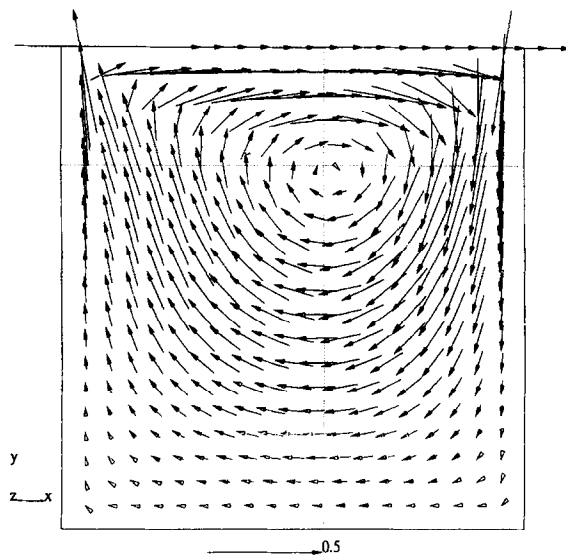

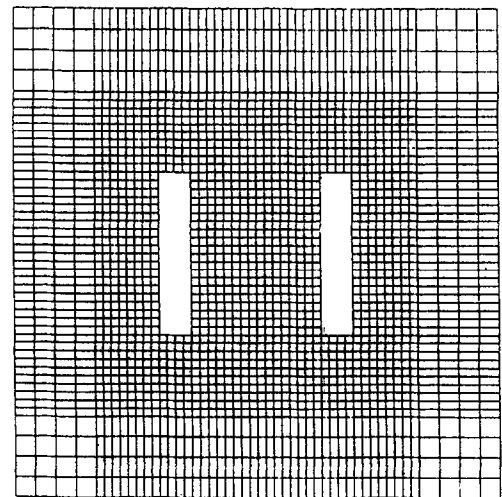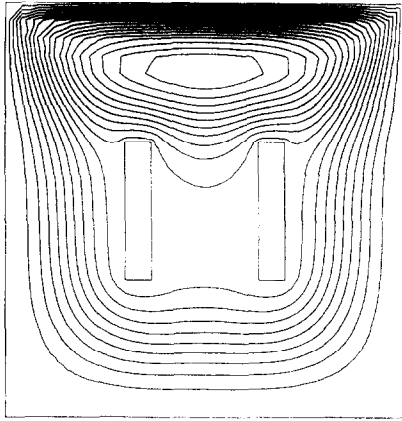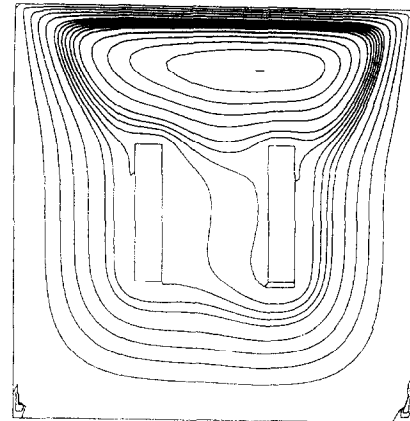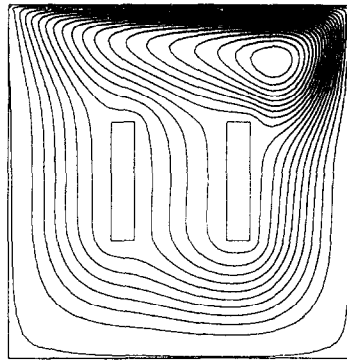Fig. 9. Lid-driven square-cavity—Velocity field for Re = 100.

Fig. 10. Multiply-connected lid-driven square-cavity—Problem description.

Fig. 11. Multiply-connected lid-driven square-cavity—Streamlines for Re = 0.

Fig. 12. Multiply-connected lid-driven square-cavity—Streamlines for Re = 50.

Fig. 13. Multiply-connected lid-driven square-cavity—Streamlines for Re = 250.

al. using a similar quantity of nodes (≈ 3600) and we conclude that they are also in good agreement with them.

In all cases we used Mach numbers below 0.05, and we selected ϵ according to Eq. (55) with $\tilde{\epsilon} = 1$.

6. Conclusions

We present a Petrov–Galerkin algorithm for the solution of the Navier–Stokes equations. Theoretical analysis on the Stokes equation shows that it generates stabilization terms which are already found in several works in the literature. The stabilization term is applied to the whole system of equations, covering both the mass and momentum equations with a unified criterion. The numerical results show that the algorithm has good properties of stability and precision with respect to the incompressibility condition and advection terms. The proposed formulation allows an easy mathematical explanation of how the incompressibility condition is stabilized without introducing a special term for it.

Acknowledgment

References

[1] T.J.R. Hughes, L. Franca and M. Balestra, A new finite element formulation for CFD: V. Circumventing the Babuska–Brezzi condition: A stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Comput. Methods Appl. Mech. Engrg. 59 (1986) 85–99.

[2] S.L. Frey, L.P. Franca and R. Sampaio, Stabilized methods for the incompressible Navier–Stokes flow, in: H. Alder et al., eds., Numerical Methods in Engineering and Applied Sciences, Proc. International Congress on Numerical Methods in Engineering and Applied Sciences, Concepción, Chile, 16–20 November, 1992.

[3] O.C. Zienkiewicz, J. Szmelter and J. Peraire, Compressible and incompressible flow: an algorithm for all seasons, Comput. Methods Appl. Mech. Engrg. 78 (1990) 105–121.

[4] O.C. Zienkiewicz and J. Wu, Incompressibility without tears—How to avoid restrictions on mixed formulations, Int. J. Numer. Methods Engrg. 32 (1991/1992) 1189–1204.

[5] M. Storti, N. Nigro and S. Idelsohn, Stabilizing equal-order interpolations for mixed formulations of Navier–Stokes equations, in: E. Oñate, ed., Numerical Methods in Engineering and Applied Sciences, Proc. 8th International Conference on Finite Elements in Fluids, Barcelona, Spain, 20–24 November, 1993 (Pineridge Press and CIMNE, Barcelona, 1993) 194–202.

[6] T.J.R. Hughes and T.E. Tezduyar, Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations, Comput. Methods Appl. Mech. Engrg. 45 (1984) 217-284.

[7] T.J.R. Hughes, M. Mallet and A. Mizukami, A new finite element method for CFD: II. Beyond SUPG, Comput. Methods Appl. Mech. Engrg. 54 (1986) 341–355.

[8] T.J.R. Hughes and M. Mallet, A new finite element method for CFD: IV. A discontinuity-capturing operator for multidimensional advective–diffusive systems, Comput. Methods Appl. Mech. Engrg. 58 (1986) 329–336.

[9] M. Mallet, A finite element method for CFD, Ph.D Thesis, Stanford University, November 1985.

[10] C. Baiocchi, F. Brezzi and L. Franca, Virtual bubbles and Galerkin-least-squares type methods (Ga.L.S.), Comput. Methods Appl. Mech. Engrg. 105 (1993) 125–141.

[11] C. Baumann, M. Storti and S. Idelsohn, A Petrov–Galerkin technique for the solution of transonic and supersonic flows, Comput. Methods Appl. Mech. Engrg. 95 (1992) 49–70.

[12] M. Storti, C. Baumann and S. Idelsohn, A preconditioning mass matrix to accelerate the convergence to the steady Euler solutions using explicit schemes, Int. J. Numer. Methods Engrg. 34 (1992) 519–541.

[13] R. Schreiber and H.P. Keller, Driven cavity flows by efficient numerical techniques, J. Comput. Phys. 49 (1983) 310–333.

[14] A. Lippke and H. Wagner, Numerical solution of the Navier–Stokes equations in multiply domains connected, Comput. Fluids 20(1) (1991) 19–27.