INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **47**:271–323
Published online 3 December 2004 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.808

A segregated method for compressible flow computation Part I: isothermal compressible flows

Guillermo Hauke[1,*,†], Aitor Landaberea[2], Iñaki Garmendia[2] and Javier Canales[3]

[1]*Departamento de Mecánica de Fluidos, Centro Politécnico Superior, C/Maria de Luna 3, 50.018 Zaragoza, Spain*
[2]*Product Engineering Dpt. Fundación INASMET, Mikeletegi 2, 20.009 San Sebastian, Spain*
[3]*Department of Mechanical Engineering, Universidad del País Vasco, Alda. Urquijo, s/n 48.013 Bilbao, Spain*

SUMMARY

Traditionally, coupled methods have been employed for the computation of compressible flows, whereas segregated methods have been preferred for the computation of incompressible flows. Compared to coupled methods, segregated solvers present the advantage of reduced computer memory and CPU time requirements, although at the cost of an inferior robustness. Therefore, in a series of papers we present unified computational techniques to compute compressible and incompressible flows with segregated stabilized methods. The proposed algorithms have an increased robustness compared to existing techniques, while possessing additional benefits such as employing standard pressure boundary conditions. In this first part, the thermodynamics of isothermal, thermally perfect compressible flows is set up in the framework of symmetric systems and the corresponding segregated algorithms are introduced. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: compressible flow; stabilized finite element methods; segregated methods

1. INTRODUCTION

In the computational fluid dynamics community, compressible flows have been typically solved with coupled solvers, whereas incompressible flows have been tackled with segregated techniques. The reasons are varied.

On the one hand, the fluid thermodynamics of compressible flows couples very tightly the dependent variables, mainly the thermodynamic variables such as the pressure and temperature. In incompressible flows, however, this coupling seems weaker.

*Correspondence to: G. Hauke, Departamento de Mecánica de Fluidos, Centro Politécnico Superior, C/Maria de Luna 3, 50.018 Zaragoza, Spain.
†E-mail: ghauke@unizar.es

Contract/grant sponsor: Ministerio de Educacion y Ciencia, Spain
Contract/grant sponsor: University of Basque Country, Spain; contract/grant number: Project 1/UPV/EHU 00145.345-EA-8246/2000

Received 21 January 2004
Revised 6 August 2004
Accepted 14 October 2004

On the other hand, in incompressible flows, the indefinite structure of the left hand side finds in segregated formulations an escape to easy inversion. The literature and the variety of solutions on this topic is very rich, including artificial compressibility methods [1], penalty methods [2, 3], augmented Lagrangian methods [4–6], projection methods and fractional momentum methods [1, 7–10] and mixed methods [11–13]. For further information, the interested reader can consult [14–16] and references therein.

Another avenue for flow computation is the general class of stabilized methods, including Petrov–Galerkin methods. Even when these methods are applied to incompressible flows, the equations have been solved segregatedly (see References [17–20] and references therein). An exception to this can be found in the work of References [21, 22], where a coupled method is proposed for non-isothermal incompressible flows.

While coupled solvers typically are deemed as more robust, segregated solvers may have computational advantages such as an inferior memory and CPU time demand, and a higher modularity and flexibility in order to incorporate new physical phenomena modeled by additional partial differential equations. These advantages are even more dramatic in the solution of large scale problems and in the application to parallel codes. However, some segregated solvers based on pressure Poisson-like equations require non-trivial pressure boundary conditions.

In the past, there has been a few attempts to develop segregated formulations for compressible flows, but success has only be partial. For instance, the characteristic Galerkin method is proposed in References [23–27], but their formulation requires to solve fractional momentum equations and therefore, two momentum passes per iteration. Another difficulty of this formulation is coping with strong discontinuities and imposing boundary conditions through the various fractional steps.

Therefore, in a series of papers we develop and explore segregated formulations that can be used for both, compressible and incompressible flows and lack the above drawbacks. In particular, the variational framework employed in this work is based on stabilized formulations, which have been successfully applied to the solution of compressible and incompressible flows with coupled solvers [17, 28–40].

The point of departure of the series of papers is this one, where segregated methods for isothermal thermally perfect compressible flows are proposed. The variational formulation is set up for the set of pressure primitive variables. For that purpose, the thermodynamics of this kind of flows is developed in the framework of symmetric systems. The corresponding generalized entropy function and fluxes are made explicit and the stability principle developed. It will be shown that, in the incompressible limit, the stability principle leads to the classical mechanical energy stability. Examples will be shown for very low Mach number flows, as well as for high Mach number cases with strong discontinuities. Then, this paper will be followed by a second one, where methods for fully compressible flows, robust enough to tackle strong discontinuities, will be shown.

2. THE EQUATIONS FOR COMPRESSIBLE ISOTHERMAL FLOW

By definition, an isothermal flow is that in which the temperature is constant. Therefore, the energy equation becomes redundant, and the thermodynamic state of a general divariant substance can be determined solely by one thermodynamically independent variable. As a con-

sequence, the number of dependent variables necessary to describe an isothermal compressible flow is reduced by one compared to those needed for compressible flow.

In Cartesian co-ordinates the set of the Navier–Stokes equations (continuity plus momentum equations) can be written in conservative form as

$$\mathbf{U}_{,t} + \mathbf{F}^{\text{adv}}_{i,i} = \mathbf{F}^{\text{diff}}_{i,i} + \mathbf{S} \tag{1}$$

In the equation above, $\mathbf{U}$ represents the vector of conservation variables, $\mathbf{F}^{\text{adv}}_i$ is the advective flux in the ith-direction, $\mathbf{F}^{\text{diff}}_i$ is the diffusive flux in the ith-direction, and $\mathbf{S}$ is the source vector. The inferior comma denotes partial differentiation and the summation convention on repeated indices is applied throughout.

In three-dimensional Cartesian co-ordinates, the above vectors are

$$\mathbf{U} = \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{Bmatrix} = \rho \begin{Bmatrix} 1 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{2}$$

$$\mathbf{F}^{\text{adv}}_i = \rho u_i \begin{Bmatrix} 1 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} + p \begin{Bmatrix} 0 \\ \delta_{1i} \\ \delta_{2i} \\ \delta_{3i} \end{Bmatrix} \tag{3}$$

$$\mathbf{F}^{\text{diff}}_i = \begin{Bmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ \tau_{3i} \end{Bmatrix} \tag{4}$$

$$\mathbf{S} = \rho \begin{Bmatrix} 0 \\ b_1 \\ b_2 \\ b_3 \end{Bmatrix} \tag{5}$$

where u_i are the Cartesian velocity components, ρ the fluid density, p the thermodynamic pressure, δ_{ij} the Kronecker delta, τ_{ij} the viscous stress tensor and b_i the body force per unit mass.

Using any well defined set of variables $\mathbf{Y}$, it is possible to linearize (1) as

$$\mathbf{A}_0 \mathbf{Y}_{,t} + \mathbf{A}_i \mathbf{Y}_{,i} = (\mathbf{K}_{ij} \mathbf{Y}_{,j})_{,i} + \mathbf{S} \tag{6}$$

where $\mathbf{A}_0 = \mathbf{U}_{,\mathbf{Y}}$, $\mathbf{A}_i = \mathbf{F}^{\text{adv}}_{i,\mathbf{Y}}$ is the ith Euler Jacobian matrix, and $\mathbf{K} = [\mathbf{K}_{ij}]$ is the diffusivity matrix obeying $\mathbf{K}_{ij}\mathbf{Y}_{,j} = \mathbf{F}^{\text{diff}}_i$.

For the choice of pressure primitive variables,

$$\mathbf{Y} = \begin{Bmatrix} p \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{7}$$

with p the pressure and u_i the Cartesian velocity components. The explicit expression of the corresponding matrices and vectors can be found in Appendix A.

3. GENERALIZED ENTROPY FUNCTION AND SYMMETRIC FORM

Symmetric forms of (1) and (6) are those in which the coefficient matrices enjoy the properties [41]

i. $\tilde{\mathbf{A}}_0$ is symmetric, positive-definite,
ii. $\tilde{\mathbf{A}}_i$ is symmetric,
iii. $\tilde{\mathbf{K}} = [\tilde{\mathbf{K}}_{ij}]$ is symmetric, positive-semidefinite

where the tilde indicates that the above matrices stem from the symmetric form.

It is known that symmetric forms of the equations are linked to non-linear stability principles through the so-called generalized entropy function, which at the same time engenders the entropy variables [42, 43]. In the case of general compressible flows, the generalized entropy is just the physical entropy [44, 29].

Appropriately defined finite element methods can inherit their discrete stability from the above physical entropy-like principle. In this way, the robustness of the method is a result of mimmicing the physics of nature.

Thus, for isothermal, thermally perfect fluids, (i.e. fluids in which $e = e(T)$), the generalized entropy function can be defined as

$$\mathscr{H}(\mathbf{U}) = \frac{1}{2}\rho|\mathbf{u}|^2 - \rho T(s - s_0) \tag{8}$$

where T is the absolute temperature, s the physical entropy and s_0 the reference entropy. Note that for an isothermal flow, T is constant.

For an isothermal simple compressible substance, the equation of state takes on the form

$$p = p(\rho) \tag{9}$$

Furthermore, for a thermally perfect fluid, the Gibbs equation yields,

$$T\,\mathrm{d}s = \mathrm{d}e + p d\frac{1}{\rho} = -\frac{p}{\rho^2}\,\mathrm{d}\rho \tag{10}$$

As $p = p(\rho)$, also the entropy can be written solely as a function of $s(\rho)$ and

$$T \frac{\mathrm{d}s}{\mathrm{d}\rho} = -\frac{p}{\rho^2} \tag{11}$$

From $\mathscr{H}$, following [42, 43] the generalized entropy function gives rise to the set of generalized entropy variables,

$$\mathbf{V}^{\mathrm{T}} = \frac{\partial \mathscr{H}(\mathbf{U})}{\partial \mathbf{U}} \tag{12}$$

resulting in

$$\mathbf{V} = \begin{Bmatrix} \dfrac{p}{\rho} - T(s - s_0) - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{13}$$

Under the above assumptions, the entropy variables can also be written as a function of μ, the electro-chemical potential per unit mass,

$$\mathbf{V} = \begin{Bmatrix} \mu - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{14}$$

with $e = Ts_0$ since

$$\mu = h - Ts = e + \frac{p}{\rho} - Ts \tag{15}$$

Note that the change of variables $\mathbf{U} \mapsto \mathbf{V}$ induces the Hessian matrix $\mathscr{H}_{,\mathbf{UU}}$ given by

$$\tilde{\mathbf{A}}_0^{-1} = \frac{1}{\rho} \begin{bmatrix} \dfrac{1}{\rho\beta_T} + |\mathbf{u}|^2 & -u_1 & -u_2 & -u_3 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & \text{symm} & & 1 \end{bmatrix} \tag{16}$$

which induces a symmetric, positive-definite $\tilde{\mathbf{A}}_0^{-1}$ if the density $\rho > 0$ and the isothermal compressibility coefficient $\beta_T > 0$. Therefore, $\mathscr{H}(\mathbf{U})$ is a convex function of $\mathbf{U}$ and the above change of variables is well defined.

Likewise, the matrices $\tilde{\mathbf{A}}_i$ are symmetric and $\tilde{\mathbf{K}} = [\tilde{\mathbf{K}}_{ij}]$ are symmetric positive-semidefinite (see Appendix A).

4. THE GENERALIZED ENTROPY (IN)EQUALITY AND THE STABILITY PRINCIPLE

It is known that symmetric forms of the equations are linked to stability principles via the entropy variables. The stability principle is a combination of the conservation laws and can be obtained by the dot product

$$\mathbf{V}\cdot(\mathbf{U}_{,t}+\mathbf{F}^{\mathrm{adv}}_{i,i}-\mathbf{F}^{\mathrm{diff}}_{i,i}-\mathbf{S})=0 \tag{17}$$

Applying the chain rule to the diffusive term and substituting the definition of the fluxes, one gets

$$\mathbf{V}\cdot\tilde{\mathbf{A}}_0\mathbf{V}_{,t}+\mathbf{V}\cdot\mathbf{F}^{\mathrm{adv}}_{i,i}-(\mathbf{V}\cdot\mathbf{F}^{\mathrm{diff}}_{i})_{,i}-\mathbf{V}\cdot\mathbf{S}=-\mathbf{V}_{,i}\cdot\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j} \tag{18}$$

Let us examine in detail the different terms present in Equation (18)

$$\begin{aligned}\mathbf{V}\cdot\tilde{\mathbf{A}}_0\mathbf{V}_{,t}&=\mathscr{H}_{,\mathbf{U}}\mathbf{U}_{,t}=\mathscr{H}_{,t}\\&=\left[\frac{1}{2}\rho|\mathbf{u}|^2-\rho T(s-s_0)\right]_{,t}\end{aligned} \tag{19}$$

$$\begin{aligned}\mathbf{V}\cdot\mathbf{F}^{\mathrm{adv}}_{i,i}&=\left(\frac{p}{\rho}-T(s-s_0)-\frac{1}{2}|\mathbf{u}|^2\right)[\rho u_i]_{,i}+u_j[\rho u_i u_j]_{,i}+u_i p_{,i}\\&=\left[\rho u_i\left(\frac{1}{2}|\mathbf{u}|^2-T(s-s_0)+\frac{p}{\rho}\right)\right]_{,i}\end{aligned} \tag{20}$$

$$\begin{aligned}\mathbf{V}\cdot\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}&=\mathbf{V}\cdot\mathbf{F}^{\mathrm{diff}}_{i}\\&=u_j\tau_{ij}\end{aligned} \tag{21}$$

$$\begin{aligned}\mathbf{V}_{,i}\cdot\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}&=\mathbf{V}_{,i}\cdot\mathbf{F}^{\mathrm{diff}}_{i}\\&=u_{j,i}\tau_{ji}\\&=\Upsilon(\mathbf{u},\mathbf{u})\geqslant 0\end{aligned} \tag{22}$$

$$\mathbf{V}\cdot\mathbf{S}=\rho u_i b_i \tag{23}$$

In the above equations, the definition of the viscous dissipation function $\Upsilon(\mathbf{u},\mathbf{u})\geqslant 0$ has been used. Gathering all the contributions yields

$$\begin{aligned}&\left[\frac{1}{2}\rho|\mathbf{u}|^2-\rho T(s-s_0)\right]_{,t}\\&\quad+\left[\rho u_i\left(\frac{1}{2}|\mathbf{u}|^2-T(s-s_0)+\frac{p}{\rho}\right)\right]_{,i}-[u_j\tau_{ij}]_{,i}-\rho u_i b_i=-\Upsilon(\mathbf{u},\mathbf{u})\\&\quad\leqslant 0\end{aligned} \tag{24}$$

where entropy fluxes are seen to be

$$\sigma_i = \rho u_i \left(\frac{1}{2}|\mathbf{u}|^2 - T(s - s_0) + \frac{p}{\rho} \right) \tag{25}$$

Therefore, integration of Equation (24) over the domain Ω gives rise to a Clausius–Duhem like inequality,

$$\begin{aligned} \int_\Omega (\mathscr{H}_{,t} + [\sigma_i]_{,i} - [u_j \tau_{ij}]_{,i} - \rho u_i b_i)\, \mathrm{d}\Omega &= -\frac{1}{T} \int_\Omega \Upsilon(\mathbf{u}, \mathbf{u})\, \mathrm{d}\Omega \\ &\leqslant 0 \end{aligned} \tag{26}$$

This implies that under appropriate boundary conditions and body source term, the generalized entropy $\mathscr{H}(\mathbf{U})$ is a bounded function. Since $\mathscr{H}(\mathbf{U})$ is a convex function of the conservation variables $\mathbf{U}$, the conservation variables are themselves bounded. Therefore, this is the non-linear stability principle for isothermal compressible flows.

4.1. Further insight into the entropy (in)equality

The internal energy equation for a substance can be written as

$$[\rho e]_{,t} + [\rho u_i e]_{,i} = -\, p u_{i,i} + u_{i,j} \tau_{ij} - q_{i,i} + \dot{q}_v \tag{27}$$

with e the specific internal energy, q_i the heat flux and $\dot{q}_v$ a volumetric heat source. For a thermally perfect, isothermal fluid, the internal energy is constant and the heat flux is zero, so for a vanishing volumetric heat source, the above equation can be simplified into

$$\begin{aligned} p u_{i,i} &= u_{i,j} \tau_{ij} \\ &= \Upsilon(\mathbf{u}, \mathbf{u}) \end{aligned} \tag{28}$$

Thus, in this case the viscous dissipation induces a volumetric expansion, which, in turn, according to the continuity equation, decreases the density of the fluid particle.

Now the mechanical energy equation can be written as

$$\left[\frac{1}{2} \rho |\mathbf{u}|^2 \right]_{,t} + \left[u_i \frac{1}{2} \rho |\mathbf{u}|^2 \right]_{,i} = -\,[p u_i]_{,i} + [u_i \tau_{ij}]_{,j} + p u_{i,i} - u_{i,j} \tau_{ij} + \rho u_i b_i \tag{29}$$

Subtracting this equation to Equation (24), yields

$$\begin{aligned} -[\rho T(s - s_0)]_{,t} - [\rho u_i (T(s - s_0))]_{,i} &= -\Upsilon(\mathbf{u}, \mathbf{u}) + [u_{i,j} \tau_{ij}]_{,i} - p u_{i,i} \\ &= -\, p u_{i,i} \\ &= -\Upsilon(\mathbf{u}, \mathbf{u}) \\ &\leqslant 0 \end{aligned} \tag{30}$$

where in the last step, Equation (28) has been introduced. Thus, integrating over the domain of interest for constant temperature,

$$\int_\Omega ([\rho(s-s_0)]_{,t} + [\rho u_i(s-s_0)]_{,i})\,\mathrm{d}\Omega = \frac{1}{T}\int_\Omega \Upsilon(\mathbf{u},\mathbf{u})\,\mathrm{d}\Omega \geqslant 0 \tag{31}$$

resulting in the second law of thermodynamics.

Therefore, the generalized entropy inequality is in this case a combination of the entropy and the mechanical energy equations.

5. PARTICULAR CASES

5.1. Thermally perfect, isothermal perfect gas

Let us take a fluid which follows the isothermal perfect gas law

$$p = K\rho \tag{32}$$

with $K>0$. The isothermal compressibility coefficient is in this case

$$\beta_T = \frac{1}{\rho K} = \frac{1}{p} > 0 \tag{33}$$

Furthermore, the entropy can be obtained from the Gibbs relation

$$T\,\mathrm{d}s = -K\frac{\mathrm{d}\rho}{\rho} = -K\frac{\mathrm{d}p}{p} \tag{34}$$

whose integration yields

$$T(s-s_0) = -K\ln\frac{\rho}{\rho_0} \tag{35}$$

where s_0 is the entropy at a density ρ_0.

So for a perfect isothermal gas, the generalized entropy function given by (8) is simplified to

$$\mathscr{H}(\mathbf{U}) = \frac{1}{2}\rho|\mathbf{u}|^2 + \rho K\ln\frac{\rho}{\rho_0} \tag{36}$$

which yields the following entropy variables:

$$\mathbf{V} = \begin{Bmatrix} K\left(1+\ln\dfrac{\rho}{\rho_0}\right) - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{37}$$

5.2. Slightly compressible liquid

The equation of state for a liquid which is slightly compressible can be written as

$$\rho = \rho_0 + \rho_0 \beta (p - p_0) \tag{38}$$

with $\beta > 0$. The isothermal compressibility coefficient is in this case

$$\beta_T = \frac{\rho_0 \beta}{\rho} > 0 \tag{39}$$

Furthermore, the entropy can be obtained integrating

$$T\,\mathrm{d}s = -\frac{p}{\rho^2}\,\mathrm{d}\rho = -\frac{1}{\rho^2 \beta}\left(\beta p_0 + \frac{\rho}{\rho_0} - 1\right)\mathrm{d}\rho \tag{40}$$

which yields

$$T(s - s_0) = -\frac{1}{\rho_0 \beta}\left[\ln\frac{\rho}{\rho_0} + \frac{\rho_0}{\rho} - 1\right] + \frac{p_0}{\rho_0}\left[\frac{\rho_0}{\rho} - 1\right] \tag{41}$$

with s_0 the entropy at the density ρ_0.

Therefore, the generalized entropy function (8) is

$$\mathscr{H}(\mathbf{U}) = \frac{1}{2}\rho|\mathbf{u}|^2 + \frac{\rho}{\rho_0 \beta}\left[\ln\frac{\rho}{\rho_0} + \frac{\rho_0}{\rho} - 1\right] - \frac{p_0}{\rho_0}\left[\frac{\rho_0}{\rho} - 1\right] \tag{42}$$

which engenders the following generalized entropy function:

$$\mathbf{V} = \begin{Bmatrix} \dfrac{p_0}{\rho_0} + \dfrac{1}{\rho_0 \beta}\ln\dfrac{\rho}{\rho_0} - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{43}$$

5.3. Incompressible fluid

For a strictly incompressible liquid, the density is constant, and the equation of state can be simply written as

$$\rho = \rho_0 \tag{44}$$

The isothermal compressibility coefficient is in this case

$$\beta_T \to 0 \tag{45}$$

limit that can be obtained from the isothermal perfect gas model with

$$K \to \infty \tag{46}$$

As in the previous cases, the entropy can be obtained integrating the Gibbs relation,

$$T\,\mathrm{d}s = \mathrm{d}e - \frac{p}{\rho^2}\,\mathrm{d}\rho = 0 \tag{47}$$

implying that the entropy is constant:

$$s = s_0 \tag{48}$$

So in the incompressible limit, the generalized entropy function tends to

$$\mathscr{H}(\mathbf{U}) = \frac{1}{2}\,\rho|\mathbf{u}|^2 \tag{49}$$

and likewise, the entropy variables are obtained as the limit of the isothermal fluid case, yielding,

$$\mathbf{V} = \begin{Bmatrix} \dfrac{p}{\rho} - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{50}$$

Note that the entropy has disappeared in the above two relations because it is constant and therefore it does not play any role regarding stability. Thus, the classical kinetic energy stability principle for incompressible fluids is recovered.

6. STUDY OF VARIOUS THERMODYNAMIC LIMITS

6.1. Incompressible limit of the formulation

In the incompressible limit, the density becomes constant and, therefore, the isothermal compressibility coefficient tends to zero,

$$\beta_T \to 0 \tag{51}$$

which in turn implies for the equations of state mentioned previously that

$$K \to \infty \tag{52}$$

$$\beta \to 0 \tag{53}$$

Then, it can be seen that the incompressible limit of the equations based on entropy variables or pressure primitive variables is well defined.

For example, the first Euler Jacobian for the entropy variables is

$$\lim_{\beta_T\to 0}\tilde{\mathbf{A}}_1=\lim_{\beta_T\to 0}\rho^2\beta_T\begin{bmatrix} u_1 & \dfrac{1}{\rho\beta_T}+u_1^2 & u_1u_2 & u_1u_3 \\ & u_1\left(\dfrac{3}{\rho\beta_T}+u_1^2\right) & u_2\left(\dfrac{1}{\rho\beta_T}+u_1^2\right) & u_3\left(\dfrac{1}{\rho\beta_T}+u_1^2\right) \\ & & u_1\left(\dfrac{1}{\rho\beta_T}+u_2^2\right) & u_1u_2u_3 \\ & \text{symm} & & u_1\left(\dfrac{1}{\rho\beta_T}+u_3^2\right) \end{bmatrix}$$

$$=\rho\begin{bmatrix} 0 & 1 & 0 & 0 \\ & 3u_1 & u_2 & u_3 \\ & & u_1 & 0 \\ & \text{symm} & & u_1 \end{bmatrix} \tag{54}$$

and for primitive variables

$$\lim_{\beta_T\to 0}\mathbf{A}_1=\lim_{\beta_T\to 0}\begin{bmatrix} \rho\beta_Tu_1 & \rho & 0 & 0 \\ \rho\beta_Tu_1^2+1 & 2\rho u_1 & 0 & 0 \\ \rho\beta_Tu_1u_2 & \rho u_2 & \rho u_1 & 0 \\ \rho\beta_Tu_1u_3 & \rho u_3 & 0 & \rho u_1 \end{bmatrix}$$

$$=\begin{bmatrix} 0 & \rho & 0 & 0 \\ 1 & 2\rho u_1 & 0 & 0 \\ 0 & \rho u_2 & \rho u_1 & 0 \\ 0 & \rho u_3 & 0 & \rho u_1 \end{bmatrix} \tag{55}$$

As a consequence, the incompressible limit is well defined in both cases, and both, entropy variables and pressure primitive variables are well suited to compute isothermal incompressible flows. For the consequences of these results, the reader is referred to Reference [22].

6.2. The isothermal limit of the general divariant fluid case

In Chalot *et al.* [45] and Hauke and Hughes [21, 22] the flow equations for a compressible general divariant fluid were derived for various sets of variables. For a general divariant fluid, all the thermodynamic variables can be written as a function of the thermal expansion

(or isobaric compressibility) coefficient and the isothermal compressibility coefficient,

$$\alpha_p = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{56}$$

$$\beta_T = \frac{1}{\rho}\left(\frac{\partial \rho}{\partial p}\right)_T \tag{57}$$

The isothermal limit of the above equations can be obtained as

$$\alpha_p \to 0 \tag{58}$$

with β_T finite. From the Mayer equation, it can be deduced that in this case the specific heats are equal,

$$c_p = c_v + \frac{\alpha_p^2 T}{\rho \beta_T} = c_v \tag{59}$$

As a consequence, the isothermal limit of the matrices for the general divariant fluid based on entropy variables $\tilde{\mathbf{A}}_i^{\mathrm{df}}$ can be related to the isothermal matrices $\tilde{\mathbf{A}}_i$ according to

$$\mathrm{block}_{4\times 4}(\tilde{\mathbf{A}}_0^{\mathrm{df}}) = T\tilde{\mathbf{A}}_0 \tag{60}$$

$$\mathrm{block}_{4\times 4}(\tilde{\mathbf{A}}_i^{\mathrm{df}}) = T\tilde{\mathbf{A}}_i \tag{61}$$

$$\mathrm{block}_{4\times 4}(\tilde{\mathbf{K}}_i^{\mathrm{df}}) = T\tilde{\mathbf{K}}_i \tag{62}$$

$$\mathrm{block}_{4\times 4}(\tilde{\mathbf{A}}_0^{\mathrm{df}})^{-1} = \frac{1}{T}\tilde{\mathbf{A}}_0^{-1} \tag{63}$$

where the operator $\mathrm{block}_{4\times 4}$ extracts the first 4 rows and 4 columns, resulting from eliminating the total energy equation and the corresponding unknown [46].

Note that the inverse of the Riemannian metric tensor follows a different transformation, that is,

$$\mathrm{block}_{4\times 4}((\tilde{\mathbf{A}}_0^{\mathrm{df}})^{-1}) \neq \frac{1}{T}\,\tilde{\mathbf{A}}_0^{-1} \tag{64}$$

Therefore, in order to obtain the matrices for isothermal flow, it is not possible to use directly all the 4×4 blocks of the general divariant fluid matrices. This has a direct impact on the formulation of the stabilized finite element method, specially on the discontinuity capturing term, which is presented in the next section (see Reference [46] for more details).

7. STABILIZED FINITE ELEMENT METHOD

The stabilized finite element method combines the SUPG/GLS method [17, 31–35] with the time-discontinuous space–time method [47]. Following the lines in References [21, 22], the

method based on entropy variables $\mathbf{V}$ can be then transformed to the variables of interest $\mathbf{Y}$ (for an elaboration on this, see also Reference [48]).

The computational domain Ω is discretized in n_{el} elements of area Ω^e and boundary Γ^e. In the absence of source terms and for linear shape functions the SUPG and GLS methods coincide.

Consider a space–time domain, where the time interval $I =]0, T[$ is subdivided into N intervals $I_n =]t_n, t_{n+1}[$, $n = 0, 1, \ldots, N-1$. We define for each time interval $Q_n = \Omega \times I_n$ and $P_n = \Gamma \times I_n$, where Ω is the spatial domain and Γ its boundary. Finally, the 'slab' Q_n is decomposed into elements Q_n^e, $e = 1, 2, \ldots, (n_{\mathrm{el}})_n$.

The weak form is: Within each Q_n, $n = 0, 1, \ldots, N-1$, find $\mathbf{Y} \in \mathcal{S}_{\mathbf{Y}}$ such that $\forall\ \mathbf{W} \in \mathcal{V}_{\mathbf{Y}}$:

$$\begin{aligned}
&\int_{Q_n} (-\mathbf{W}_{,t} \cdot \mathbf{U}(\mathbf{Y}) - \mathbf{W}_{,i} \cdot \mathbf{F}_i^{\mathrm{adv}}(\mathbf{Y}) + \mathbf{W}_{,i} \cdot \mathbf{K}_{ij} \mathbf{Y}_{,j} - \mathbf{W} \cdot \mathbf{S})\, \mathrm{d}Q \\
&\quad + \int_{\Omega} (\mathbf{W}(t_{n+1}^-) \cdot \mathbf{U}(\mathbf{Y}(t_{n+1}^-)) - \mathbf{W}(t_n^+) \cdot \mathbf{U}(\mathbf{Y}(t_n^-)))\, \mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{(n_{\mathrm{el}})_n} \int_{Q_n^e} (\mathscr{L}^{\mathrm{T}} \mathbf{W}) \cdot \boldsymbol{\tau} (\mathscr{L} \mathbf{Y} - \mathbf{S})\, \mathrm{d}Q \\
&\quad + \sum_{e=1}^{(n_{\mathrm{el}})_n} \int_{Q_n^e} \nu^h g^{ij} \mathbf{W}_{,i} \cdot \mathbf{A}_0 \mathbf{Y}_{,j}\, \mathrm{d}Q \\
&\quad = \int_{P_n} \mathbf{W} \cdot (-\mathbf{F}_i^{\mathrm{adv}}(\mathbf{Y}) + \mathbf{F}_i^{\mathrm{diff}}(\mathbf{Y})) n_i\, \mathrm{d}P
\end{aligned} \tag{65}$$

The first and last integrals constitute the Galerkin terms expressed as a function of the variables $\mathbf{Y}$, written in conservative form to ensure that the weak solution contains the correct Rankine–Hugoniot relations.

Time-discontinuous Galerkin methods enforce weakly the continuity in time of the solution across time-slabs, via the second integral, which has been defined by means of the upper and lower limits,

$$W(t_n^{\pm}) = \lim_{\varepsilon \to 0^{\pm}} W(t_n + \varepsilon) \tag{66}$$

The least-squares contribution is written in terms of the differential operator $\mathscr{L}$, which is given by

$$\mathscr{L} = \mathbf{A}_0 \frac{\partial}{\partial t} + \mathbf{A}_i \frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij} \frac{\partial}{\partial x_j} \right) \tag{67}$$

and $\mathscr{L}^{\mathrm{T}}$, which is defined by

$$\mathscr{L}^{\mathrm{T}} = \mathbf{A}_0^{\mathrm{T}} \frac{\partial}{\partial t} + \mathbf{A}_i^{\mathrm{T}} \frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij}^{\mathrm{T}} \frac{\partial}{\partial x_j} \right) \tag{68}$$

Note that when entropy variables are used, $\tilde{\mathcal{L}} = \tilde{\mathcal{L}}^{\mathrm{T}}$ because of the symmetry of the coefficient matrices and the symmetric form is recovered.

7.1. Stabilizing matrix

The stabilizing matrix is defined according to Reference [22]

$$\boldsymbol{\tau} = \mathbf{Y}_{,\mathbf{V}} \tilde{\boldsymbol{\tau}} \tag{69}$$

where $\tilde{\boldsymbol{\tau}}$ is the stabilizing matrix for entropy variables.

The adopted expression for the stabilizing matrix stems from the work in Reference [49] and references therein. In that paper, two simple approximations to the eigenvalue problem are presented, which will be used and explored in this work.

7.1.1. Diagonal tau. The first definition of the intrinsic time-scale parameter is a diagonal matrix, consisting of applying the one-dimensional theory to each component of the Navier–Stokes equations. It can also be interpreted as an extension of the incompressible tau [50, 18] to compressible flows, where the eigenvalue u has been replaced by the largest eigenvalue in the streamwise direction, i.e. $u + c$. Although there is also information travelling orthogonally to the streamline direction, in the expressions proposed below only the streamline information is retained, while information orthogonal to the streamline is dropped. This is reasonable for high speed flows, since u predominates over the disturbances propagation speed c. The proposed definition is then,

$$\boldsymbol{\tau}_d = \mathrm{diag}(\tau_c, \tau_m, \tau_m, \tau_m) \tag{70}$$

where

$$\tau_c = \min\left(\frac{|\mathbf{u}| h^e}{2}\right) \tag{71}$$

$$\tau_m = \min\left(\frac{\Delta t}{2\rho}, \frac{h^e}{2\rho(|\mathbf{u}| + c)} + \frac{h^e}{2\rho|\mathbf{u}|}, \frac{m^e (h^e)^2}{4\mu}\right) \tag{72}$$

The above expressions have been written in terms of h^e, a measure of the element size in the streamline direction; Δt, the time step; μ and κ, the viscosity and the thermal conductivity; and $m^e = \min(1/3, 2C^e)$, where C^e is a constant arising from an inverse estimate of the second derivatives of the shape functions. For linear elements $C^e = \infty$ and $m^e = 1/3$ (see Reference [18] for details).

The element size along the streamline direction h^e is defined as

$$h^e = \frac{2}{\sqrt{\alpha_s}} \tag{73}$$

where

$$\alpha_s = \xi_{i,s} \xi_{i,s} \tag{74}$$

with s the streamline direction and ξ the local element co-ordinates.

Remark
For stationary flows, the Δt terms are omitted.

Remark
In general, this simple matrix is less robust than the standard one given by the full eigenvalue problem, and so, requires a stronger discontinuity capturing operator, such as the Hughes–Mallet [51].

7.1.2. Non-diagonal tau. This definition exploits the relation for tau between pressure and entropy variables (69), which can be rearranged to give

$$\boldsymbol{\tau}_{nd} = \mathbf{Y}_{,\mathbf{U}}\hat{\boldsymbol{\tau}} \tag{75}$$

where $\hat{\boldsymbol{\tau}}$ is the stabilizing matrix for conservation variables. Thus, for conservation variables a diagonal matrix is assumed

$$\hat{\boldsymbol{\tau}} = \mathrm{diag}(\hat{\tau}_c, \hat{\tau}_m, \hat{\tau}_m, \hat{\tau}_m) \tag{76}$$

and

$$\hat{\tau}_c = \min\left(\frac{\Delta t}{2}, \frac{1}{\lambda^e}\right) \tag{77}$$

$$\hat{\tau}_m = \min\left(\frac{\Delta t}{2}, \frac{1}{\lambda^e}, \frac{\rho m^e (h^e_{\mathrm{diff}})^2}{4\mu}\right) \tag{78}$$

The eigenvalue λ^e has been approximated by

$$\frac{1}{\lambda^e} = \frac{h^e}{2(|\mathbf{u}| + c)} \tag{79}$$

The diffusive length scale has been taken as $h^e_{\mathrm{diff}} = h^e$.

Finally, transformation (75) is applied to obtain a non-diagonal matrix for primitive variables.

Remark
For the computation of stationary flows, Δt must be dropped.

Remark
In this case, the CPU-time savings are due to a less computational effort involved in the calculation of the stabilization matrix. Even though in theory the CPU-time savings for the diagonal tau are potentially larger, in practice the present choice gives a faster convergence rate.

Remark
The non-diagonal tau is much robuster than the diagonal one.

7.2. Discontinuity capturing operator

The fourth integral is the so-called *discontinuity capturing operator*, which is written in terms of the contravariant metric tensor g^{ij}, defined by

$$g^{ij} = [\xi_{k,i}\xi_{k,j}]^{-1} \tag{80}$$

where ξ_k, $k = 1, 2, 3$, are the local spatial element co-ordinates, (i.e. not including the time-dimension). The artificial diffusion v^h is a scalar function of the residual, namely, $\mathscr{L}\mathbf{Y} - \mathbf{S}$. For example, the Hughes–Mallet [51] and quadratic [35] versions are defined for the variables $\mathbf{Y}$ as [21, 22]

$$v^h_{\mathrm{HM}} = \max\left(0, \left[\frac{(\mathscr{L}\mathbf{Y} - \mathbf{S}) \cdot \tilde{\mathbf{A}}_0^{-1}(\mathscr{L}\mathbf{Y} - \mathbf{S})}{g^{ij}\mathbf{Y}_{,i} \cdot \mathbf{A}_0^{\mathrm{DC}}\mathbf{Y}_{,j}}\right]^{1/2} - \frac{(\mathscr{L}\mathbf{Y} - \mathbf{S}) \cdot \tilde{\boldsymbol{\tau}}(\mathscr{L}\mathbf{Y} - \mathbf{S})}{g^{ij}\mathbf{Y}_{,i} \cdot \mathbf{A}_0^{\mathrm{DC}}\mathbf{Y}_{,j}}\right) \tag{81}$$

$$v^h_{\mathrm{quad}} = 2\,\frac{(\mathscr{L}\mathbf{Y} - \mathbf{S}) \cdot \tilde{\boldsymbol{\tau}}\ (\mathscr{L}\mathbf{Y} - \mathbf{S})}{g^{ij}\mathbf{Y}_{,i} \cdot \mathbf{A}_0^{\mathrm{DC}}\mathbf{Y}_{,j}} \tag{82}$$

where

$$\mathbf{A}_0^{\mathrm{DC}} = \mathbf{V}_{,\mathbf{Y}}^{\mathrm{T}}\tilde{\mathbf{A}}_0\mathbf{V}_{,\mathbf{Y}} = \mathbf{V}_{,\mathbf{Y}}^{\mathrm{T}}\mathbf{A}_0 \tag{83}$$

The matrix $\tilde{\boldsymbol{\tau}}$ is obtained from the version $\boldsymbol{\tau}$ for the variables $\mathbf{Y}$ as follows:

$$\tilde{\boldsymbol{\tau}} = \tilde{\mathbf{A}}_0^{-1}\mathbf{A}_0\boldsymbol{\tau} \tag{84}$$

8. SEGREGATED FORMULATION

Substitution of the finite element spaces in the above weak form leads to a coupled system of non-linear equations, which can be written as a residual depending on $\mathbf{y}_{(n)}$ and $\mathbf{y}_{(n+1)}$, the nodal unknowns at time t_n and t_{n+1}, respectively. That is,

$$\mathbf{R}(\mathbf{y}_{(n+1)}; \mathbf{y}_{(n)}) = 0 \tag{85}$$

An interesting family of methods to drive the residual to zero are predictor multi-corrector algorithms [17]. These methods update the unknowns iteratively, like a Newton–Raphson method, where the update is computed from the linear system,

$$\begin{aligned} \mathbf{M}^* \Delta\mathbf{y} &= -\mathbf{R} \\ \mathbf{y}^{\mathrm{new}} &= \mathbf{y}^{\mathrm{old}} + \Delta\mathbf{y} \end{aligned} \tag{86}$$

Typically, the matrix $\mathbf{M}^*$ can be the consistent tangent (i.e. the Jacobian of the residual with respect to the unknowns),

$$\mathbf{M}^* = \frac{\partial \mathbf{R}}{\partial \mathbf{y}_{(n+1)}} \tag{87}$$

or an approximation of the above matrix.

Thus, let us split the vector of dependent variables into two parts that gather the unknowns of the continuity and momentum equations,

$$\mathbf{Y} = \begin{Bmatrix} \mathbf{Y}_\rho \\ \mathbf{Y}_u \end{Bmatrix} \tag{88}$$

For instance, for pressure primitive variables

$$\mathbf{Y}_\rho = \{p\} \tag{89}$$

$$\mathbf{Y}_u = \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{90}$$

The nodal unknowns and the equations can be ordered in a similar manner,

$$\mathbf{y} = \begin{Bmatrix} \mathbf{y}_\rho \\ \mathbf{y}_u \end{Bmatrix} \tag{91}$$

$$\mathbf{R} = \begin{Bmatrix} \mathbf{R}_\rho \\ \mathbf{R}_u \end{Bmatrix} \tag{92}$$

where the equations of mass conservation are set in $\mathbf{R}_\rho$ and those of the momentum equations in $\mathbf{R}_u$.

Then, the Newton–Raphson method can be divided into blocks as

$$\begin{bmatrix} \mathbf{M}_{\rho\rho} & \mathbf{M}_{\rho u} \\ \mathbf{M}_{u\rho} & \mathbf{M}_{uu} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{y}_\rho \\ \Delta \mathbf{y}_u \end{Bmatrix} = - \begin{Bmatrix} \mathbf{R}_\rho \\ \mathbf{R}_u \end{Bmatrix} \tag{93}$$

The explicit expressions of the blocks can be found in Appendix B.

Although traditionally coupled solvers have been used to tackle problem (93), the purpose of this paper is to investigate the application of segregated algorithms. Three strategies are proposed, whose analysis is presented next.

8.1. Diagonal iterative method

This simplest option in the resolution of system (93) consists of ignoring, along the iterative process, the coupling between the two types of unknowns, that is, the off-diagonal blocks of $\mathbf{M}$. This leads at each iteration to

$$\begin{aligned} \mathbf{M}_{\rho\rho}\Delta\mathbf{y}_\rho &= -\mathbf{R}_\rho \\ \mathbf{M}_{uu}\Delta\mathbf{y}_u &= -\mathbf{R}_u \end{aligned} \tag{94}$$

Thus, the solution of the coupled problem is replaced by two smaller problems sequentially solved. The final predictor multi-corrector algorithm is given in Algorithm 1, where i is the iteration counter, and

$$\mathbf{R}_\rho^{(i)} = \mathbf{R}_\rho(\mathbf{y}_{(n+1)}^{(i)}; \mathbf{y}_{(n)}) \tag{95}$$

$$\mathbf{R}_u^{(i)} = \mathbf{R}_u(\mathbf{y}_{(n+1)}^{(i)}; \mathbf{y}_{(n)}) \tag{96}$$

Algorithm 1. Predictor multi-corrector algorithm. Diagonal iterative method.

(Predictor phase)
$\mathbf{y}_\rho^{(0)} = \mathbf{y}_{\rho;(n)}$
$\mathbf{y}_u^{(0)} = \mathbf{y}_{u;(n)}$
(Multi-corrector phase)
for $i = 0,\ 1,\ \ldots,\ i_{\max} - 1$
  $\mathbf{M}_{\rho\rho}\Delta\mathbf{y}_\rho^{(i)} = -\mathbf{R}_\rho^{(i)}$
  $\mathbf{M}_{uu}\Delta\mathbf{y}_u^{(i)} = -\mathbf{R}_u^{(i)}$
  $\mathbf{y}_\rho^{(i+1)} = \mathbf{y}_\rho^{(i)} + \Delta\mathbf{y}_\rho^{(i)}$
  $\mathbf{y}_u^{(i+1)} = \mathbf{y}_u^{(i)} + \Delta\mathbf{y}_u^{(i)}$
$\mathbf{y}_{\rho(n+1)} = \mathbf{y}_\rho^{(i_{\max})}$
$\mathbf{y}_{u(n+1)} = \mathbf{y}_u^{(i_{\max})}$

8.2. Jacobi iterative method

The next method consists of taking into account the coupling between equations in an explicit manner, via a Jacobi iterative method. The coupling terms between mass-conservation and momentum equations are evaluated at the previous iteration, so its value is known before hand.

This corresponds to reordering equation (93) as

$$\begin{aligned} \mathbf{M}_{\rho\rho}\Delta\mathbf{y}_\rho &= -\mathbf{R}_\rho - \mathbf{M}_{\rho u}\Delta\mathbf{y}_u \\ \mathbf{M}_{uu}\Delta\mathbf{y}_u &= -\mathbf{R}_u - \mathbf{M}_{u\rho}\Delta\mathbf{y}_\rho \end{aligned} \tag{97}$$

and evaluating the right hand side at the values of the previous iteration (see Algorithm 2).

Algorithm 2. Predictor multi-corrector algorithm. Jacobi iterative method.

(Predictor phase)
$\mathbf{y}_\rho^{(0)} = \mathbf{y}_{\rho;(n)}$
$\mathbf{y}_u^{(0)} = \mathbf{y}_{u;(n)}$
(Multi-corrector phase)
for $i = 0,\ 1,\ \ldots,\ i_{\max} - 1$
$\mathbf{M}_{\rho\rho}\Delta\mathbf{y}_\rho^{(i)} = -\mathbf{R}_\rho^{(i)} - \mathbf{M}_{\rho u}\Delta\mathbf{y}_u^{(i-1)}$
$\mathbf{M}_{uu}\Delta\mathbf{y}_u^{(i)} = -\mathbf{R}_u^{(i)} - \mathbf{M}_{u\rho}\Delta\mathbf{y}_\rho^{(i-1)}$
$\mathbf{y}_\rho^{(i+1)} = \mathbf{y}_\rho^{(i)} + \Delta\mathbf{y}_\rho^{(i)}$
$\mathbf{y}_u^{(i+1)} = \mathbf{y}_u^{(i)} + \Delta\mathbf{y}_u^{(i)}$
$\mathbf{y}_{\rho(n+1)} = \mathbf{y}_\rho^{(i_{\max})}$
$\mathbf{y}_{u(n+1)} = \mathbf{y}_u^{(i_{\max})}$

8.3. Gauss–Seidel iterative method

Convergence of the Jacobi method can be accelerated using the Gauss–Seidel modification, where at each iteration, the most up-to-date values of the unknowns are used to compute the solution of the system. Again, this corresponds to reordering Equation (93) as in the Jacobi method (see Equation (97)) and evaluating the right hand side with the most up-to-date values of the present iteration (see Algorithm 3).

Algorithm 3. Predictor multi-corrector algorithm. Gauss–Seidel iterative method.

(Predictor phase)
$\mathbf{y}_\rho^{(0)} = \mathbf{y}_{\rho;(n)}$
$\mathbf{y}_u^{(0)} = \mathbf{y}_{u;(n)}$
(Multi-corrector phase)
for $i = 0,\ 1,\ \ldots,\ i_{\max} - 1$
$\mathbf{M}_{\rho\rho}\Delta\mathbf{y}_\rho^{(i)} = -\mathbf{R}_\rho^{(i)} - \mathbf{M}_{\rho u}\Delta\mathbf{y}_u^{(i-1)}$
$\mathbf{M}_{uu}\Delta\mathbf{y}_u^{(i)} = -\mathbf{R}_u^{(i)} - \mathbf{M}_{u\rho}\Delta\mathbf{y}_\rho^{(i)}$
$\mathbf{y}_\rho^{(i+1)} = \mathbf{y}_\rho^{(i)} + \Delta\mathbf{y}_\rho^{(i)}$
$\mathbf{y}_u^{(i+1)} = \mathbf{y}_u^{(i)} + \Delta\mathbf{y}_u^{(i)}$
$\mathbf{y}_{\rho(n+1)} = \mathbf{y}_\rho^{(i_{\max})}$
$\mathbf{y}_{u(n+1)} = \mathbf{y}_u^{(i_{\max})}$

8.4. Under and over relaxation

The convergence of the solution can be accelerated or slowed down by controlling the amount of change of the solutions,

$$\begin{aligned} \mathbf{y}_\rho^{(i+1)} &= \mathbf{y}_\rho^{(i)} + \alpha_\rho \Delta\mathbf{y}_\rho^{(i)} \\ \mathbf{y}_u^{(i+1)} &= \mathbf{y}_u^{(i)} + \alpha_u \Delta\mathbf{y}_u^{(i)} \end{aligned} \tag{98}$$

In some cases, it is necessary to limit the variation of the variables, precluding divergence at the cost of a stiffer tangent, i.e. $0 \leqslant \alpha_\rho,\ \alpha_u \leqslant 1$.

9. NUMERICAL EXAMPLES

In this section, the different versions of the segregated algorithm presented previously are evaluated in detail through their application to numerical examples. These examples cover a wide spectrum of isothermal steady state flows ranging from incompressible subsonic flows to compressible supersonic flows. The incompressible problems have been treated as nearly incompressible with a Mach number $M = 0.1$. All computations have been performed with bilinear quadrilateral elements with standard 2×2 Gaussian quadrature.

9.1. Incompressible flows

9.1.1. Poiseuille flow. The Poiseuille flow is a fully developed, laminar flow in a channel between two infinite parallel plates characterized by a quadratic velocity profile and a decreasing linear pressure field. This example serves to illustrate the difficulties which arise in the stabilized formulation when coarse meshes with low order elements are used. To show this, a unit square domain has been defined as shown in Figure 1 where a parabolic velocity profile has been specified at the inlet boundary, zero velocity along the top and bottom walls and a constant pressure $p_0 = 100$ at the outlet boundary which corresponds to a Mach number of $M = 0.1$. The Reynolds number based on the centreline velocity and the height of the channel is 1000.

The domain has been discretized into meshes of 5×5, 10×10 and 20×20 square elements. Figure 2 shows the velocity contours obtained with the coarsest and finest meshes which match

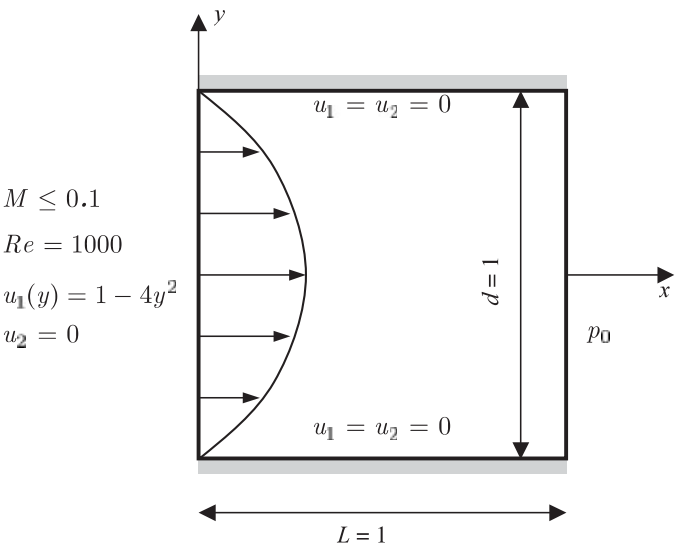

Figure 1. Poiseuille flow. Problem setup.

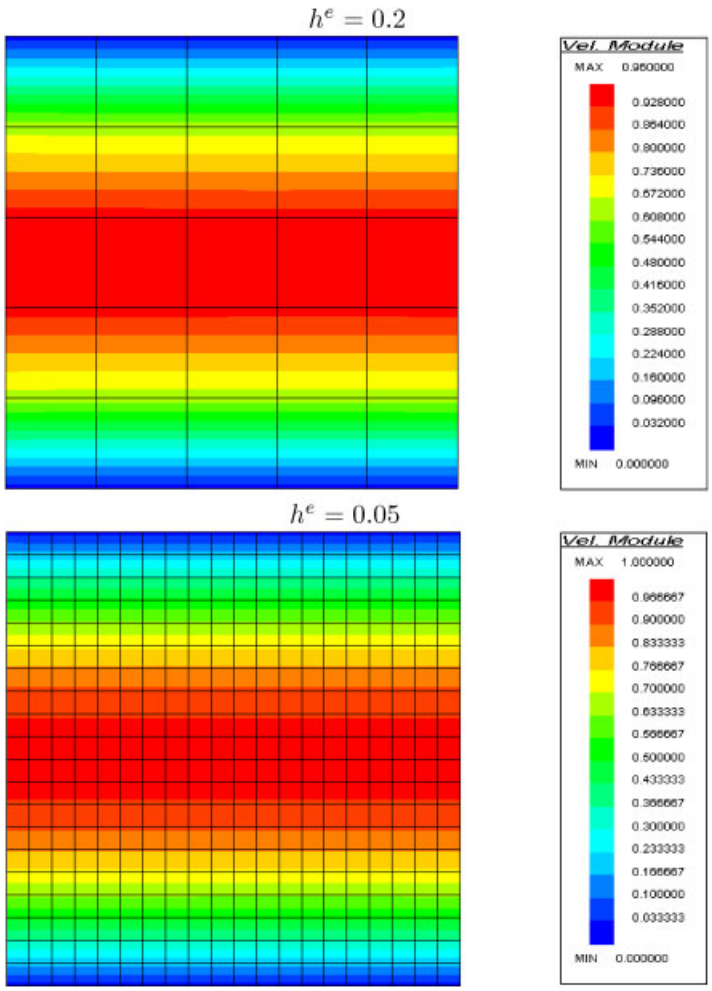

Figure 2. Poiseuille flow. Velocity distribution.

well with the expected results. The velocity field for the 10×10 mesh is practically the same and there are not significant differences between the formulations based on the diagonal and the non-diagonal stabilizing matrices.

The pressure distributions obtained with the diagonal and non-diagonal taus are presented in Figures 3 and 4, respectively. It can be seen that oscillations on the pressure field near the inlet boundary are generated which decrease as the mesh is refined. This fact is better observed in the pressure variation along the centreline plotted in Figures 5 and 6. It is also shown in these figures that the oscillations reduce when the non-diagonal tau is used in the formulation.

The origin of these oscillations lies on the utilization of low order elements in the discretization. The GLS method is a weighted residual formulation and the stabilization term depends

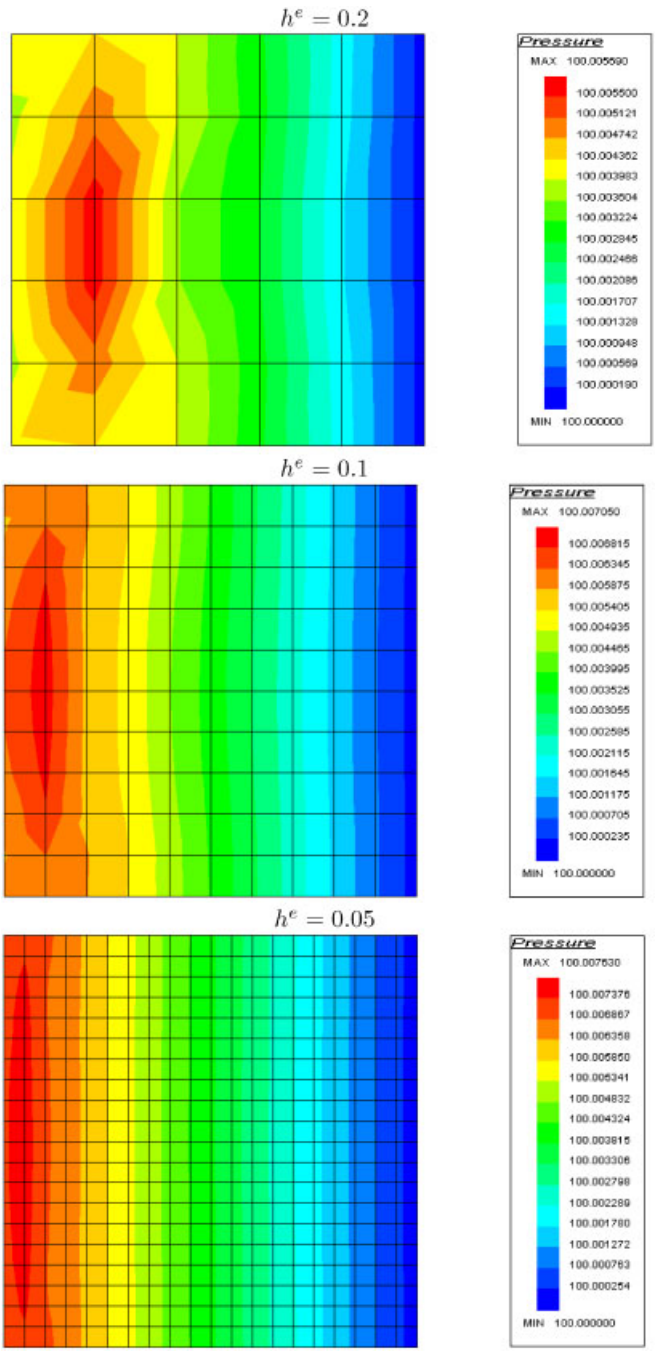

Figure 3. Poiseuille flow. Pressure distribution for τ_d.

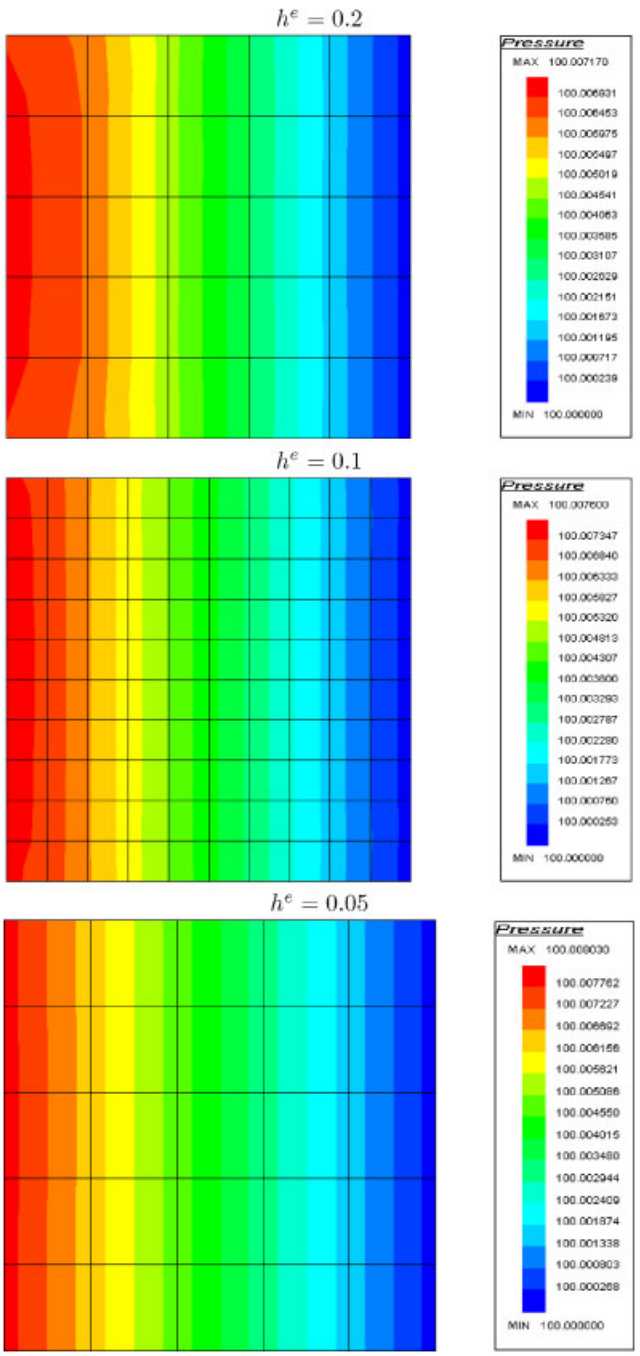

Figure 4. Poiseuille flow. Pressure distribution for τ_{nd}.

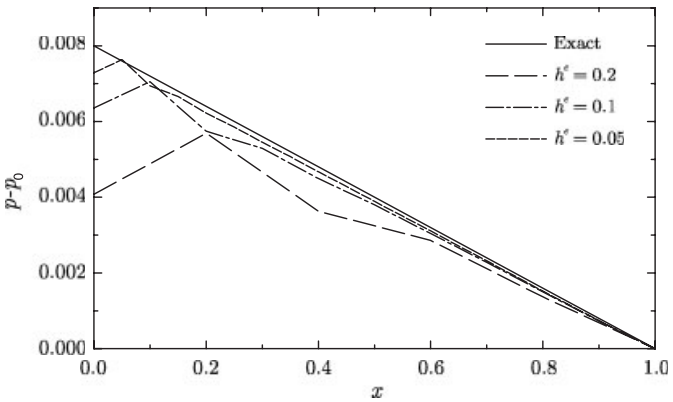

Figure 5. Poiseuille flow. Pressure distribution along $y = 0$ for τ_d.

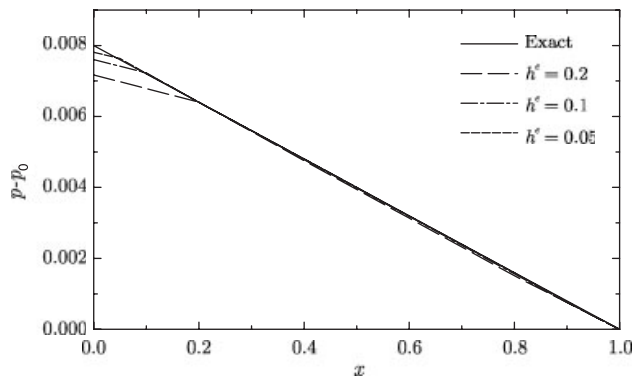

Figure 6. Poiseuille flow. Pressure distribution along $y = 0$ for τ_{nd}.

on the complete residual of the differential equations. However, in the case of the Navier–Stokes equations, piecewise linear elements cannot capture the second derivatives related to the viscous stresses leading to what Jansen *et al.* [52] have termed as poor consistency expressed by an incomplete approximation to the residual affecting the accuracy of the numerical solution. The effect of this poor consistency can be reduced by the use of fine meshes since the stabilization term includes the matrix tau which vanishes as the mesh is refined, as already highlighted earlier when the pressure distributions have been introduced. High order elements do not exhibit this problem because they are capable of effectively drive the residual to zero and hence the consistency is maintained. From the practical point of view, the utilization of low order elements presents clear advantages and alternatives to high order elements have been proposed by Hauke [53] and Jansen *et al.* [52] who keep the discretization with low order elements and introduce an estimation of the second order derivatives of the residual in the stabilization term by a least square projection.

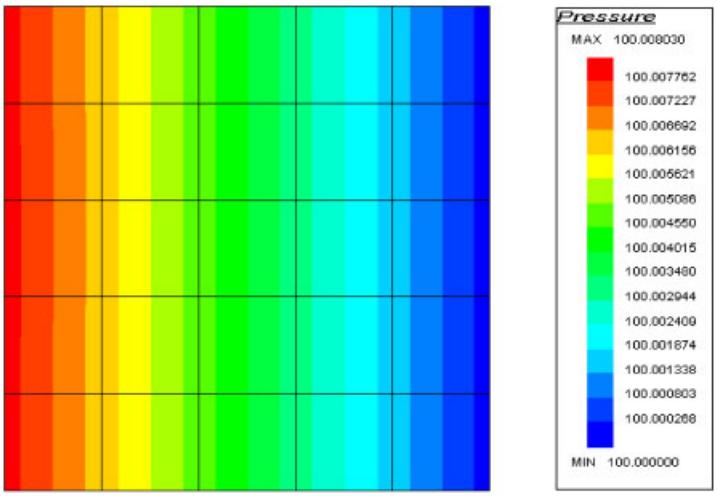

Figure 7. Poiseuille flow. Pressure distribution for the enhanced GLS residual with $h^e = 0.2$ and τ_d.

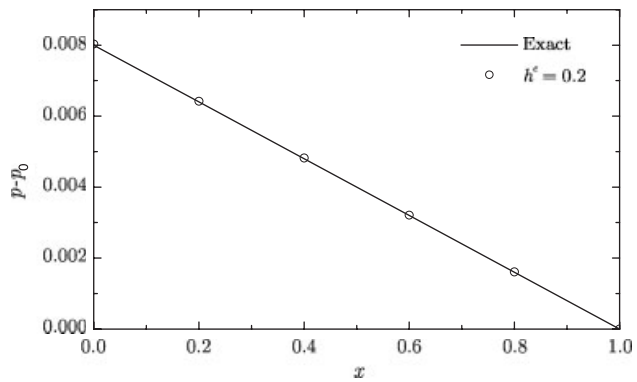

Figure 8. Poiseuille flow. Pressure distribution along $y = 0$ for the enhanced residual with τ_d.

Other solutions involve the modification of the variational formulation [19] or the addition of boundary integrals [54, 55].

To illustrate the effect of the poor consistency on the GLS/SUPG method with low order elements, the residual included in the stabilization term has been directly modified in the code since in Poiseuille flow the momentum equation reduces to a balance between the pressure gradient and the viscous stress tensor. The resulting pressure field obtained with the mesh of 5×5 elements and the pressure distribution along the centreline (see Figures 7 and 8, respectively) show a drastic elimination of the oscillations. These results have been derived with the diagonal tau matrix though the non-diagonal tau matrix and finer meshes yield similar distributions.

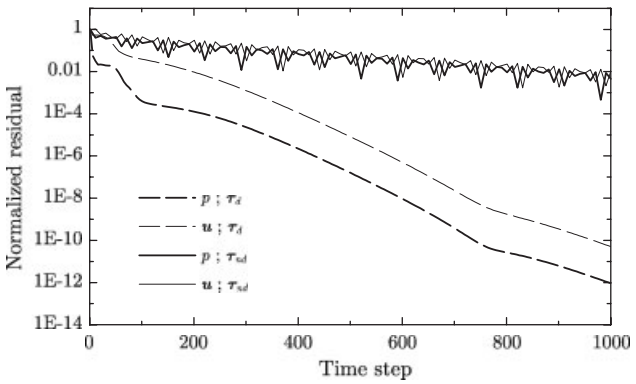

Figure 9. Poiseuille flow. Residual convergence for CFL = 1, $h^e = 0.2$.

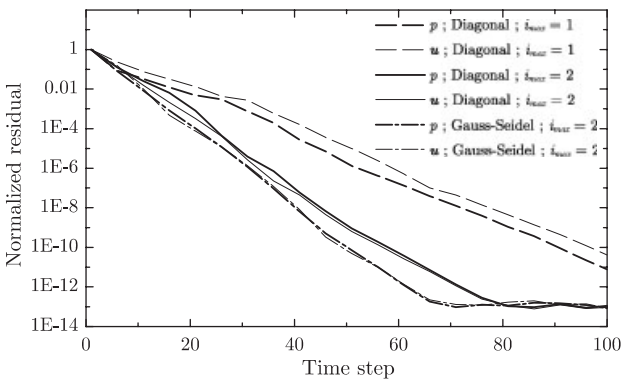

Figure 10. Poiseuille flow. Residual convergence for CFL = 10, $h^e = 0.2$ and τ_d.

Figure 9 shows the convergence of the normalized residual for the diagonal algorithm with one corrector pass and CFL = 1. The formulation with the diagonal stabilization matrix presents a better convergence rate and, as will be seen later, allows higher CFL numbers so the evolution to the steady state solution can be accelerated. This can be attributed to the fact that the diagonal matrix is an extension to compressible flows of the intrinsic time matrix derived for incompressible flows (see for example Reference [18]) and, hence, is better suited for nearly incompressible problems.

Using the formulation with the diagonal tau, the normalized residuals for different versions of the segregated algorithm and CFL = 10 are presented in Figure 10. There, it can be observed that two passes through the corrector loop increases the rate of convergence with respect to one pass but requiring a considerably higher computational effort. Analogously, the Gauss–Seidel version attains better convergence than the diagonal one but at an extra cost in computational resources. On the other hand, the Jacobi algorithm needs low CFL numbers to achieve convergence.

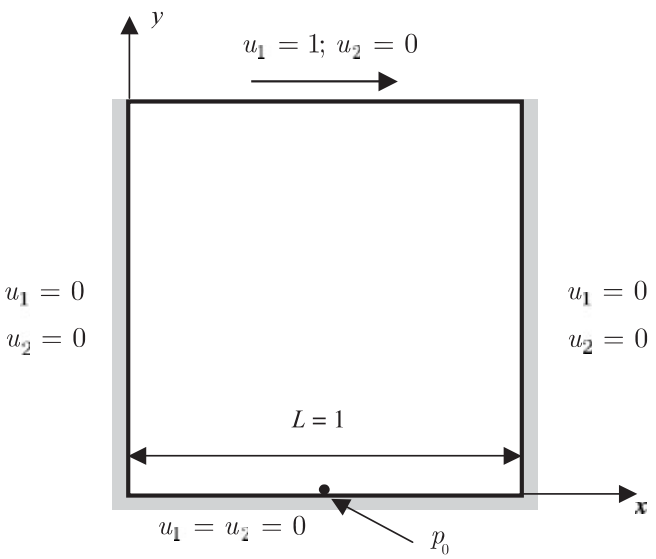

Figure 11. Non-leaking cavity flow. Problem setup.

9.1.2. Non-leaking cavity flow. This problem is a classical test to evaluate the behaviour of numerical algorithms for incompressible flows. The example setup is stated in Figure 11 and consists of a unit square domain containing a viscous fluid which is sheared at the top edge via a horizontal velocity causing recirculation to develop inside the cavity. Figure 11 also sketches the applied boundary conditions, that is, no-slip condition on all edges, including the top horizontal edge, where the velocity components $u_1 = 1$ and $u_2 = 0$ are specified. The pressure field is constrained by fixing the value of the central node located at the bottom edge to a value of $p_0 = 100$ which corresponds to a Mach number $M = 0.1$.

The domain is discretized by a mesh of 40×40 square elements with refinements at both top corners to control the singularities of the numerical solution, as shown in Figure 12. The results of Reference [22] confirmed that for the coupled solver this mesh was sufficient.

The streamlines and pressure distributions for a Reynolds number of 400 and 1000 are presented in Figures 13 and 14, respectively. In Figure 15, the velocity components along the vertical and horizontal central lines are compared with the results published by Ghia *et al.* [56] giving an excellent agreement.

The residual convergence of the diagonal version of the segregated algorithm with the diagonal tau matrix, a condition CFL = 10 and global time step strategy is presented in Figure 16. It can be seen that as the number of corrector passes increases, the better is the convergence rate. However, though the improvement achieved from one to two corrector passes can be regarded as significant, this does not hold from two to four corrector passes. In this case, the slight enhancement of the convergence rate obtained with four corrector passes is at the expense of a much greater computational effort.

Figure 17 compares the residual convergence with local and global time step strategies for CFL = 10 and two corrector passes, showing that the local time strategy improves the convergence rate.

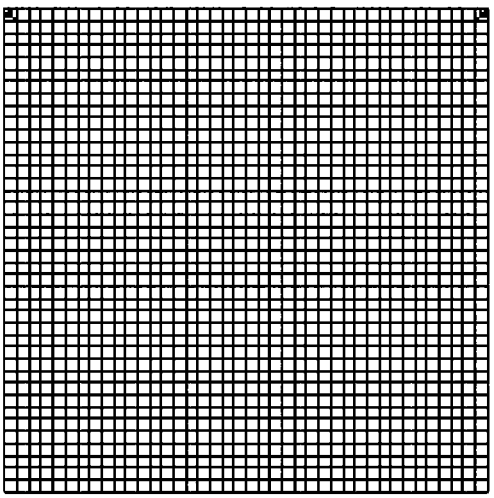

Figure 12. Non-leaking cavity flow. Mesh.

Several runs have been performed with the Jacobi and the Gauss–Seidel versions but they require lower CFL numbers to attain convergence. Thus, although the Jacobi and specially the Gauss–Seidel methods give better convergence rates than the diagonal algorithm, the latter has been proved to be the most robust, allowing speed up of the evolution of the residuals through the use of higher CFL numbers.

Finally, the segregated algorithm shows lower convergence rates than the analogous coupled method (see Reference [53]).

9.1.3. Backward facing step. This example studies the behaviour of a viscous flow which experiments a sudden expansion caused by the presence of a circulation generating step. The problem setup is given in Figure 18. Taking the height of the step h as a longitudinal reference, the inlet is located four step heights upstream of the step while the outlet is 36 step heights downstream, sufficiently far from the step to ensure fully developed flow conditions. The expansion ratio at the step is 2:3. The applied boundary conditions are also sketched in Figure 18. A parabolic velocity profile, corresponding to a fully developed flow between two parallel plates, is specified at the inlet section. At the outlet, a reference pressure $p_0 = 100$ has been stated leading to a Mach number $M = 0.1$. Non-slip conditions are imposed on the rest of the boundaries.

Several cases have been analysed for Reynolds numbers of 73, 191 and 229 based on the average velocity at the inlet and the step height. The results are afterwards compared with the experimental data of Denham and Patrick [58] and the numerical data published in Zienkiewicz *et al.* [57].

As shown in Figure 19 and following the work of Reference [57], in the discretization a non-uniform mesh of 1660 quadrilateral elements and 1751 nodes has been employed. The smallest element, with dimensions $h_x^e = 0.178h$ and $h_y^e = 0.048h$, is located at the upper corner of the step.

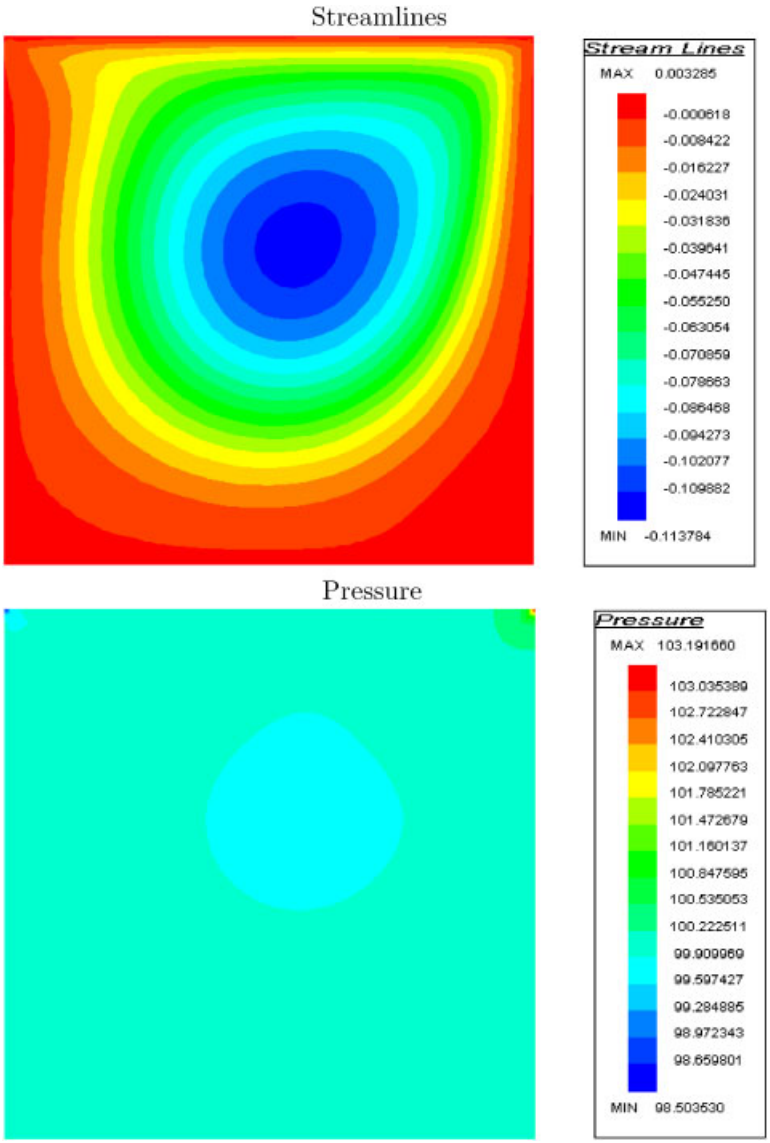

Figure 13. Non-leaking cavity flow $Re = 400$. Streamlines and pressure.

Following the conclusions derived from the previous example, the calculations have been carried out with the diagonal algorithm, the diagonal stabilization matrix, two corrector passes and the local time step strategy. CFL = 1 has been used due to convergence limitations. For each simulation, the initial value has been taken from the solution of the immediately lower Reynolds number case.

The obtained pressure and streamline contours are shown in Figures 20 and 21, respectively showing a good correlation with the results presented by the above mentioned references.

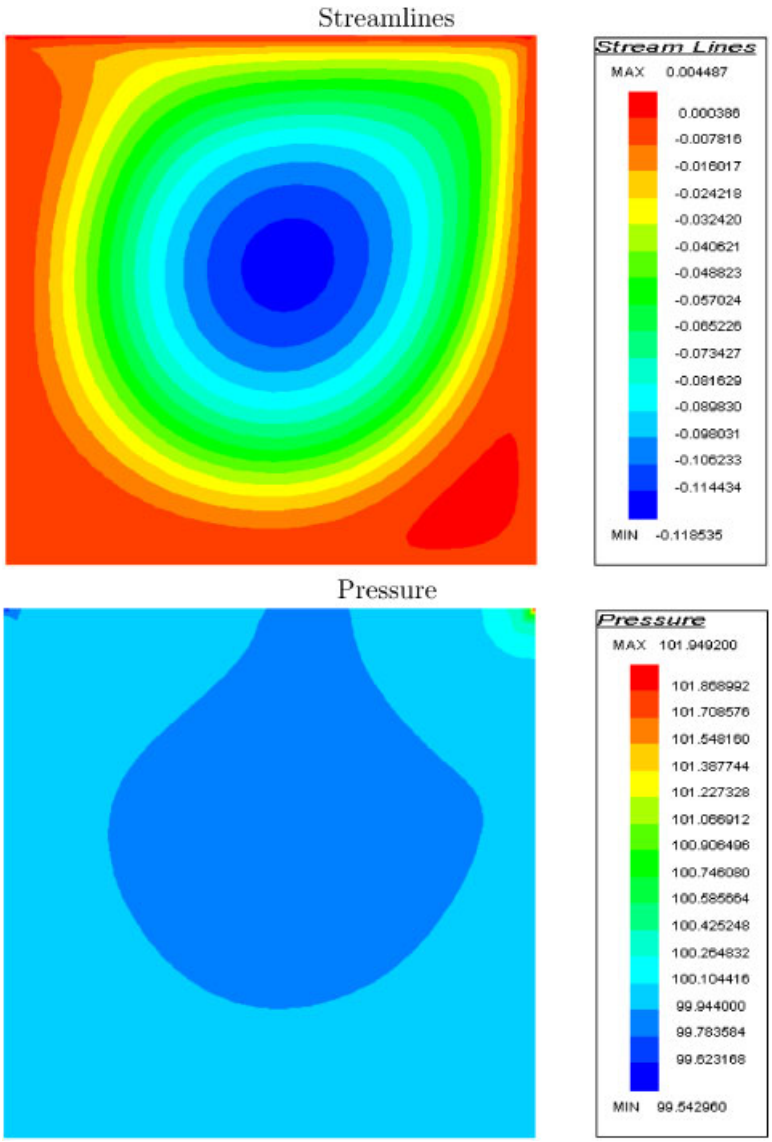

Figure 14. Non-leaking cavity flow $Re = 1000$. Streamlines and pressure.

Figure 22 compares the numerical values yielded by the present work with the measurements of Denham and Patrick [58] of the horizontal velocity u_1 in several sections downstream the step. The agreement attained is fully satisfactory.

One parameter which serves to assess the performance of the numerical algorithm in this problem is the length of the recirculation zone given by the location of the reattachment point after the step. Table I compares the results reported by the different authors. For low

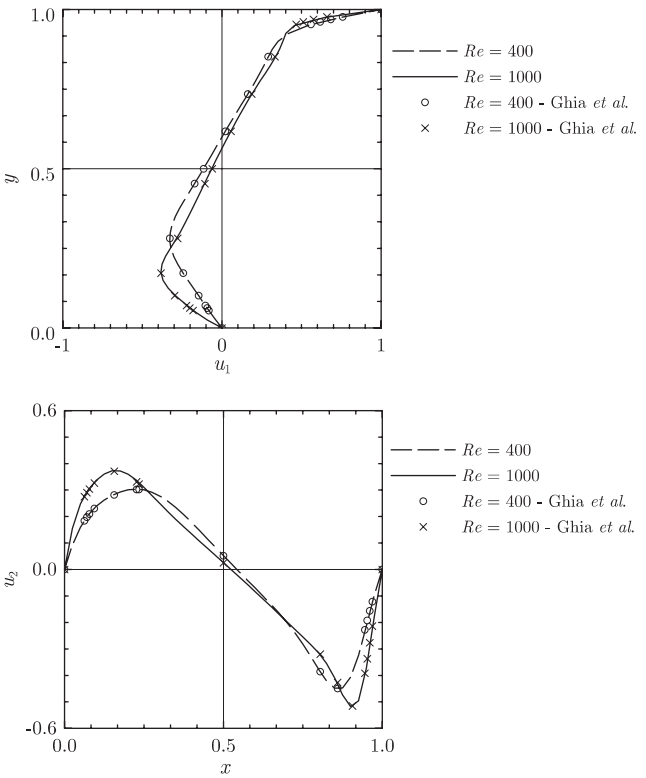

Figure 15. Non-leaking cavity flow. Horizontal velocity component along the vertical centreline and vertical velocity component along the horizontal centreline.

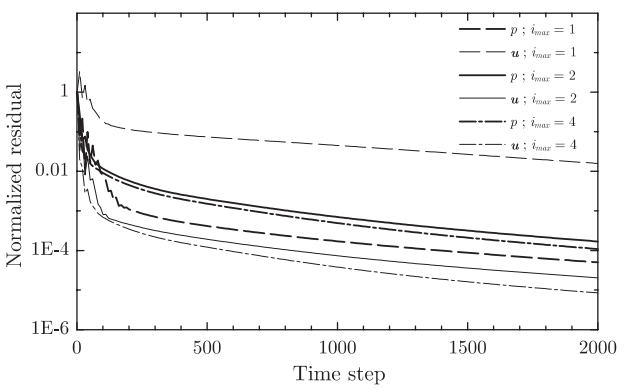

Figure 16. Non-leaking cavity flow $Re = 400$. Residual evolution for the diagonal algorithm with CFL = 10, global time stepping.

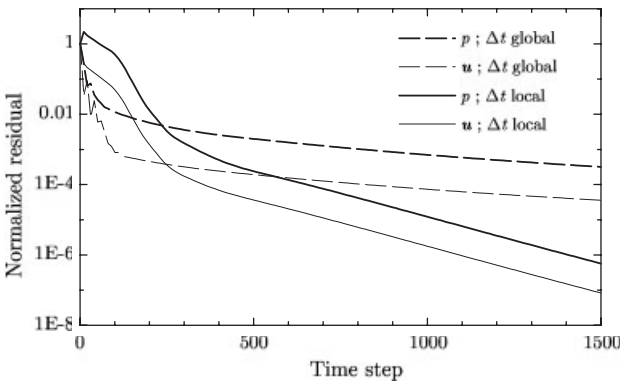

Figure 17. Non-leaking cavity flow $Re = 400$. Residual evolution for the diagonal algorithm with CFL = 10, $i_{\max} = 2$.

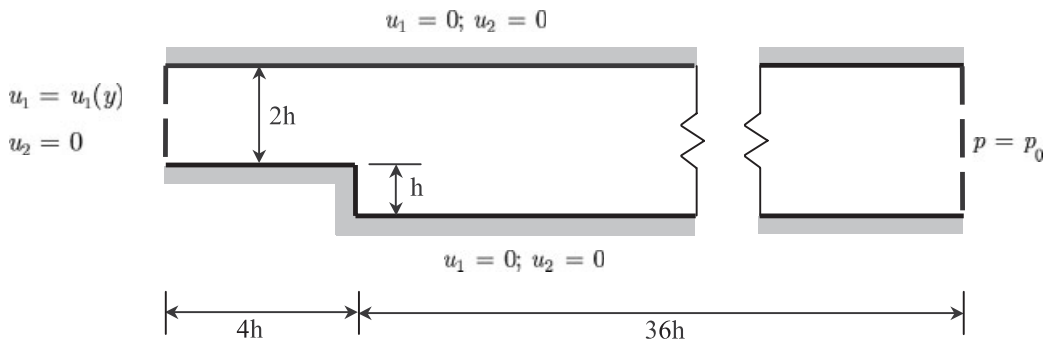

Figure 18. Backward facing step. Problem setup.

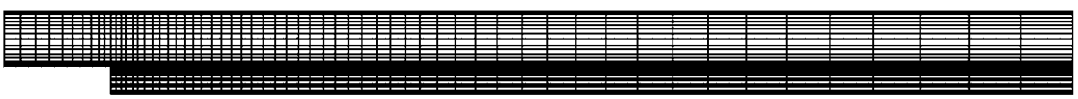

Figure 19. Backward facing step. Mesh.

Reynolds numbers, the algorithm developed in the present work predicts slightly longer lengths than those reported by Zienkiewicz *et al.* but at the higher Reynolds number this tendency reverses and the reattachment length given by our algorithm is closer to the experimental value of Denham and Patrick.

9.2. Compressible flows

9.2.1. Normal shock. The present example consists of a one-dimensional inviscid flow which crosses a Mach 2 shock. A mesh of 21×1 square elements covering the domain $-2.1 \leqslant x \leqslant 2.1$

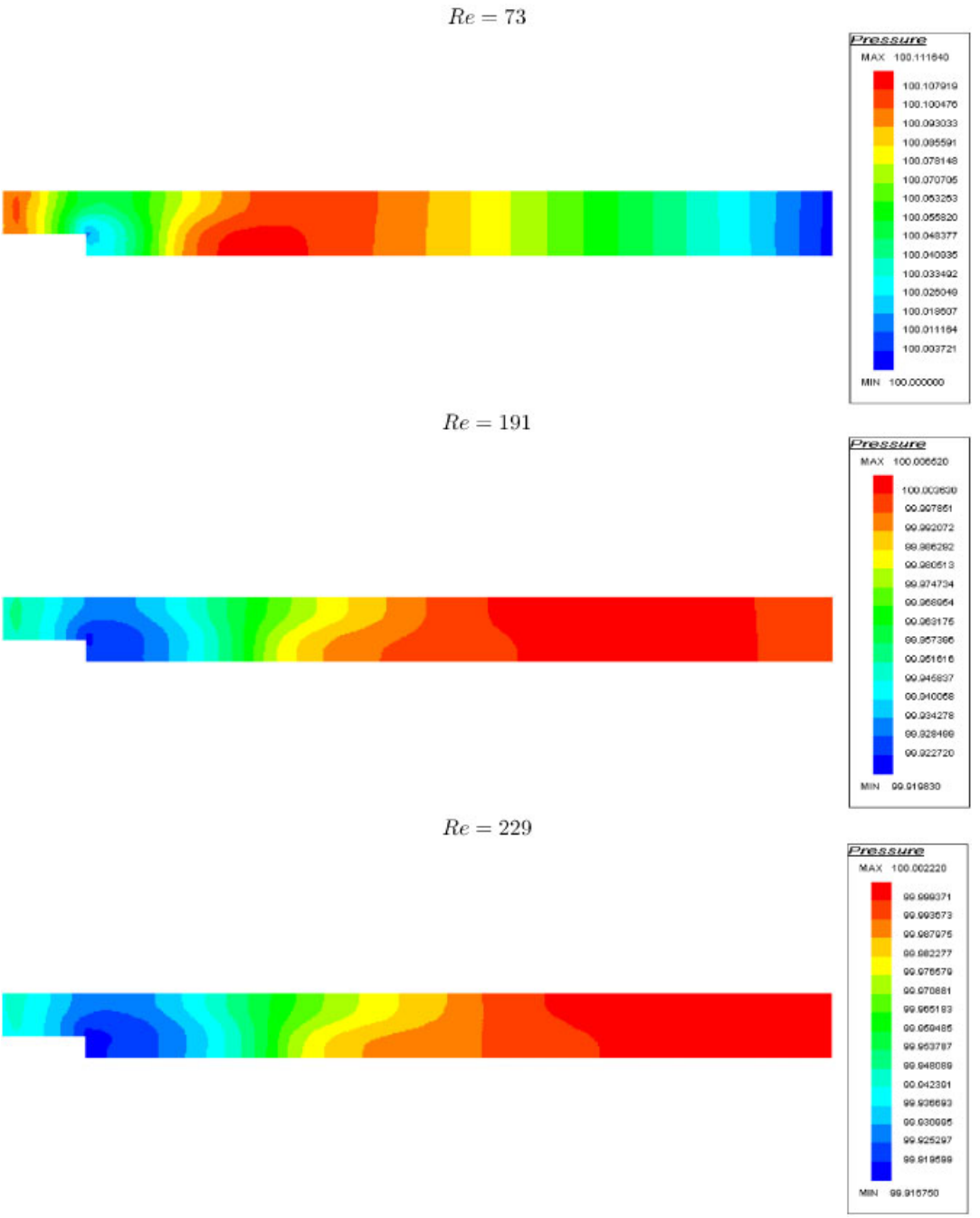

Figure 20. Backward facing step. Pressure contours.

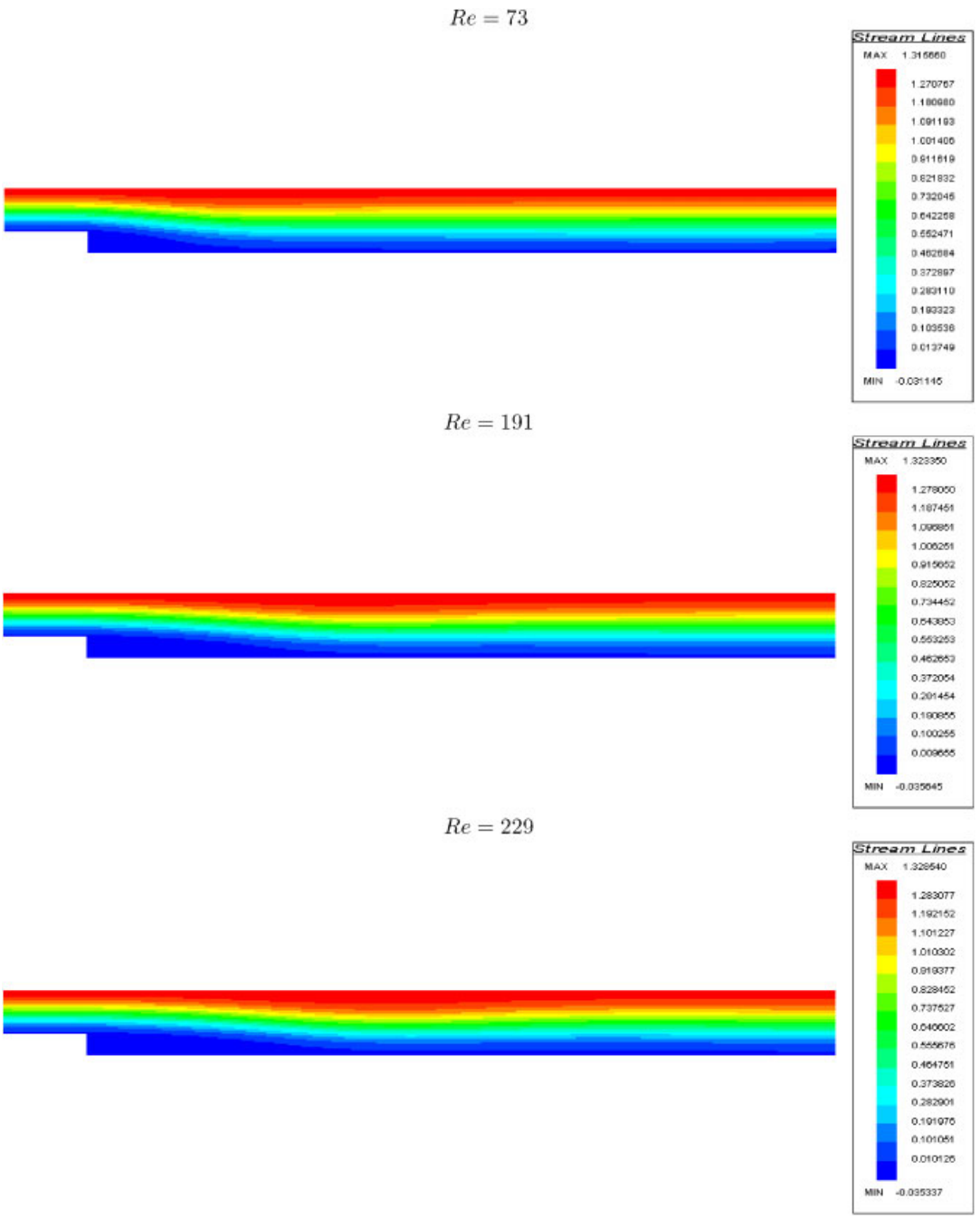

Figure 21. Backward facing step. Streamfunction.

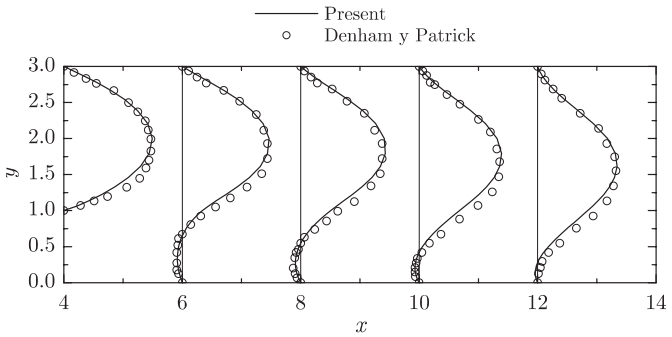

Figure 22. Backward facing step. Velocity profiles after the step.

Table I. Backward facing step. Recirculation length for various methods as a function of Reynolds number. Values between parenthesis denote percentage error with respect to experimental values [58].

Re	Denham and Patrick [58]	Zienkiewicz *et al.* [57]	Present
73	3.9	4.8 (23.0)	4.9 (25.6)
191	8.6	9.2 (7.0)	9.5 (10.5)
229	10.0	10.9 (9.0)	10.4 (4.0)

has been employed. The shock is placed at $x=0$ and the following initial conditions have been applied:

$$x<0\begin{cases} M=2 \\ u_1=1 \\ \rho=1 \\ p=0.25 \end{cases}, \qquad x>0\begin{cases} M=0.5 \\ u_1=0.25 \\ \rho=4 \\ p=1 \end{cases} \tag{99}$$

As boundary conditions, the velocity components and the density have been specified at the inlet. The vertical velocity component has been set to zero in the entire domain.

Figures 23 and 24 give the density and horizontal velocity variations across the shock, respectively. The figures show that the numerical method is capable of capturing the shock within three elements. The solution has been obtained with CFL = 1, the non-diagonal stabilization matrix and the H–M shock capturing operator in order to achieve convergence.

The behaviour of the different versions of the segregated algorithm is evaluated through the evolution of the normalized residuals. Figure 25 shows the residual convergence corresponding to the diagonal version with global time step strategy and different corrector passes. Initially, the algorithm presents better convergence rate when two corrector passes are used but at the end the one corrector pass attains the same levels of residual reduction. The global and local time step strategies are analysed in Figures 26 and 27. Both strategies present similar performance although the global time step strategy drives the residuals to lower values. In

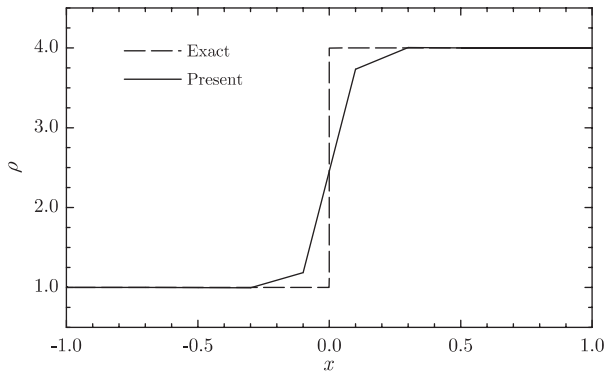

Figure 23. Normal shock wave $M = 2$. Density.

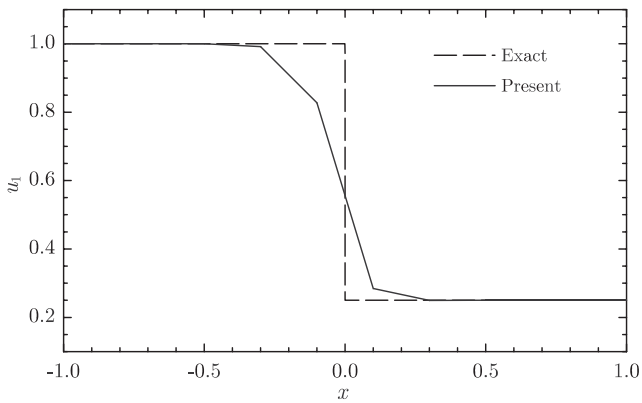

Figure 24. Normal shock wave $M = 2$. x velocity component.

Figure 28 the diagonal and Gauss–Seidel algorithms are compared demonstrating that the latter does not represent a practical alternative to the diagonal version.

9.2.2. Oblique shock wave. This problem corresponds to an inviscid uniform flow with $M = 2$ over a wedge at an angle of 10° generating an oblique shock with an angle of 27.4° emanating from the leading edge of the wedge. See Figure 29 for a layout of the problem.

The computational domain is a unit square discretized by a mesh of 20×20 square elements. As boundary conditions, the velocity vector and the density have been prescribed at the inflow and top boundaries; none have been specified at the exit; and zero normal velocity component has been set at the wall, i.e. $u_2 = 0$.

Figure 30 shows the Mach number distribution in the domain where the presence of the shock can be clearly identified. The variations of the density and the horizontal velocity component along the vertical line $x = 0.9$ are represented in Figures 31 and 32, respectively. In this example, since the discontinuity across the shock is lower than the case of the normal

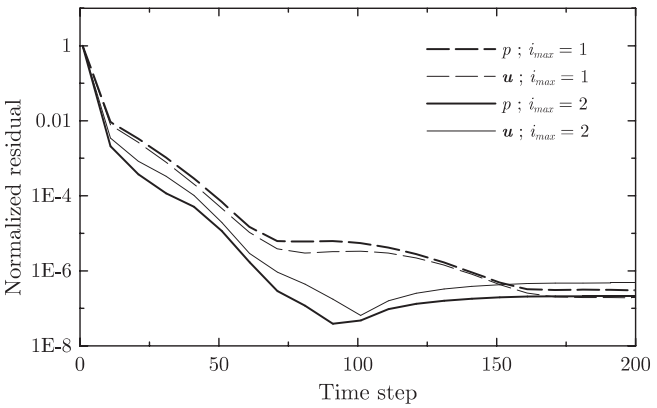

Figure 25. Normal shock wave $M = 2$. Residual convergence for diagonal algorithm with global Δt.

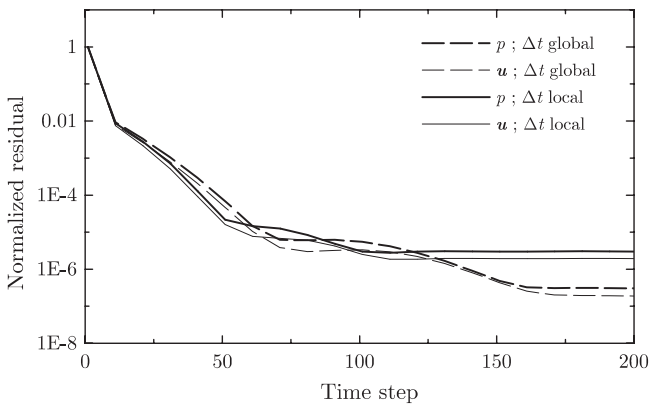

Figure 26. Normal shock wave $M = 2$. Residual convergence for diagonal algorithm with $i_{\max} = 1$.

shock, the algorithm has converged with both the diagonal and non-diagonal stabilizing matrices and even without the activation of the discontinuity capturing (DC) operator for low CFL numbers. In the figures, the effect of the DC operator can be clearly observed since the oscillations near the shock reduce significantly when it is included in the formulation. Observe also that the version with the non-diagonal matrix tau and the DC operator gives the most accurate results.

Regarding the residual evolution, Figure 33 shows the normalized residuals for the diagonal algorithm with global time step strategy and CFL = 10. In addition to more accurate results, the use of the non-diagonal tau matrix yields better convergence rates than the diagonal tau. Also, increasing the number of corrector passes to two initially improves that rate. Finally, the non-diagonal tau with one corrector pass attains comparable rates to two passes with the diagonal tau.

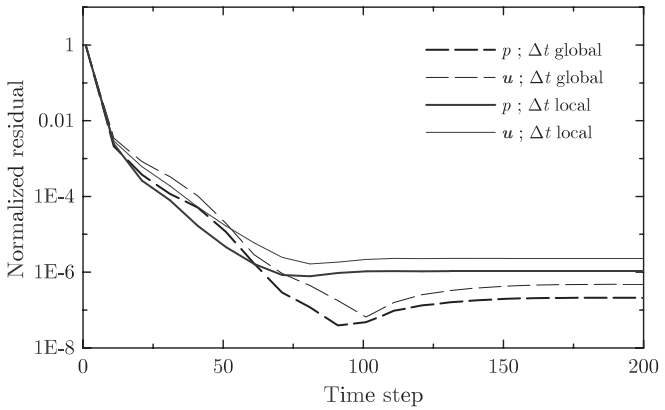

Figure 27. Normal shock wave $M = 2$. Residual convergence for diagonal algorithm with $i_{\max} = 2$.

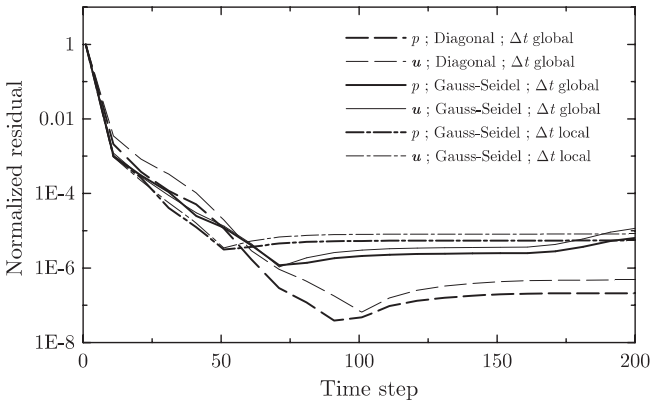

Figure 28. Normal shock wave $M = 2$. Residual convergence for $i_{\max} = 2$.

In Figure 34 the residuals for local time stepping are presented. In general, local time stepping gives slightly higher rates than the global strategy (with the exception of the formulation with the non-diagonal tau matrix and two corrector passes).

Figure 35 compares the residual evolution obtained with the diagonal and the Gauss–Seidel iterative methods with the non-diagonal tau matrix. The diagonal tau matrix Gauss–Seidel method required diminish the CFL to unity. Although the Gauss–Seidel method has a faster convergence rate per time step, the extra cost makes the diagonal method more efficient. At the beginning both methods converge at the same rate, but then, the Gauss–Seidel algorithm is able to drive the residual faster to zero.

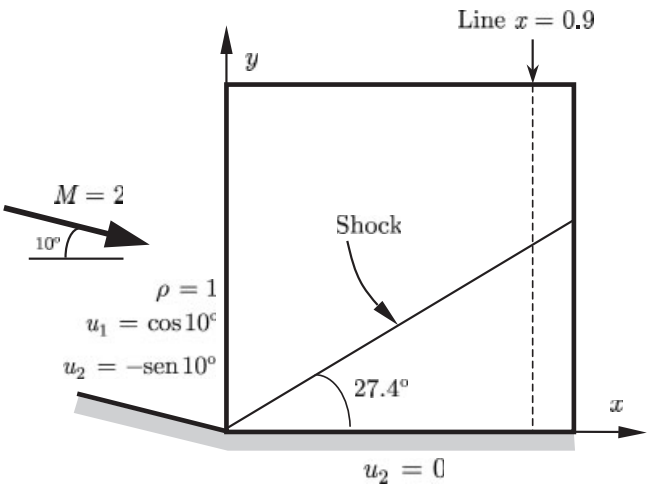

Figure 29. Oblique shock wave $M = 2$. Problem setup.

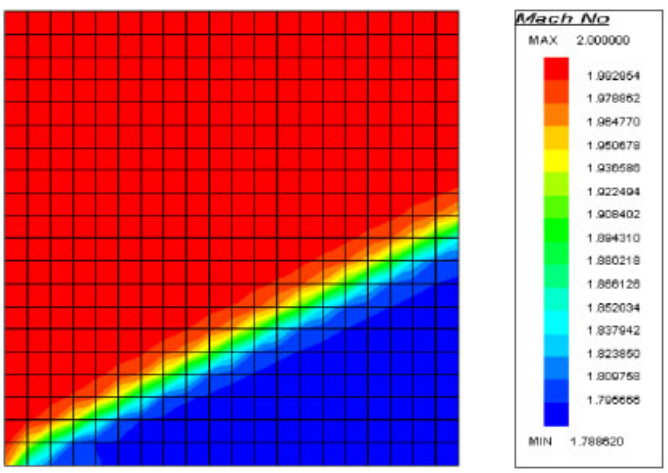

Figure 30. Oblique shock wave $M = 2$. Mach number.

10. CONCLUSIONS

The thermodynamics of thermally perfect isothermal flows has been clarified and set up in the framework of symmetric systems. The generalized entropy function has been identified as the kinetic energy minus the entropy, leading to the corresponding set of entropy variables. For this class of problems, the non-linear stability principle is a combination of the second law of thermodynamics and the mechanical energy equation.

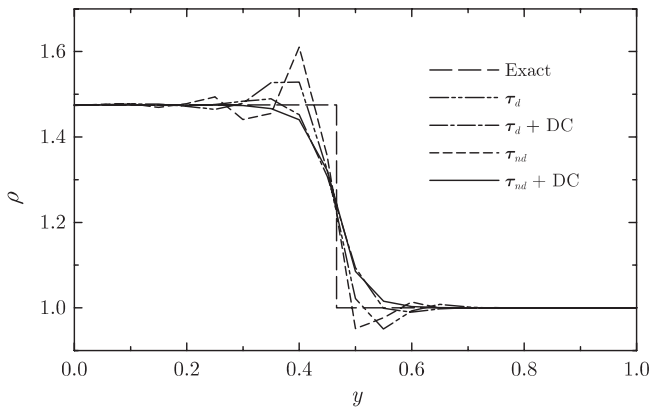

Figure 31. Oblique shock wave $M = 2$. Density along $x = 0.9$.

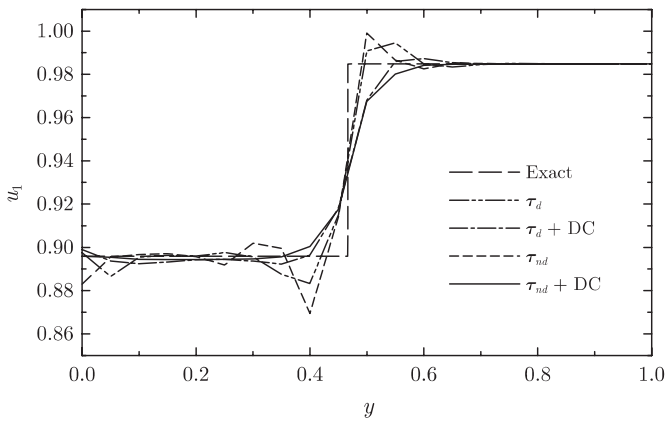

Figure 32. Oblique shock wave $M = 2$. x-velocity component along $x = 0.9$.

The examples of perfect gas, slightly incompressible liquid and incompressible fluid have been addressed. In particular, the incompressible limit recovers the classical mechanical energy principle as the stability principle.

Then, the flow equations have been solved with a stabilized method. Various segregated strategies, namely the diagonal, Jacobi and Gauss–Seidel strategies, have been proposed that work well for incompressible and compressible flows, even in the presence of strong discontinuities. The incompressible limit of the formulation is well behaved for the set of both, entropy and pressure primitive variables.

Due to its simplicity and robustness, from the practical point of view the diagonal algorithm is the most advantageous for both compressible and incompressible flows because in general it allows higher CFL numbers. The Gauss–Seidel version usually gives good convergence rates but it requires a bigger computational cost than the diagonal method. Finally, the Jacobi strategy shows poor performance.

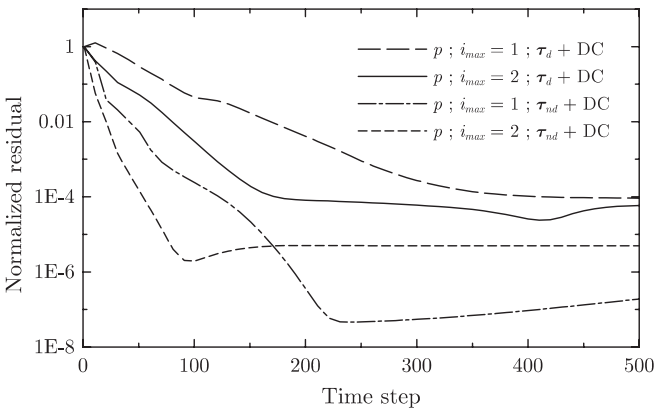

Figure 33. Oblique shock wave $M = 2$. Residual convergence for diagonal algorithm. Global time stepping. CFL = 10.

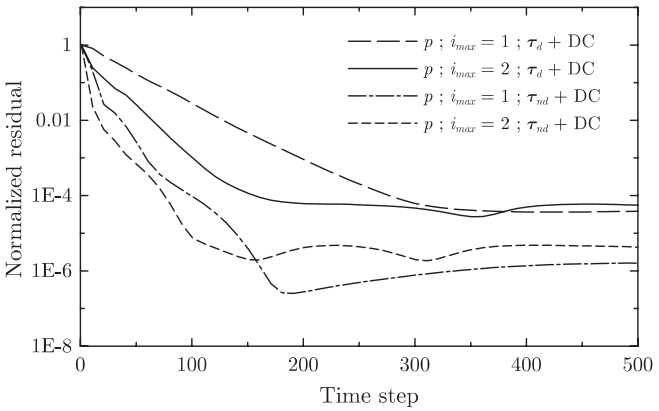

Figure 34. Oblique shock wave $M = 2$. Residual convergence for diagonal algorithm. Local time stepping. CFL = 10.

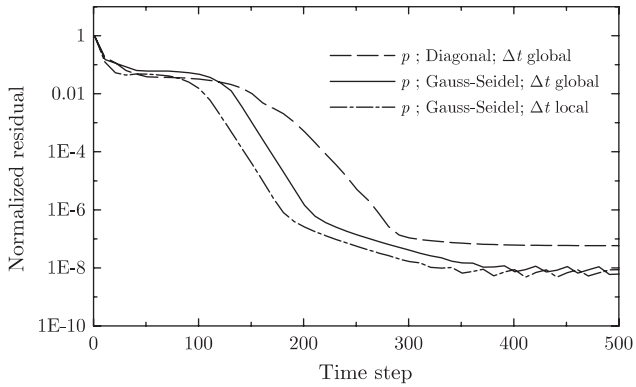

Figure 35. Oblique shock wave $M = 2$. Residual convergence for diagonal algorithm and Gauss–Seidel. CFL = 1, $i_{\max} = 1$.

For nearly incompressible flows, the best choice of algorithm parameters is to use the formulation with the diagonal tau matrix, two corrector passes and local time stepping. For supersonic compressible flows, the most suitable option both, in terms of numerical accuracy and residual convergence, corresponds to the formulation with the non-diagonal tau matrix and one corrector pass. The influence of the time step strategy in this case is less clear and either the global or local time stepping could be used.

APPENDIX A

A.1. Coefficient matrices for a thermally perfect, isothermal fluid

Let be a thermally perfect ($e = e(T)$) fluid with equation of state

$$p = p(\rho) \tag{A1}$$

As a function of v the specific volume, the isothermal expansion coefficient can be expressed as

$$\beta_T = -\frac{1}{v}\left(\frac{\partial v}{\partial p}\right)_T = \frac{1}{\rho}\left(\frac{\partial \rho}{\partial p}\right)_T \tag{A2}$$

So

$$\left(\frac{\partial p}{\partial \rho}\right)_T = \frac{1}{\rho\beta_T} \tag{A3}$$

Also, the viscous coefficients are μ^{visc}, λ^{visc} and

$$\chi^{\text{visc}} = \lambda^{\text{visc}} + 2\mu^{\text{visc}} \tag{A4}$$

A.1.1. Coefficient matrices for primitive variables. For the set of primitive variables the coefficient matrices can be written as follows:

$$\mathbf{Y} = \begin{Bmatrix} p \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{A5}$$

The metric tensors can be expressed as

$$\mathbf{A}_0^{-1} = \frac{1}{\rho}\begin{bmatrix} \dfrac{1}{\beta_T} & 0 & 0 & 0 \\ -u_1 & 1 & 0 & 0 \\ -u_2 & 0 & 1 & 0 \\ -u_3 & 0 & 0 & 1 \end{bmatrix} \tag{A6}$$

$$\mathbf{A}_0 = \rho \begin{bmatrix} \beta_T & 0 & 0 & 0 \\ \beta_T u_1 & 1 & 0 & 0 \\ \beta_T u_2 & 0 & 1 & 0 \\ \beta_T u_3 & 0 & 0 & 1 \end{bmatrix} \tag{A7}$$

The Euler Jacobians with respect to $\mathbf{Y}$, $\mathbf{A}_i = \mathbf{F}^{\text{adv}}_{i,\mathbf{Y}}$, are given by

$$\mathbf{A}_1 = \begin{bmatrix} \rho\beta_T u_1 & \rho & 0 & 0 \\ \rho\beta_T u_1^2 + 1 & 2\rho u_1 & 0 & 0 \\ \rho\beta_T u_1 u_2 & \rho u_2 & \rho u_1 & 0 \\ \rho\beta_T u_1 u_3 & \rho u_3 & 0 & \rho u_1 \end{bmatrix} \tag{A8}$$

$$\mathbf{A}_2 = \begin{bmatrix} \rho\beta_T u_2 & 0 & \rho & 0 \\ \rho\beta_T u_1 u_2 & \rho u_2 & \rho u_1 & 0 \\ \rho\beta_T u_2^2 + 1 & 0 & 2\rho u_2 & 0 \\ \rho\beta_T u_2 u_3 & 0 & \rho u_3 & \rho u_2 \end{bmatrix} \tag{A9}$$

$$\mathbf{A}_3 = \begin{bmatrix} \rho\beta_T u_3 & 0 & 0 & \rho \\ \rho\beta_T u_1 u_3 & \rho u_3 & 0 & \rho u_1 \\ \rho\beta_T u_2 u_3 & 0 & \rho u_3 & \rho u_2 \\ \rho\beta_T u_3^2 + 1 & 0 & 0 & 2\rho u_3 \end{bmatrix} \tag{A10}$$

The diffusivity coefficient matrices $\mathbf{K}_{ij}$, where $\mathbf{K}_{ij}\mathbf{Y}_{,j} = \mathbf{F}^{\text{diff}}_i$, are

$$\mathbf{K}_{11} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \chi^{\text{visc}} & 0 & 0 \\ 0 & 0 & \mu^{\text{visc}} & 0 \\ 0 & 0 & 0 & \mu^{\text{visc}} \end{bmatrix} \tag{A11}$$

$$\mathbf{K}_{22} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & \chi^{\text{visc}} & 0 \\ 0 & 0 & 0 & \mu^{\text{visc}} \end{bmatrix} \tag{A12}$$

$$\mathbf{K}_{33} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & \mu^{\text{visc}} & 0 \\ 0 & 0 & 0 & \chi^{\text{visc}} \end{bmatrix} \tag{A13}$$

$$\mathbf{K}_{12} = \mathbf{K}_{21}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda^{\text{visc}} & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{A14}$$

$$\mathbf{K}_{13} = \mathbf{K}_{31}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^{\text{visc}} \\ 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \end{bmatrix} \tag{A15}$$

$$\mathbf{K}_{23} = \mathbf{K}_{32}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^{\text{visc}} \\ 0 & 0 & \mu^{\text{visc}} & 0 \end{bmatrix} \tag{A16}$$

Note that the viscous submatrices $\mathbf{K}_{ij}$ are the same as those for entropy variables $\tilde{\mathbf{K}}_{ij}$. Also,

$$\mathbf{V}_{,\mathbf{Y}} = \begin{bmatrix} 1/\rho & -u_1 & -u_2 & -u_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A17}$$

and

$$\mathbf{A}_0^{\mathrm{DC}} = \mathbf{V}_{,\mathbf{Y}}^{\mathrm{T}} \mathbf{A}_0 = \begin{bmatrix} \beta_T & 0 & 0 & 0 \\ 0 & \rho & 0 & 0 \\ 0 & 0 & \rho & 0 \\ 0 & 0 & 0 & \rho \end{bmatrix} \tag{A18}$$

A.1.2. Coefficient matrices for entropy variables. For the set of entropy variables the coefficient matrices can be written as follows:

$$\mathbf{V} = \begin{Bmatrix} \dfrac{p}{\rho} - T(s - s_0) - \dfrac{1}{2}|\mathbf{u}|^2 \\ u_1 \\ u_2 \\ u_3 \end{Bmatrix} \tag{A19}$$

The metric tensors can be expressed as

$$\tilde{\mathbf{A}}_0^{-1} = \frac{1}{\rho} \begin{bmatrix} \dfrac{1}{\rho\beta_T} + |\mathbf{u}|^2 & -u_1 & -u_2 & -u_3 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & \text{symm} & & 1 \end{bmatrix} \tag{A20}$$

$$\tilde{\mathbf{A}}_0 = \rho^2 \beta_T \begin{bmatrix} 1 & u_1 & u_2 & u_3 \\ & \dfrac{1}{\rho\beta_T} + u_1^2 & u_1 u_2 & u_1 u_3 \\ & & \dfrac{1}{\rho\beta_T} + u_2^2 & u_2 u_3 \\ & \text{symm} & & \dfrac{1}{\rho\beta_T} + u_3^2 \end{bmatrix} \tag{A21}$$

The Euler Jacobians with respect to $\mathbf{V}$, $\tilde{\mathbf{A}}_i = \mathbf{F}^{\mathrm{adv}}_{i,\mathbf{V}}$, are given by

$$\tilde{\mathbf{A}}_1 = \rho^2\beta_T \begin{bmatrix} u_1 & \frac{1}{\rho\beta_T}+u_1^2 & u_1u_2 & u_1u_3 \\ & u_1\left(\frac{3}{\rho\beta_T}+u_1^2\right) & u_2\left(\frac{1}{\rho\beta_T}+u_1^2\right) & u_3\left(\frac{1}{\rho\beta_T}+u_1^2\right) \\ & & u_1\left(\frac{1}{\rho\beta_T}+u_2^2\right) & u_1u_2u_3 \\ & \text{symm} & & u_1\left(\frac{1}{\rho\beta_T}+u_3^2\right) \end{bmatrix} \tag{A22}$$

$$\tilde{\mathbf{A}}_2 = \rho^2\beta_T \begin{bmatrix} u_2 & u_1u_2 & \frac{1}{\rho\beta_T}+u_2^2 & u_2u_3 \\ & u_2\left(\frac{1}{\rho\beta_T}+u_1^2\right) & u_1\left(\frac{1}{\rho\beta_T}+u_2^2\right) & u_1u_2u_3 \\ & & u_2\left(\frac{3}{\rho\beta_T}+u_2^2\right) & u_3\left(\frac{1}{\rho\beta_T}+u_2^2\right) \\ & \text{symm} & & u_2\left(\frac{1}{\rho\beta_T}+u_3^2\right) \end{bmatrix} \tag{A23}$$

$$\tilde{\mathbf{A}}_3 = \rho^2\beta_T \begin{bmatrix} u_3 & u_1u_3 & u_2u_3 & \left(\frac{1}{\rho\beta_T}+u_3^2\right) \\ & u_3\left(\frac{1}{\rho\beta_T}+u_1^2\right) & u_1u_2u_3 & u_1\left(\frac{1}{\rho\beta_T}+u_3^2\right) \\ & & u_3\left(\frac{1}{\rho\beta_T}+u_2^2\right) & u_2\left(\frac{1}{\rho\beta_T}+u_3^2\right) \\ & \text{symm} & & u_3\left(\frac{3}{\rho\beta_T}+u_3^2\right) \end{bmatrix} \tag{A24}$$

The diffusivity coefficient matrices $\tilde{\mathbf{K}}_{ij}$, where $\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j} = \mathbf{F}^{\mathrm{diff}}_i$, are

$$\tilde{\mathbf{K}}_{11} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \chi^{\mathrm{visc}} & 0 & 0 \\ 0 & 0 & \mu^{\mathrm{visc}} & 0 \\ 0 & 0 & 0 & \mu^{\mathrm{visc}} \end{bmatrix} \tag{A25}$$

$$\tilde{\mathbf{K}}_{22} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & \chi^{\text{visc}} & 0 \\ 0 & 0 & 0 & \mu^{\text{visc}} \end{bmatrix} \tag{A26}$$

$$\tilde{\mathbf{K}}_{33} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & \mu^{\text{visc}} & 0 \\ 0 & 0 & 0 & \chi^{\text{visc}} \end{bmatrix} \tag{A27}$$

$$\tilde{\mathbf{K}}_{12} = \tilde{\mathbf{K}}_{21}^{\text{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda^{\text{visc}} & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{A28}$$

$$\tilde{\mathbf{K}}_{13} = \tilde{\mathbf{K}}_{31}^{\text{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^{\text{visc}} \\ 0 & 0 & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 & 0 \end{bmatrix} \tag{A29}$$

$$\tilde{\mathbf{K}}_{23} = \tilde{\mathbf{K}}_{32}^{\text{T}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^{\text{visc}} \\ 0 & 0 & \mu^{\text{visc}} & 0 \end{bmatrix} \tag{A30}$$

A.2. Coefficient matrices and vectors for the segregated formulation of a thermally perfect isothermal fluid

The coefficient matrices for the segregated formulation based on pressure primitive variables are as follows:

$$\begin{aligned}\mathbf{M}_{\rho\rho ab} = {} & \frac{1}{\Delta t}\int_{\Omega^e} N_a^e \mathbf{A}_{0\rho\rho} N_b^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_a^e \mathbf{A}_{i\rho\rho} N_{b,i}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{K}_{ij\rho\rho} N_{b,j}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{A}_i \boldsymbol{\tau} \mathbf{A}_j N_{b,j}^e \,\mathrm{d}\Omega \Big|_{\rho\rho} \\ & + \int_{\Omega^e} v^e g^{ij} N_{a,i}^e \mathbf{A}_{0\rho\rho} N_{b,j}^e \,\mathrm{d}\Omega \end{aligned} \tag{A31}$$

$$\begin{aligned}\mathbf{M}_{uuab} = {} & \frac{1}{\Delta t}\int_{\Omega^e} N_a^e \mathbf{A}_{0uu} N_b^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_a^e \mathbf{A}_{iuu} N_{b,i}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{K}_{ijuu} N_{b,j}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{A}_i \boldsymbol{\tau} \mathbf{A}_j N_{b,j}^e \,\mathrm{d}\Omega \Big|_{uu} \\ & + \int_{\Omega^e} v^e g^{ij} N_{a,i}^e \mathbf{A}_{0uu} N_{b,j}^e \,\mathrm{d}\Omega \end{aligned} \tag{A32}$$

$$\begin{aligned}\mathbf{M}_{\rho uab} = {} & \frac{1}{\Delta t}\int_{\Omega^e} N_a^e \mathbf{A}_{0\rho u} N_b^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_a^e \mathbf{A}_{i\rho u} N_{b,i}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{K}_{ij\rho u} N_{b,j}^e \,\mathrm{d}\Omega \\ & + \int_{\Omega^e} N_{a,i}^e \mathbf{A}_i \boldsymbol{\tau} \mathbf{A}_j N_{b,j}^e \,\mathrm{d}\Omega \Big|_{\rho u} \\ & + \int_{\Omega^e} v^e g^{ij} N_{a,i}^e \mathbf{A}_{0\rho u} N_{b,j}^e \,\mathrm{d}\Omega \end{aligned} \tag{A33}$$

$$\mathbf{M}_{u\rho ab} = \frac{1}{\Delta t}\int_{\Omega^e} N_a^e \mathbf{A}_{0u\rho} N_b^e \,\mathrm{d}\Omega + \int_{\Omega^e} N_a^e \mathbf{A}_{iu\rho} N_{b,i}^e \,\mathrm{d}\Omega + \int_{\Omega^e} N_{a,i}^e \mathbf{K}_{iju\rho} N_{b,j}^e \,\mathrm{d}\Omega + \int_{\Omega^e} N_{a,i}^e \mathbf{A}_i \boldsymbol{\tau} \mathbf{A}_j N_{b,j}^e \,\mathrm{d}\Omega\Big|_{u\rho} + \int_{\Omega^e} v^e g^{ij} N_{a,i}^e \mathbf{A}_{0u\rho} N_{b,j}^e \,\mathrm{d}\Omega \tag{A34}$$

The above expressions have been written as a function sub-matrices, which are explicited below. For the mass conservation equation these are:

$$\mathbf{A}_{0\rho\rho} = [\rho\beta_T] \tag{A35}$$

$$\mathbf{A}_{1\rho\rho} = [\rho\beta_T u_1] \tag{A36}$$

$$\mathbf{A}_{2\rho\rho} = [\rho\beta_T u_2] \tag{A37}$$

$$\mathbf{A}_{3\rho\rho} = [\rho\beta_T u_3] \tag{A38}$$

$$\mathbf{K}_{ij\rho\rho} = [0] \tag{A39}$$

and the sub-matrices to build the momentum equations are

$$\mathbf{A}_{0uu} = \rho \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A40}$$

$$\mathbf{A}_{1uu} = \rho \begin{bmatrix} 2u_1 & 0 & 0 \\ u_2 & u_1 & 0 \\ u_3 & 0 & u_1 \end{bmatrix} \tag{A41}$$

$$\mathbf{A}_{2uu} = \rho \begin{bmatrix} u_2 & u_1 & 0 \\ 0 & 2u_2 & 0 \\ 0 & u_3 & u_2 \end{bmatrix} \tag{A42}$$

$$\mathbf{A}_{3uu} = \rho \begin{bmatrix} u_3 & 0 & u_1 \\ 0 & u_3 & u_2 \\ 0 & 0 & 2u_3 \end{bmatrix} \tag{A43}$$

$$\mathbf{K}_{11uu} = \begin{bmatrix} \chi^{\text{visc}} & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 \\ 0 & 0 & \mu^{\text{visc}} \end{bmatrix} \tag{A44}$$

$$\mathbf{K}_{22uu} = \begin{bmatrix} \mu^{\text{visc}} & 0 & 0 \\ 0 & \chi^{\text{visc}} & 0 \\ 0 & 0 & \mu^{\text{visc}} \end{bmatrix} \tag{A45}$$

$$\mathbf{K}_{33uu} = \begin{bmatrix} \mu^{\text{visc}} & 0 & 0 \\ 0 & \mu^{\text{visc}} & 0 \\ 0 & 0 & \chi^{\text{visc}} \end{bmatrix} \tag{A46}$$

$$\mathbf{K}_{12uu} = \mathbf{K}_{21uu}^{\mathrm{T}} = \begin{bmatrix} 0 & \lambda^{\text{visc}} & 0 \\ \mu^{\text{visc}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A47}$$

$$\mathbf{K}_{13uu} = \mathbf{K}_{31uu}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & \lambda^{\text{visc}} \\ 0 & 0 & 0 \\ \mu^{\text{visc}} & 0 & 0 \end{bmatrix} \tag{A48}$$

$$\mathbf{K}_{23uu} = \mathbf{K}_{32uu}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \lambda^{\text{visc}} \\ 0 & \mu^{\text{visc}} & 0 \end{bmatrix} \tag{A49}$$

Finally, the cross coefficient sub-matrices are

$$\mathbf{A}_{0\rho u} = [0 \quad 0 \quad 0] \tag{A50}$$

$$\mathbf{A}_{1\rho u} = [\rho \quad 0 \quad 0] \tag{A51}$$

$$\mathbf{A}_{2\rho u} = [0 \quad \rho \quad 0] \tag{A52}$$

$$\mathbf{A}_{3\rho u} = [0 \quad 0 \quad \rho] \tag{A53}$$

$$\mathbf{K}_{ij\rho u} = [0 \quad 0 \quad 0] \tag{A54}$$

and

$$\mathbf{A}_{0u\rho} = \begin{bmatrix} \rho\beta_T u_1 \\ \rho\beta_T u_2 \\ \rho\beta_T u_3 \end{bmatrix} \tag{A55}$$

$$\mathbf{A}_{1u\rho} = \begin{bmatrix} \rho\beta_T u_1^2 + 1 \\ \rho\beta_T u_1 u_2 \\ \rho\beta_T u_1 u_3 \end{bmatrix} \tag{A56}$$

$$\mathbf{A}_{2u\rho} = \begin{bmatrix} \rho\beta_T u_1 u_2 \\ \rho\beta_T u_2^2 + 1 \\ \rho\beta_T u_2 u_3 \end{bmatrix} \tag{A57}$$

$$\mathbf{A}_{3u\rho} = \begin{bmatrix} \rho\beta_T u_1 u_3 \\ \rho\beta_T u_2 u_3 \\ \rho\beta_T u_3^2 + 1 \end{bmatrix} \tag{A58}$$

$$\mathbf{K}_{iju\rho} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{A59}$$

ACKNOWLEDGEMENTS

The authors acknowledge conversations with T.J.R. Hughes. This work has been partially sponsored by the Ministerio de Educación y Ciencia, Spain and by the University of the Basque Country, Spain project 1/UPV/EHU 00145.345-EA-8246/2000.

REFERENCES

1. Chorin AJ. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics* 1967; **2**:12–26.
2. Reddy JN. On penalty function methods in the finite-element analysis of flow problems. *International Journal for Numerical Methods in Fluids* 1982; **2**:151–151.
3. Hughes TJR, Liu WK, Brooks A. Finite element analysis of incompressible flows by the penalty function formulation. *Journal of Computational Physics* 1979; **30**:1–60.
4. Segal A. On the numerical solution of the Stokes equations using the finite element methods. *Computer Methods in Applied Mechanics and Engineering* 1979; **19**:165–185.
5. Fortin M, Fortin A. A generalization of Uzawa's algorithm for the solution of the Navier–Stokes equations. *Communications in Applied Numerical Methods* 1985; **1**:205–208.
6. Simo JC, Armero F. Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier–Stokes and Euler equations. *Computer Methods in Applied Mechanics and Engineering* 1994; **111**: 111–154.

7. Benim AC, Zinser W. A segregated formulation of Navier–Stokes equations with finite elements. *Computer Methods in Applied Mechanics and Engineering* 1986; **57**:223–237.
8. Rice JG, Schnipke RJ. An equal-order velocity–pressure formulation that does not exhibit spurious pressure modes. *Computer Methods in Applied Mechanics and Engineering* 1986; **58**:135–149.
9. Shaw CT. Using a segregated finite element scheme to solve the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1991; **12**:81–92.
10. Haroutunian V, Engelman MS, Hasbani I. Segregated finite element algorithms for the numerical solution of large-scale incompressible flow problems. *International Journal for Numerical Methods in Fluids* 1993; **17**:323–348.
11. Taylor C, Hood P. A numerical solution of the Navier–Stokes equations using the finite element methods. *Computer and Fluids* 1973; **1**:73–100.
12. Huyakorn RS, Taylor C, Lee RL, Gresho PM. A comparison of various mixed-interpolation finite elements in the velocity–pressure formulation of the Navier–Stokes equations. *Computer and Fluids* 1978; **6**:25–35.
13. Lee RL, Gresho PM, Sani RL. Smoothing techniques for certain primitive variable solutions of the Navier–Stokes equations. *International Journal for Numerical Methods in Engineering* 1979; **14**:1785–1804.
14. Gunzburger MD. *Finite Element Methods for Viscous Incompressible Flows*. Academic Press: New York, 1989.
15. Brezzi F, Fortin M. *Mixed and Hybrid Finite Element Methods*. Springer: New York, 1991.
16. Gresho PM, Sani RL. *Incompressible Flow and the Finite Element Method*: *Advection–Diffusion and Isothermal Laminar Flow*. Wiley: England, 1998.
17. Brooks AN, Hughes TJR. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.
18. Franca LP, Frey SL. Stabilized finite element methods: II. The incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1992; **99**:209–233.
19. Taylor CA, Hughes TJR, Zarins CK. Finite element modeling of blood flow in arteries. *Computer Methods in Applied Mechanics and Engineering* 1998; **158**:155–196.
20. Franca LP, Hughes TJR. Convergence analyses of Galerkin least-squares methods for symmetric advective–diffusive forms of the Stokes and incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1993; **105**:285–298.
21. Hauke G, Hughes TJR. A unified approach to compressible and incompressible flows. *Computer Methods in Applied Mechanics and Engineering* 1994; **113**:389–395.
22. Hauke G, Hughes TJR. A comparative study of different sets of variables for solving compressible and incompressible flows. *Computer Methods in Applied Mechanics and Engineering* 1998; **153**:1–44.
23. Zienkiewicz OC, Wu J. Incompressibility without tears—how to avoid restrictions of mixed formulation. *International Journal for Numerical Methods in Engineering* 1991; **32**:1189–1203.
24. Zienkiewicz OC, Codina R. A general algorithm for compressible and incompressible flow—Part I. The split, characteristic-based scheme. *International Journal for Numerical Methods in Fluids* 1995; **20**:869–885.
25. Zienkiewicz OC, Morgan K, Satya Sai BVK, Codina R, Vasquez M. A general algorithm for compressible and incompressible flow—Part II. Tests on the explicit form. *International Journal for Numerical Methods in Fluids* 1995; **20**:887–913.
26. Codina R, Vazquez M, Zienkiewicz OC. A general algorithm for compressible and incompressible flow—Part III. The semi-implicit form. *International Journal for Numerical Methods in Fluids* 1998; **27**:13–32.
27. Zienkiewicz OC, Wu J. A general explicit or semi-explicit algorithm for compressible and incompressible flows. *International Journal for Numerical Methods in Engineering* 1992; **35**:457–479.
28. Hughes TJR, Tezduyar TE. Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Computer Methods in Applied Mechanics and Engineering* 1984; **45**: 217–284.
29. Hughes TJR, Franca LP, Mallet M. A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier–Stokes equations and the second law of thermodynamics. *Computer Methods in Applied Mechanics and Engineering* 1986; **54**:223–234.
30. Hughes TJR, Mallet M, Mizukami A. A new finite element formulation for computational fluid dynamics: II. Beyond SUPG. *Computer Methods in Applied Mechanics and Engineering* 1986; **54**:341–355.
31. Hughes TJR, Mallet M. A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advection-diffusion systems. *Computer Methods in Applied Mechanics and Engineering* 1986; **58**:305–328.
32. Hughes TJR, Franca LP, Mallet M. A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multidimensional advective–diffusive systems. *Computer Methods in Applied Mechanics and Engineering* 1987; **63**:97–112.
33. Hughes TJR, Franca LP, Hulbert G. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations. *Computer Methods in Applied Mechanics and Engineering* 1989; **73**:173–189.

34. Shakib F, Hughes TJR. A new finite element formulation for computational fluid dynamics: IX. Fourier analysis of space–time Galerkin/least-squares algorithms. *Computer Methods in Applied Mechanics and Engineering* 1991; **87**:35–58.
35. Shakib F, Hughes TJR, Johan Z. A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1991; **89**:141–219.
36. Brueckner FP, Heinrich JC. Petrov–Galerkin finite element model for compressible flows. *International Journal for Numerical Methods in Engineering* 1991; **32**:255–274.
37. Le Beau GJ, Ray SE, Aliabadi SK, Tezduyar TE. SUPG finite element computation of compressible flows with the entropy and conservation variables formulations. *Computer Methods in Applied Mechanics and Engineering* 1993; **104**:397–422.
38. Hansbo P. Explicit streamline diffusion finite element methods for the compressible Euler equations in conservation variables. *Journal of Computational Physics* 1993; **109**:274–288.
39. Hughes TJR, Hauke G, Jansen K, Johan Z. Stabilized finite element methods in fluids: inspirations, origins, status and recent developments. In *A Book Dedicated to Robert L. Taylor*, Hughes TJR, Oñate E, Zienkiewicz OC (eds). CIMNE: Barcelona, 1994.
40. Soulaimani A, Fortin M. Finite element solution of compressible viscous flows using conservative variables. *Computer Methods in Applied Mechanics and Engineering* 1994; **118**:319–350.
41. Friedriechs KO, Lax PD. Systems of conservation equations with a convex extension. *Proceedings of the National Academy of Sciences of the United States of America* 1971; **68**:1686–1688.
42. Godunov SK. The problem of a generalized solution in the theory of quasi-linear equations and in gas dynamics. *Russian Mathematical Surveys* 1962; **17**:145–156.
43. Mock M. Systems of conservation laws of mixed type. *Journal of Differential Equations* 1980; **37**:70–88.
44. Harten A. On the symmetric form of systems of conservation laws with entropy. *Journal of Computational Physics* 1983; **49**:151–164.
45. Chalot F, Hughes TJR, Shakib F. Symmetrization of conservation laws with entropy for high-temperature hypersonic computations. *Computer Systems in Engineering* 1990; **1**:495–521.
46. Landaberea A. Un método segregado para la resolución de las ecuaciones de dinámica de fluidos mediante el método de los elementos finitos. *Ph.D. Thesis*, Universidad del Pais Vasco, 2002.
47. Lesaint P, Raviart PA. On a finite element method for solving the neutron transport equation. In *Mathematical Aspects of Finite Elements in Partial Differential Equations*, de Boor C (ed.). Academic Press: New York, 1974.
48. Codina R. On stabilized finite element methods for linear systems of convection-diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering* 2000; **188**(1–3):61–82.
49. Hauke G. Simple stabilizing matrices for the computation of compressible flows in primitive variables. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:6881–6893.
50. Franca LP, Frey SL, Hughes TJR. Stabilized finite element methods: I. Application to the advective–diffusive model. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**:253–276.
51. Hughes TJR, Mallet M. A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective–diffusive systems. *Computer Methods in Applied Mechanics and Engineering* 1986; **58**:329–336.
52. Jansen K, Collins S, Whiting C, Shakib F. A better consistency for low-order stabilized finite element methods. *Computer Methods in Applied Mechanics and Engineering* 1999; **174**:153–170.
53. Hauke G. A unified approach to compressible and incompressible flows and a new entropy-consistent formulation of the *k*-epsilon model. *Ph.D. Thesis*, Division of Applied Mechanics, Stanford University, 1995.
54. Droux JJ, Hughes TJR. A boundary integral modification of the Galerkin least squares formulation for the Stokes problem. *Computer Methods in Applied Mechanics and Engineering* 1994; **113**:173–182.
55. Illinca F, Hétu JF, Pelletier D. On stabilized finite element formulations for incompressible advective–diffusive transport and fluid flow problems. *Computer Methods in Applied Mechanics and Engineering* 2000; **188**: 235–255.
56. Ghia U, Ghia KN, Shin CT. High-Re solutions for incompressible flow using the Navier–Stokes equations and multigrid method. *Journal of Computational Physics* 1982; **48**:387–441.
57. Zienkiewicz OC, Satya Sai BVK, Morgan K, Codina R. Split, characteristic based semi-implicit algorithm for laminar/turbulent incompressible flows. *International Journal for Numerical Methods in Fluids* 1996; **23**: 787–809.
58. Denham MK, Patrick MM. Laminar flow over a downstream-facing step in a two dimensional flow channel. *Transactions of the Institution of Chemical Engineers* 1974; **52**:361–367.

