Applications of Computational Mechanics in Structures and Fluids
(eds. S.R. Idelsohn and V. Sonzogni), CIMNE, Barcelona (2005) 1-19.

APPLICATIONS OF COMPUTATIONAL MECHANICS
IN STRUCTURES AND FLUIDS
S.R. Idelsohn and V. Sozogni (Eds.)

CALCULATION OF THE STABILIZATION PARAMETERS IN FINITE ELEMENT FORMULATIONS OF FLOW PROBLEMS

Tayfun E. Tezduyar

Mechanical Engineering, Rice University – MS 321
6100 Main Street, Houston, TX 77005, USA
e-mail: tezduyar@rice.edu

Abstract. This is an overview of the calculation techniques we developed for the stabilization parameters used in the streamline-upwind/Petrov-Galerkin (SUPG) and pressure-stabilizing/Petrov-Galerkin (PSPG) methods. The SUPG and PSPG methods are used extensively in finite element formulations, including the interface-tracking and interface-capturing techniques we developed for computation of flow problems with moving boundaries and interfaces. The stabilization parameters described here are designed for the semi-discrete and space–time formulations of the advection–diffusion and Navier–Stokes equations. Some of these parameters are calculated with the element-level matrices and vectors. Some others are calculated with the local length scales for the advection- and diffusion-dominated limits.

Key words: Stabilization parameters, Element length scales, SUPG formulation, PSPG formulation, Space–time formulation

1 Introduction

Stabilized formulations, such as the streamline-upwind/Petrov-Galerkin (SUPG) [1, 2] and pressure-stabilizing/Petrov-Galerkin (PSPG) [3] methods, are used extensively in finite element computation of fluid mechanics problems. The stabilization methods prevent numerical oscillations and other instabilities in solving problems with advection-dominated flows and when using equal-order interpolation functions for velocity and pressure. The SUPG and PSPG stabilizations are essential components of the the interface-tracking and interface-capturing techniques we developed in recent years (see [3, 4, 5]) for computation of flow problems with moving boundaries and interfaces. Interface-tracking techniques require meshes that move to "track" the interfaces as the flow evolves. The Deforming-Spatial-Domain/Stabilized Space–Time (DSD/SST) formulation [3] is an example of the interface-tracking techniques. In interface-capturing techniques, on the other hand, the computations are based on meshes that are typically not moving or deforming. An interface function, marking the location of the interface and governed by a time-dependent advection equation, is computed to "capture" the interface over the non-moving mesh. In this technique, stabilized semi-discrete formulations are used for both the Navier–Stokes equations and the time-dependent advection equation.

The SUPG and PSPG formulations include stabilization parameters that are called by most researchers "τ"s. These parameters play an important role in determining the accuracy of the formulation. The parameters involve a measure of the local length scale (or "element length") and other parameters such as the local Reynolds and Courant numbers. Early versions of τs in the context of incompressible and compressible flows were proposed in [1, 2], which were followed by those introduced in [6] and subsequently reported SUPG and PSPG methods. A number of τs, dependent upon spatial and temporal discretizations, were introduced and tested in [7]. Later, τs which are applicable to higher-order elements were proposed in [8].

Calculating the τs from the element-level matrices and vectors was first reported in [9]. These definitions are expressed in terms of the ratios of the norms of the relevant matrices or vectors and take into account the local length scales, advection field and the element-level Reynolds number. With this approach, a τ can be calculated for each element, or even for each element node or degree of freedom or element equation. Certain variations and complements of these new τs were described in [10, 11, 12, 13].

In later sections, we will describe, for the semi-discrete and space–time formulations and for the advection–diffusion and Navier–Stokes equations, these new ways of calculating the stabilization parameters.

2 Governing Equations

Let $\Omega_t \subset I\!R^{n_{sd}}$ be the spatial fluid mechanics domain with boundary Γ_t at time $t \in (0, T)$, where the subscript t indicates the time-dependence of the spatial domain. The Navier–Stokes equations of incompressible flows can be written on Ω_t and $\forall t \in (0, T)$ as

$$\rho(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla}\mathbf{u} - \mathbf{f}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = 0, \tag{1}$$

$$\boldsymbol{\nabla} \cdot \mathbf{u} = 0, \tag{2}$$

where ρ, $\mathbf{u}$ and $\mathbf{f}$ are the density, velocity and the external force, and $\boldsymbol{\sigma}$ is the stress tensor:

$$\boldsymbol{\sigma}(p, \mathbf{u}) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\mathbf{u}), \quad \boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2}((\boldsymbol{\nabla}\mathbf{u}) + (\boldsymbol{\nabla}\mathbf{u})^T). \tag{3}$$

Here p is pressure, $\mathbf{I}$ is the identity tensor, $\mu = \rho\nu$ is viscosity, ν is the kinematic viscosity, and $\boldsymbol{\varepsilon}(\mathbf{u})$ is the strain-rate tensor. The essential and natural boundary conditions for Eq. (1) are represented as

$$\mathbf{u} = \mathbf{g} \text{ on } (\Gamma_{\mathrm{t}})_{\mathrm{g}}, \qquad \mathbf{n} \cdot \boldsymbol{\sigma} = \mathbf{h} \text{ on } (\Gamma_{\mathrm{t}})_{\mathrm{h}}, \tag{4}$$

where $(\Gamma_t)_g$ and $(\Gamma_t)_h$ are complementary subsets of the boundary Γ_t, $\mathbf{n}$ is the unit normal vector, and $\mathbf{g}$ and $\mathbf{h}$ are given functions. A divergence-free velocity field $\mathbf{u}_0(\mathbf{x})$ is specified as the initial condition.

If the problem does not involve any moving boundaries or interfaces, the spatial domain does not need to change with respect to time, and the subscript t can be dropped from Ω_t and Γ_t. This might be the case even for flows with moving boundaries and interfaces, if in the formulation used the spatial domain is not defined to be the part of the space occupied by the fluid(s). For example, we can have a fixed spatial domain, and model the fluid–fluid interfaces by assuming that the domain is occupied by two immiscible fluids, A and B, with densities ρ_A and ρ_B and viscosities μ_A and μ_B. In modeling a free-surface problem where Fluid B is irrelevant, we assign a sufficiently low density to Fluid B. An interface function ϕ serves as a marker identifying Fluid A and B with the definition $\phi = \{1$ for Fluid A and 0 for Fluid B$\}$. The interface between the two fluids is approximated to be at $\phi = 0.5$. In this context, ρ and μ are defined as

$$\rho = \phi\rho_A + (1-\phi)\rho_B, \quad \mu = \phi\mu_A + (1-\phi)\mu_B. \tag{5}$$

The evolution of the interface function ϕ, and therefore the motion of the interface, is governed by a time-dependent advection equation, written on Ω and $\forall t \in (0, T)$ as

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla}\phi = 0. \tag{6}$$

As a generalization of Eq. (6), let us consider over a domain Ω with boundary Γ the following time-dependent advection–diffusion equation, written on Ω and $\forall t \in (0, T)$ as

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla}\phi - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla}\phi) = 0, \tag{7}$$

where ϕ represents the quantity being transported (e.g. temperature), and ν is the diffusivity. The essential and natural boundary conditions associated with Eq. (7) are represented as

$$\phi = \mathrm{g} \text{ on } \Gamma_{\mathrm{g}}, \qquad \mathbf{n} \cdot \nu \boldsymbol{\nabla}\phi = \mathrm{h} \text{ on } \Gamma_{\mathrm{h}}. \tag{8}$$

A function $\phi_0(\mathbf{x})$ is specified as the initial condition.

3 Stabilized Formulation for Advection–Diffusion Equation

Let us assume that we have constructed some suitably-defined finite-dimensional trial solution and test function spaces $\mathcal{S}_\phi^h$ and $\mathcal{V}_\phi^h$. The stabilized finite element formulation of Eq. (7) can then be written as follows: find $\phi^h \in \mathcal{S}_\phi^h$ such that $\forall w^h \in \mathcal{V}_\phi^h$:

$$\int_\Omega w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla}\phi^h \right) d\Omega + \int_\Omega \boldsymbol{\nabla} w^h \cdot \nu \boldsymbol{\nabla}\phi^h d\Omega - \int_{\Gamma_\mathrm{h}} w^h \mathrm{h}^h d\Gamma$$
$$+ \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{\text{SUPG}} \mathbf{u}^h \cdot \boldsymbol{\nabla} w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla}\phi^h - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla}\phi^h \right) \right) d\Omega = 0. \tag{9}$$

Here n_{el} is the number of elements, Ω^e is the element domain, and τ_{SUPG} is the SUPG stabilization parameter.

4 Element-Matrix-Based Stabilization Parameters for Advection–Diffusion Equation

Let us use the notation $\mathbf{b} : \int_{\Omega^e} (\ldots) d\Omega : \mathbf{b}_{\mathrm{V}}$ to denote the element-level matrix $\mathbf{b}$ and element-level vector $\mathbf{b}_{\mathrm{V}}$ corresponding to the element-level integration term $\int_{\Omega^e} (\ldots) d\Omega$. We define the following element-level matrices and vectors:

$$\mathbf{m} : \quad \int_{\Omega^e} w^h \frac{\partial \phi^h}{\partial t} d\Omega \quad : \mathbf{m}_{\mathrm{V}}, \tag{10}$$

$$\mathbf{c} : \quad \int_{\Omega^e} w^h \mathbf{u}^h \cdot \boldsymbol{\nabla} \phi^h d\Omega \quad : \mathbf{c}_{\mathrm{V}}, \tag{11}$$

$$\mathbf{k} : \quad \int_{\Omega^e} \boldsymbol{\nabla} w^h \cdot \nu \boldsymbol{\nabla} \phi^h d\Omega \quad : \mathbf{k}_{\mathrm{V}}, \tag{12}$$

$$\tilde{\mathbf{k}} : \quad \int_{\Omega^e} \mathbf{u}^h \cdot \boldsymbol{\nabla} w^h \ \mathbf{u}^h \cdot \boldsymbol{\nabla} \phi^h d\Omega \quad : \tilde{\mathbf{k}}_{\mathrm{V}}, \tag{13}$$

$$\tilde{\mathbf{c}} : \quad \int_{\Omega^e} \mathbf{u}^h \cdot \boldsymbol{\nabla} w^h \frac{\partial \phi^h}{\partial t} d\Omega \quad : \tilde{\mathbf{c}}_{\mathrm{V}}. \tag{14}$$

We define the element-level Reynolds and Courant numbers as follows:

$$Re = \frac{\|\mathbf{u}^h\|^2}{\nu} \frac{\|\mathbf{c}\|}{\|\tilde{\mathbf{k}}\|}, \tag{15}$$

$$Cr_u = \frac{\Delta t}{2} \frac{\|\mathbf{c}\|}{\|\mathbf{m}\|}, \tag{16}$$

where $\|\mathbf{b}\|$ is the norm of matrix $\mathbf{b}$.

The components of element-matrix-based τ_{SUPG} are defined as follows:

$$\tau_{\mathrm{S1}} = \frac{\|\mathbf{c}\|}{\|\tilde{\mathbf{k}}\|}, \tag{17}$$

$$\tau_{\mathrm{S2}} = \frac{\Delta t}{2} \frac{\|\mathbf{c}\|}{\|\tilde{\mathbf{c}}\|}, \tag{18}$$

$$\tau_{\mathrm{S3}} = \tau_{\mathrm{S1}} Re = \left(\frac{\|\mathbf{c}\|}{\|\tilde{\mathbf{k}}\|} \right) Re. \tag{19}$$

We construct τ_{SUPG} from its components by using the expression

$$\tau_{\mathrm{SUPG}} = \left(\frac{1}{\tau_{\mathrm{S1}}^r} + \frac{1}{\tau_{\mathrm{S2}}^r} + \frac{1}{\tau_{\mathrm{S3}}^r} \right)^{-\frac{1}{r}}, \tag{20}$$

which is based on the inverse of τ_{SUPG} being defined as the r-norm of the vector with components $\frac{1}{\tau_{\mathrm{S1}}}$, $\frac{1}{\tau_{\mathrm{S2}}}$ and $\frac{1}{\tau_{\mathrm{S3}}}$. We note that the higher the integer r is, the sharper the switching between τ_{S1}, τ_{S2} and τ_{S3} becomes. This "*r-switch*" was introduced in [9]. Typically, we set $r = 2$.

The components of the element-vector-based τ_{SUPG} are defined as follows:

$$\tau_{\mathrm{SV1}} = \frac{\|\mathbf{c}_{\mathrm{V}}\|}{\|\tilde{\mathbf{k}}_{\mathrm{V}}\|}, \tag{21}$$

$$\tau_{\mathrm{SV2}} = \frac{\|\mathbf{c}_{\mathrm{V}}\|}{\|\tilde{\mathbf{c}}_{\mathrm{V}}\|}, \tag{22}$$

$$\tau_{\mathrm{SV3}} = \tau_{\mathrm{SV1}} Re = \left(\frac{\|\mathbf{c}_{\mathrm{V}}\|}{\|\tilde{\mathbf{k}}_{\mathrm{V}}\|} \right) Re. \tag{23}$$

With these three components,

$$(\tau_{\text{SUPG}})_{\text{V}} = \left(\frac{1}{\tau^r_{\text{SV1}}} + \frac{1}{\tau^r_{\text{SV2}}} + \frac{1}{\tau^r_{\text{SV3}}} \right)^{-\frac{1}{r}} . \tag{24}$$

Remark 1 *The definition of* τ_{SUPG} *given by Eqs. (21)-(24) can be seen as a nonlinear definition because it depends on the solution. However, in marching from time level* n *to* $n+1$ *the element vectors can be evaluated at level* n*. This might be preferable in some cases, as it spares us from ending up with a nonlinear semi-discrete equation system.*

5 Stabilized Formulation for Navier–Stokes Equations

Let us assume that we have some suitably-defined finite-dimensional trial solution and test function spaces for velocity and pressure: $\mathcal{S}^h_{\mathbf{u}}$, $\mathcal{V}^h_{\mathbf{u}}$, $\mathcal{S}^h_p$ and $\mathcal{V}^h_p = \mathcal{S}^h_p$. The stabilized finite element formulation of Eqs. (1)-(2) can then be written as follows: find $\mathbf{u}^h \in \mathcal{S}^h_{\mathbf{u}}$ and $p^h \in \mathcal{S}^h_p$ such that $\forall \mathbf{w}^h \in \mathcal{V}^h_{\mathbf{u}}$ and $q^h \in \mathcal{V}^h_p$:

$$\begin{aligned}
& \int_\Omega \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h - \mathbf{f}^h \right) d\Omega + \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h) d\Omega - \int_{\Gamma_\text{h}} \mathbf{w}^h \cdot \mathbf{h}^h d\Gamma \\
+ & \int_\Omega q^h \boldsymbol{\nabla} \cdot \mathbf{u}^h d\Omega + \sum_{e=1}^{n_{el}} \int_{\Omega^e} \frac{1}{\rho} [\tau_{\text{SUPG}} \rho \mathbf{u}^h \cdot \nabla \mathbf{w}^h + \tau_{\text{PSPG}} \nabla q^h] \cdot \left[\text{Ł}(p^h, \mathbf{u}^h) - \rho \mathbf{f}^h \right] d\Omega \\
+ & \sum_{e=1}^{n_{el}} \int_{\Omega^e} \nu_{\text{LSIC}} \boldsymbol{\nabla} \cdot \mathbf{w}^h \rho \boldsymbol{\nabla} \cdot \mathbf{u}^h d\Omega \ = \ 0,
\end{aligned} \tag{25}$$

where

$$\text{Ł}(q^h, \mathbf{w}^h) = \rho \left(\frac{\partial \mathbf{w}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h \right) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(q^h, \mathbf{w}^h). \tag{26}$$

Here τ_{PSPG} and ν_{LSIC} are the PSPG and LSIC (least-squares on incompressibility constraint) stabilization parameters.

6 Element-Matrix-Based Stabilization Parameters for Navier–Stokes Equations

We define the following element-level matrices and vectors:

$$\mathbf{m} : \quad \int_{\Omega^e} \mathbf{w}^h \cdot \rho \frac{\partial \mathbf{u}^h}{\partial t} d\Omega \quad : \mathbf{m}_{\text{V}}, \tag{27}$$

$$\mathbf{c} : \quad \int_{\Omega^e} \mathbf{w}^h \cdot \rho (\mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h) d\Omega \quad : \mathbf{c}_{\text{V}}, \tag{28}$$

$$\mathbf{k} : \quad \int_{\Omega^e} \boldsymbol{\varepsilon}(\mathbf{w}^h) : 2\mu \boldsymbol{\varepsilon}(\mathbf{u}^h) d\Omega \quad : \mathbf{k}_{\text{V}}, \tag{29}$$

$$\mathbf{g} : \quad \int_{\Omega^e} (\boldsymbol{\nabla} \cdot \mathbf{w}^h) p^h d\Omega \quad : \mathbf{g}_{\text{V}}, \tag{30}$$

$$\mathbf{g}^{\text{T}} : \quad \int_{\Omega^e} q^h (\boldsymbol{\nabla} \cdot \mathbf{u}^h) d\Omega \quad : \mathbf{g}^{\text{T}}_{\text{V}}, \tag{31}$$

$$\tilde{\mathbf{k}} : \quad \int_{\Omega^e} (\mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h) \cdot \rho (\mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h) d\Omega \quad : \tilde{\mathbf{k}}_{\text{V}}, \tag{32}$$

$$\tilde{\mathbf{c}}: \qquad \int_{\Omega^e} (\mathbf{u}^h \cdot \boldsymbol{\nabla}\mathbf{w}^h) \cdot \rho \frac{\partial \mathbf{u}^h}{\partial t} d\Omega \qquad : \tilde{\mathbf{c}}_{\mathrm{V}}, \tag{33}$$

$$\tilde{\gamma}: \qquad \int_{\Omega^e} (\mathbf{u}^h \cdot \boldsymbol{\nabla}\mathbf{w}^h) \cdot \boldsymbol{\nabla} p^h d\Omega \qquad : \tilde{\gamma}_{\mathrm{V}}, \tag{34}$$

$$\beta: \qquad \int_{\Omega^e} \boldsymbol{\nabla} q^h \cdot \frac{\partial \mathbf{u}^h}{\partial t} d\Omega \qquad : \beta_{\mathrm{V}}, \tag{35}$$

$$\gamma: \qquad \int_{\Omega^e} \boldsymbol{\nabla} q^h \cdot (\mathbf{u}^h \cdot \boldsymbol{\nabla}\mathbf{u}^h) d\Omega \qquad : \gamma_{\mathrm{V}}, \tag{36}$$

$$\theta: \qquad \int_{\Omega^e} \boldsymbol{\nabla} q^h \cdot \boldsymbol{\nabla} p^h d\Omega \qquad : \theta_{\mathrm{V}}, \tag{37}$$

$$\mathbf{e}: \qquad \int_{\Omega^e} (\boldsymbol{\nabla} \cdot \mathbf{w}^h) \rho (\boldsymbol{\nabla} \cdot \mathbf{u}^h) d\Omega \qquad : \mathbf{e}_{\mathrm{V}}. \tag{38}$$

Remark 2 *In the definition of the element-level matrices listed above, we assume that $\mathbf{u}^h$ appearing in the advective operator (i.e. in $\mathbf{u}^h \cdot \boldsymbol{\nabla}\mathbf{u}^h$ and $\mathbf{u}^h \cdot \boldsymbol{\nabla}\mathbf{w}^h$) is evaluated at time level n rather than $n+1$. The definition would essentially be the same if we, alternatively, assumed that it is evaluated at time level $n+1$ but nonlinear iteration level i rather than $i+1$. Except, in the first option, in the advective operator we use $(\mathbf{u}^h)_n$, whereas in the second option we use $(\mathbf{u}^h)^i_{n+1}$. The second option can be seen as a nonlinear definition. The first option might be preferable in some cases, as it spares us from another level of nonlinearity coming from the way τ is defined. In the definition of the element-level-vectors, we face the same choices in terms of the evaluation of $\mathbf{u}^h$ in the advective operator.*

The element-level Reynolds and Courant numbers are defined the same way as they were defined before, as given by Eqs. (15)-(16). The components of the element-matrix-based τ_{SUPG} are defined the same way as they were defined before, as given by Eqs. (17)-(19). τ_{SUPG} is constructed from its components the same way as it was constructed before, as given by Eq. (20). The components of the element-vector-based τ_{SUPG} are defined the same way as they were defined before, as given by Eqs. (21)-(23). The construction of $(\tau_{\mathrm{SUPG}})_{\mathrm{V}}$ is also the same as it was before, given by Eq. (24).

The components of the element-matrix-based τ_{PSPG} are defined as follows:

$$\tau_{\mathrm{P1}} = \frac{\|\mathbf{g}^{\mathrm{T}}\|}{\|\gamma\|}, \tag{39}$$

$$\tau_{\mathrm{P2}} = \frac{\Delta t}{2} \frac{\|\mathbf{g}^{\mathrm{T}}\|}{\|\beta\|}, \tag{40}$$

$$\tau_{\mathrm{P3}} = \tau_{\mathrm{P1}} Re = \left(\frac{\|\mathbf{g}^{\mathrm{T}}\|}{\|\gamma\|} \right) Re. \tag{41}$$

τ_{PSPG} is constructed from its components as follows:

$$\tau_{\mathrm{PSPG}} = \left(\frac{1}{\tau^r_{\mathrm{P1}}} + \frac{1}{\tau^r_{\mathrm{P2}}} + \frac{1}{\tau^r_{\mathrm{P3}}} \right)^{-\frac{1}{r}}. \tag{42}$$

The components of the element-vector-based τ_{PSPG} are defined as follows:

$$\tau_{\mathrm{PV1}} = \tau_{\mathrm{P1}}, \tag{43}$$

$$\tau_{\mathrm{PV2}} = \tau_{\mathrm{PV1}} \frac{\|\gamma_{\mathrm{V}}\|}{\|\beta_{\mathrm{V}}\|}, \tag{44}$$

$$\tau_{\mathrm{PV3}} = \tau_{\mathrm{PV1}} Re. \tag{45}$$

With these components,

$$(\tau_{\mathrm{PSPG}})_{\mathrm{V}} = \left(\frac{1}{\tau^r_{\mathrm{PV1}}} + \frac{1}{\tau^r_{\mathrm{PV2}}} + \frac{1}{\tau^r_{\mathrm{PV3}}} \right)^{-\frac{1}{r}}. \tag{46}$$

The element-matrix-based ν_{LSIC} is defined as follows:

$$\nu_{\mathrm{LSIC}} = \frac{\|\mathbf{c}\|}{\|\mathbf{e}\|}. \tag{47}$$

We define the element-vector-based ν_{LSIC} as:

$$(\nu_{\mathrm{LSIC}})_{\mathrm{V}} = \nu_{\mathrm{LSIC}}. \tag{48}$$

Remark 3 *We can also calculate a separate τ for each element node, or degree of freedom, or element equation. In that case, each component of τ would be calculated separately for each element node, or degree of freedom, or element equation. For this, we first represent an element matrix $\mathbf{b}$ in terms of its row matrices: $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n_{ex}}$ and an element vector $\mathbf{b}_V$ in terms of its subvectors: $(\mathbf{b}_V)_1, (\mathbf{b}_V)_2, \ldots, (\mathbf{b}_V)_{n_{ex}}$. If we want a separate τ for each element node, then $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n_{ex}}$ and $(\mathbf{b}_V)_1, (\mathbf{b}_V)_2, \ldots, (\mathbf{b}_V)_{n_{ex}}$ would be the row matrices and subvectors corresponding to each element node, with $n_{ex} = n_{en}$, where n_{en} is the number of element nodes. If we want a separate τ for each degree of freedom, then $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n_{ex}}$ and $(\mathbf{b}_V)_1, (\mathbf{b}_V)_2, \ldots, (\mathbf{b}_V)_{n_{ex}}$ would be the row matrices and subvectors corresponding to each degree of freedom, with $n_{ex} = n_{dof}$, where n_{dof} is the number of degrees of freedom. If we want a separate τ for each element equation, then $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n_{ex}}$ and $(\mathbf{b}_V)_1, (\mathbf{b}_V)_2, \ldots, (\mathbf{b}_V)_{n_{ex}}$ would be the row matrices and subvectors corresponding to each element equation, with $n_{ex} = n_{ee}$, where n_{ee} is the number of element equations. Based on this, the components of τ would be calculated using the norms of these row matrices or subvectors, instead of the element matrices or vectors. For example, a separate τ_{S1} or τ_{SV1} for each element node would be calculated by using the expression*

$$(\tau_{S1})_a = \frac{\|\mathbf{c}_a\|}{\|\tilde{\mathbf{k}}_a\|}, \quad a = 1, 2, \ldots, n_{en} \tag{49}$$

or

$$(\tau_{SV1})_a = \frac{\|(\mathbf{c}_V)_a\|}{\|(\tilde{\mathbf{k}}_V)_a\|}, \quad a = 1, 2, \ldots, n_{en}. \tag{50}$$

In flow computations, the τs calculated for the element nodes or element equations would be used in interpolating the values of τs at the integration points.

Remark 4 *The concept of calculating a separate τ for each element node or equation can be extended to calculating a separate τ for each global node or equation. This can be accomplished by first representing a global matrix or vector in terms of its row matrices or subvectors associated with the global nodes or equations, and then by calculating the components of τ using the norms of these global row matrices or subvectors. With this approach, applying the stabilization techniques described in this paper to element-free methods would become more direct.*

Remark 5 *We can also calculate a separate τ for each integration point by using for that integration point the ratios of the norms of the element matrices or vectors contributed by that integration point. For example, a separate τ_{S1} or τ_{SV1} for each element integration point l would be calculated by using the expression*

$$(\tau_{S1})_l = \frac{\|\mathbf{c}_l\|}{\|\tilde{\mathbf{k}}_l\|}, \quad l = 1, 2, \ldots, n_{int} \tag{51}$$

or

$$(\tau_{SV1})_l = \frac{\|(\mathbf{c}_V)_l\|}{\|(\tilde{\mathbf{k}}_V)_l\|}, \quad l = 1, 2, \ldots, n_{int}. \tag{52}$$

Here n_{int} is the number of integration points, $\mathbf{c}_l$ and $\tilde{\mathbf{k}}_l$ are the element matrices contributed by the integration point l, and $(\mathbf{c}_V)_l$ and $(\tilde{\mathbf{k}}_V)_l$ are the element vectors contributed by the integration point l.

Remark 6 *It was hinted in Remarks 1 and 2 that in marching from time level n to $n+1$, there are advantages in calculating the τs from the flow field at time level n. That is*

$$\tau \;\leftarrow\; \tau_n \;, \tag{53}$$

where τ is the stabilization parameter to be used in marching from time level n to $n+1$, and τ_n is the stabilization parameter calculated from the flow field at time level n. One of the main advantages in doing that, as it was pointed out in Remarks 1 and 2, is avoiding another level of nonlinearity coming from the way τs are defined. In general, we suggest making τs less dependent on short-term variations in the flow field. For this purpose, we propose a recursive time-averaging in determining the τs to be used in marching from time level n to $n+1$:

$$\tau \;\leftarrow\; z_1\,\tau_n \;+\; z_2\,\tau_{n-1} \;+\; (1 - z_1 - z_2)\,\tau \;, \tag{54}$$

where τ_n and τ_{n-1} are the stabilization parameters calculated from the flow field at time levels n and $n-1$, and the τ on the right-hand-side is the stabilization parameter that was used in marching from time level $n-1$ to n. The magnitudes and the number of the "averaging parameters" z_1, z_2, ... can be adjusted to create the desired outcome in terms of giving more weight to recently calculated τs or making the averaging closer to being a trailing average.

7 UGN-Based Parameters for Navier–Stokes Equations

For the purpose of comparison, we define here also the stabilization parameters that are based on an earlier definition of the length scale h [6]:

$$h_{\mathrm{UGN}} = 2\,\|\mathbf{u}^h\| \left(\sum_{a=1}^{n_{en}} |\mathbf{u}^h \cdot \boldsymbol{\nabla} N_a| \right)^{-1}, \tag{55}$$

where N_a is the interpolation function associated with node a. The stabilization parameters are defined as follows:

$$\tau_{\text{SUGN1}} = \frac{h_{\text{UGN}}}{2\|\mathbf{u}^h\|}, \tag{56}$$

$$\tau_{\text{SUGN2}} = \frac{\Delta t}{2}, \tag{57}$$

$$\tau_{\text{SUGN3}} = \frac{h_{\text{UGN}}^2}{4\nu}, \tag{58}$$

$$(\tau_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{\tau_{\text{SUGN1}}^r} + \frac{1}{\tau_{\text{SUGN2}}^r} + \frac{1}{\tau_{\text{SUGN3}}^r}\right)^{-\frac{1}{r}}, \tag{59}$$

$$(\tau_{\text{PSPG}})_{\text{UGN}} = (\tau_{\text{SUPG}})_{\text{UGN}}, \tag{60}$$

$$(\nu_{\text{LSIC}})_{\text{UGN}} = \frac{h_{\text{UGN}}}{2}\|\mathbf{u}^h\|\ z. \tag{61}$$

Here z is given as follows:

$$z = \begin{cases} \left(\frac{Re_{\text{UGN}}}{3}\right) & Re_{\text{UGN}} \leq 3, \\ 1 & Re_{\text{UGN}} > 3, \end{cases} \tag{62}$$

where $Re_{\text{UGN}} = \frac{\|\mathbf{u}^h\| h_{\text{UGN}}}{2\nu}$.

Comparisons between the performances of these earlier stabilization parameters and the ones proposed here can be found in [9]. These comparisons show that, especially for special element geometries, the performances are similar.

It was pointed out in [12] that the expression for τ_{SUGN1} can be written more directly as

$$\tau_{\text{SUGN1}} = \left(\sum_{a=1}^{n_{en}} |\mathbf{u}^h \cdot \boldsymbol{\nabla} N_a|\right)^{-1}, \tag{63}$$

and based on that, the expression for h_{UGN} can be written as

$$h_{\text{UGN}} = 2\|\mathbf{u}^h\|\ \tau_{\text{SUGN1}}. \tag{64}$$

A rationale for τ_{SUGN1} given by Eq. (63) was provided in [12].

8 Discontinuity-Capturing Directional Dissipation (DCDD)

As a potential alternative or complement to the LSIC stabilization, we proposed in [10, 11, 12] the Discontinuity-Capturing Directional Dissipation (DCDD) stabilization. In describing the DCDD stabilization, we first define the unit vectors $\mathbf{s}$ and $\mathbf{r}$:

$$\mathbf{s} = \frac{\mathbf{u}^h}{\|\mathbf{u}^h\|}, \qquad \mathbf{r} = \frac{\boldsymbol{\nabla}\|\mathbf{u}^h\|}{\|\ \boldsymbol{\nabla}\|\mathbf{u}^h\|\ \|}, \tag{65}$$

and the element-level matrices and vectors $\mathbf{c}_{\text{r}}$, $\tilde{\mathbf{k}}_{\text{r}}$, $(\mathbf{c}_{\text{r}})_{\text{V}}$, and $(\tilde{\mathbf{k}}_{\text{r}})_{\text{V}}$:

$$\mathbf{c}_{\text{r}} : \quad \int_{\Omega^e} \mathbf{w}^h \cdot \rho(\mathbf{r} \cdot \boldsymbol{\nabla}\mathbf{u}^h) d\Omega \quad : (\mathbf{c}_{\text{r}})_{\text{V}}\ , \tag{66}$$

$$\tilde{\mathbf{k}}_{\text{r}} : \quad \int_{\Omega^e} (\mathbf{r} \cdot \boldsymbol{\nabla}\mathbf{w}^h) \cdot \rho(\mathbf{r} \cdot \boldsymbol{\nabla}\mathbf{u}^h) d\Omega \quad : (\tilde{\mathbf{k}}_{\text{r}})_{\text{V}}\ . \tag{67}$$

Next, we define the "DCDD viscosity" ν_{DCDD} and the DCDD stabilization parameter τ_{DCDD}. The element-matrix-based and element-vector-based DCDD viscosities are:

$$\nu_{\text{DCDD}} = |\mathbf{r} \cdot \mathbf{u}^h| \frac{\|\mathbf{c}_{\text{r}}\|}{\|\tilde{\mathbf{k}}_{\text{r}}\|} , \tag{68}$$

$$(\nu_{\text{DCDD}})_{\text{V}} = |\mathbf{r} \cdot \mathbf{u}^h| \frac{\|(\mathbf{c}_{\text{r}})_{\text{V}}\|}{\|(\tilde{\mathbf{k}}_{\text{r}})_{\text{V}}\|} . \tag{69}$$

An approximate version of the expression given by Eq. (68) can be written as

$$\nu_{\text{DCDD}} = |\mathbf{r} \cdot \mathbf{u}^h| \frac{h_{\text{RGN}}}{2}, \tag{70}$$

where

$$h_{\text{RGN}} = 2 \left(\sum_{a=1}^{n_{en}} |\mathbf{r} \cdot \boldsymbol{\nabla} N_a| \right)^{-1} . \tag{71}$$

A different way of determining ν_{DCDD} can be expressed as

$$\nu_{\text{DCDD}} = \tau_{\text{DCDD}} \|\mathbf{u}^h\|^2, \tag{72}$$

where

$$\tau_{\text{DCDD}} = \frac{h_{\text{DCDD}}}{2u_{\text{cha}}} \frac{\| \boldsymbol{\nabla} \|\mathbf{u}^h\| \| h_{\text{DCDD}}}{u_{\text{ref}}} . \tag{73}$$

Here u_{ref} is a reference velocity (such as $\|\mathbf{u}^h\|$ at the inflow, or the difference between the estimated maximum and minimum values of $\|\mathbf{u}^h\|$), and u_{cha} is a characteristic velocity (such as u_{ref} or $\|\mathbf{u}^h\|$). We propose to set $u_{\text{cha}} = u_{\text{ref}}$. For h_{DCDD}, we can use the expression

$$h_{\text{DCDD}} = 2 \frac{\|\mathbf{c}_{\text{r}}\|}{\|\tilde{\mathbf{k}}_{\text{r}}\|} , \tag{74}$$

or the approximation

$$h_{\text{DCDD}} = h_{\text{RGN}} . \tag{75}$$

The DCDD stabilization is defined as

$$S_{\text{DCDD}} = \sum_{e=1}^{n_{el}} \int_{\Omega^e} \rho \boldsymbol{\nabla} \mathbf{w}^h : \left([\nu_{\text{DCDD}} \mathbf{r}\mathbf{r} - \boldsymbol{\kappa}_{\text{CORR}}] \cdot \boldsymbol{\nabla} \mathbf{u}^h \right) d\Omega , \tag{76}$$

where $\boldsymbol{\kappa}_{\text{CORR}}$ was defined in [10, 11, 12, 13] as

$$\boldsymbol{\kappa}_{\text{CORR}} = \nu_{\text{DCDD}} (\mathbf{r} \cdot \mathbf{s})^2 \mathbf{s}\mathbf{s} . \tag{77}$$

As a possible alternative, we propose

$$\boldsymbol{\kappa}_{\text{CORR}} = \nu_{\text{SUPG}} (\mathbf{r} \cdot \mathbf{s})^2 \mathbf{r}\mathbf{r} . \tag{78}$$

As two other possible alternatives, we propose

$$\boldsymbol{\kappa}_{\text{CORR}} = switch\left(\nu_{\text{SUPG}}\ ,\ \nu_{\text{DCDD}}(\mathbf{r}\cdot\mathbf{s})^2\right)\mathbf{ss}\ , \tag{79}$$

$$\boldsymbol{\kappa}_{\text{CORR}} = switch\left(\nu_{\text{DCDD}}\ ,\ \nu_{\text{SUPG}}(\mathbf{r}\cdot\mathbf{s})^2\right)\mathbf{rr}\ , \tag{80}$$

where the "*switch*" function is defined as the "*min*" function:

$$switch\,(\alpha,\beta) = min\,(\alpha,\beta) \tag{81}$$

or as the "*r-switch*" given in Section 4 :

$$switch\,(\alpha,\beta) = \left(\frac{1}{\alpha^r} + \frac{1}{\beta^r}\right)^{-\frac{1}{r}}\ . \tag{82}$$

Remark 7 *Remark 6 applies also to the calculation of* ν_{DCDD}.

9 UGN/RGN-Based Parameters for Navier–Stokes Equations

In [11, 12], we proposed to re-define τ_{PSPG} and provided the reason for doing that. We described how we re-define τ_{PSPG} by modifying the definitions of τ_{P3} and τ_{PV3} given by Eqs. (41) and (45). We proposed to accomplish that by using the expressions

$$\tau_{\text{P3}} = \tau_{\text{P1}}\ \frac{\|\mathbf{c}\|}{\nu\ \|\tilde{\mathbf{k}}_{\text{r}}\|}\ ,\qquad \tau_{\text{PV3}} = \tau_{\text{PV1}}\ \frac{\|\mathbf{c}\|}{\nu\ \|\tilde{\mathbf{k}}_{\text{r}}\|}\ , \tag{83}$$

or the approximations

$$\tau_{\text{P3}} = \tau_{\text{P1}}\ Re\left(\frac{h_{\text{RGN}}}{h_{\text{UGN}}}\right)^2\ ,\qquad \tau_{\text{PV3}} = \tau_{\text{PV1}}\ Re\left(\frac{h_{\text{RGN}}}{h_{\text{UGN}}}\right)^2\ . \tag{84}$$

In [11], we further stated that these modifications can also be applied to τ_{S3} and τ_{SV3} given by Eqs. (19) and (23). In [12], we wrote those expressions explicitly as follows:

$$\tau_{\text{S3}} = \tau_{\text{S1}}\ \frac{\|\mathbf{c}\|}{\nu\ \|\tilde{\mathbf{k}}_{\text{r}}\|}\ ,\qquad \tau_{\text{SV3}} = \tau_{\text{SV1}}\ \frac{\|\mathbf{c}\|}{\nu\ \|\tilde{\mathbf{k}}_{\text{r}}\|}\ , \tag{85}$$

$$\tau_{\text{S3}} = \tau_{\text{S1}}\ Re\left(\frac{h_{\text{RGN}}}{h_{\text{UGN}}}\right)^2\ ,\qquad \tau_{\text{SV3}} = \tau_{\text{SV1}}\ Re\left(\frac{h_{\text{RGN}}}{h_{\text{UGN}}}\right)^2\ . \tag{86}$$

We noted in [12] that if we are dealing with just an advection–diffusion equation, rather than the Navier–Stokes equations of incompressible flows, then the definition of the unit vector $\mathbf{r}$ changes as follows:

$$\mathbf{r} = \frac{\boldsymbol{\nabla}|\phi^h|}{\|\ \boldsymbol{\nabla}|\phi^h|\ \|}\ . \tag{87}$$

We also proposed in [12] to re-define τ_{SUGN3} given by Eq. (58) as follows:

$$\tau_{\text{SUGN3}} = \frac{h_{\text{RGN}}^2}{4\nu}\ . \tag{88}$$

Furthermore, we proposed in [12] to replace $(\nu_{\mathrm{LSIC}})_{\mathrm{UGN}}$ given by Eq. (61) as follows:

$$(\nu_{\mathrm{LSIC}})_{\mathrm{UGN}} = (\tau_{\mathrm{SUPG}})_{\mathrm{UGN}} \left\|\mathbf{u}^h\right\|^2 . \tag{89}$$

We further commented in [12] that the "element length"s h_{UGN} (given by Eq. (55)) and h_{RGN} (Eq. (71)) can be viewed as the local length scales corresponding to the advection- and diffusion-dominated limits, respectively.

10 Deforming-Spatial-Domain/Stabilized Space–Time (DSD/SST) Formulation

In the DSD/SST method [3], the finite element formulation of the governing equations is written over a sequence of N space–time slabs Q_n, where Q_n is the slice of the space–time domain between the time levels t_n and t_{n+1}. At each time step, the integrations involved in the finite element formulation are performed over Q_n. The space–time finite element interpolation functions are continuous within a space–time slab, but discontinuous from one space–time slab to another. The notation $(\cdot)_n^-$ and $(\cdot)_n^+$ denotes the function values at t_n as approached from below and above. Each Q_n is decomposed into elements Q_n^e, where $e = 1, 2, \ldots, (n_{el})_n$. The subscript n used with n_{el} is for the general case in which the number of space–time elements may change from one space–time slab to another. The Dirichlet- and Neumann-type boundary conditions are enforced over $(P_n)_{\mathrm{g}}$ and $(P_n)_{\mathrm{h}}$, the complementary subsets of the lateral boundary of the space–time slab. The finite element trial function spaces $(\mathcal{S}_{\mathbf{u}}^h)_n$ for velocity and $(\mathcal{S}_p^h)_n$ for pressure, and the test function spaces $(\mathcal{V}_{\mathbf{u}}^h)_n$ and $(\mathcal{V}_p^h)_n = (\mathcal{S}_p^h)_n$ are defined by using, over Q_n, first-order polynomials in both space and time. The DSD/SST formulation [3, 12] is written as follows: given $(\mathbf{u}^h)_n^-$, find $\mathbf{u}^h \in (\mathcal{S}_{\mathbf{u}}^h)_n$ and $p^h \in (\mathcal{S}_p^h)_n$ such that $\forall \mathbf{w}^h \in (\mathcal{V}_{\mathbf{u}}^h)_n$ and $q^h \in (\mathcal{V}_p^h)_n$:

$$\begin{aligned}
& \int_{Q_n} \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h - \mathbf{f}^h \right) dQ + \int_{Q_n} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h) dQ \\
- & \int_{(P_n)_{\mathrm{h}}} \mathbf{w}^h \cdot \mathbf{h}^h dP + \int_{Q_n} q^h \boldsymbol{\nabla} \cdot \mathbf{u}^h dQ + \int_{\Omega_n} (\mathbf{w}^h)_n^+ \cdot \rho \left((\mathbf{u}^h)_n^+ - (\mathbf{u}^h)_n^- \right) d\Omega \\
+ & \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho} \left[\tau_{\mathrm{SUPG}} \rho \left(\frac{\partial \mathbf{w}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{w}^h \right) + \tau_{\mathrm{PSPG}} \nabla q^h \right] \cdot \left[\text{Ł}(p^h, \mathbf{u}^h) - \rho \mathbf{f}^h \right] dQ \\
+ & \sum_{e=1}^{n_{el}} \int_{Q_n^e} \nu_{\mathrm{LSIC}} \boldsymbol{\nabla} \cdot \mathbf{w}^h \rho \boldsymbol{\nabla} \cdot \mathbf{u}^h dQ = 0.
\end{aligned} \tag{90}$$

This formulation is applied to all space–time slabs $Q_0, Q_1, Q_2, \ldots, Q_{N-1}$, starting with $(\mathbf{u}^h)_0^- = \mathbf{u}_0$. For an earlier, detailed reference on the DSD/SST formulation see [3].

11 Element-Matrix-Based Parameters for the DSD/SST Formulation

For extensions of the τ calculations based on matrix norms to the DSD/SST formulation, in [12] we defined the space–time augmented versions of the element-level matrices and vectors given by Eqs. (28), (32), and (36) as follows:

$$\mathbf{c}_{\mathrm{A}} : \qquad \int_{Q_n^e} \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h \right) dQ \qquad : (\mathbf{c}_{\mathrm{A}})_{\mathrm{V}}, \tag{91}$$

$$\tilde{\mathbf{k}}_{\mathrm{A}} : \quad \int_{Q_n^e} \left(\frac{\partial \mathbf{w}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h \right) \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h \right) dQ \quad : (\tilde{\mathbf{k}}_{\mathrm{A}})_{\mathrm{V}}, \tag{92}$$

$$\gamma_{\mathrm{A}} : \quad \int_{Q_n^e} \boldsymbol{\nabla} q^h \cdot \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \mathbf{u}^h \right) dQ \quad : (\gamma_{\mathrm{A}})_{\mathrm{V}}. \tag{93}$$

The components of element-matrix-based τ_{SUPG} were defined in [12] as follows:

$$\tau_{\mathrm{S12}} = \frac{\|\mathbf{c}_{\mathrm{A}}\|}{\|\tilde{\mathbf{k}}_{\mathrm{A}}\|}, \tag{94}$$

$$\tau_{\mathrm{S3}} = \tau_{\mathrm{S12}} \frac{\|\mathbf{c}_{\mathrm{A}}\|}{\nu \, \|\tilde{\mathbf{k}}_{\mathrm{r}}\|}, \tag{95}$$

where $\tilde{\mathbf{k}}_{\mathrm{r}}$ is the space–time version (i.e. integrated over the space–time element domain Q_n^e) of the element-level matrix given by Eq. (67). To construct τ_{SUPG} from its components we proposed in [12] the form

$$\tau_{\mathrm{SUPG}} = \left(\frac{1}{\tau_{\mathrm{S12}}^r} + \frac{1}{\tau_{\mathrm{S3}}^r} \right)^{-\frac{1}{r}}. \tag{96}$$

The components of the element-vector-based τ_{SUPG} were defined in [12] as

$$\tau_{\mathrm{SV12}} = \frac{\|(\mathbf{c}_{\mathrm{A}})_{\mathrm{V}}\|}{\|(\tilde{\mathbf{k}}_{\mathrm{A}})_{\mathrm{V}}\|}, \tag{97}$$

$$\tau_{\mathrm{SV3}} = \tau_{\mathrm{SV12}} \frac{\|\mathbf{c}_{\mathrm{A}}\|}{\nu \, \|\tilde{\mathbf{k}}_{\mathrm{r}}\|}. \tag{98}$$

From these two components,

$$(\tau_{\mathrm{SUPG}})_{\mathrm{V}} = \left(\frac{1}{\tau_{\mathrm{SV12}}^r} + \frac{1}{\tau_{\mathrm{SV3}}^r} \right)^{-\frac{1}{r}}. \tag{99}$$

The components of element-matrix-based τ_{PSPG} were defined in [12] as follows:

$$\tau_{\mathrm{P12}} = \frac{\|\mathbf{g}^{\mathrm{T}}\|}{\|\gamma_{\mathrm{A}}\|}, \tag{100}$$

$$\tau_{\mathrm{P3}} = \tau_{\mathrm{P12}} \frac{\|\mathbf{c}_{\mathrm{A}}\|}{\nu \, \|\tilde{\mathbf{k}}_{\mathrm{r}}\|}, \tag{101}$$

where $\mathbf{g}^{\mathrm{T}}$ is the space–time version of the element-level matrix given by Eq. (31). To construct τ_{PSPG} from its components, we proposed in [12] the form

$$\tau_{\mathrm{PSPG}} = \left(\frac{1}{\tau_{\mathrm{P12}}^r} + \frac{1}{\tau_{\mathrm{P3}}^r} \right)^{-\frac{1}{r}}. \tag{102}$$

The components of the element-vector-based τ_{PSPG} were defined in [12] as follows:

$$\tau_{\mathrm{PV12}} = \frac{\|\mathbf{g}_{\mathrm{V}}^{\mathrm{T}}\|}{\|(\gamma_{\mathrm{A}})_{\mathrm{V}}\|}, \tag{103}$$

$$\tau_{\mathrm{PV3}} = \tau_{\mathrm{PV12}} \frac{\|\mathbf{c}_{\mathrm{A}}\|}{\nu \, \|\tilde{\mathbf{k}}_{\mathrm{r}}\|}. \tag{104}$$

From these components,

$$(\tau_{\text{PSPG}})_{\text{V}} = \left(\frac{1}{\tau_{\text{PV12}}^r} + \frac{1}{\tau_{\text{PV3}}^r} \right)^{-\frac{1}{r}} . \tag{105}$$

The element-matrix-based ν_{LSIC} was defined in [12] as

$$\nu_{\text{LSIC}} = \frac{\|\mathbf{c}_{\text{A}}\|}{\|\mathbf{e}\|}, \tag{106}$$

where $\mathbf{e}$ is the space–time version of the element-level matrix given by Eq. (38).

The element-vector-based ν_{LSIC} was defined in [12] as

$$(\nu_{\text{LSIC}})_{\text{V}} = \nu_{\text{LSIC}}. \tag{107}$$

12 UGN/RGN-Based Parameters for the DSD/SST Formulation

The space–time versions of τ_{SUGN1}, τ_{SUGN2}, τ_{SUGN3}, $(\tau_{\text{SUPG}})_{\text{UGN}}$, $(\tau_{\text{PSPG}})_{\text{UGN}}$, and $(\nu_{\text{LSIC}})_{\text{UGN}}$, given respectively by Eqs. (56), (57), (88), (59), (60), and (89), were defined in [12] as follows:

$$\tau_{\text{SUGN12}} = \left(\sum_{a=1}^{n_{en}} \left| \frac{\partial N_a}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} N_a \right| \right)^{-1}, \tag{108}$$

$$\tau_{\text{SUGN3}} = \frac{h_{\text{RGN}}^2}{4\nu}, \tag{109}$$

$$(\tau_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{\tau_{\text{SUGN13}}^r} + \frac{1}{\tau_{\text{SUGN3}}^r} \right)^{-\frac{1}{r}}, \tag{110}$$

$$(\tau_{\text{PSPG}})_{\text{UGN}} = (\tau_{\text{SUPG}})_{\text{UGN}}, \tag{111}$$

$$(\nu_{\text{LSIC}})_{\text{UGN}} = (\tau_{\text{SUPG}})_{\text{UGN}} \left\|\mathbf{u}^h\right\|^2 . \tag{112}$$

Here, n_{en} is the number of nodes for the space–time element, and N_a is the space–time interpolation function associated with node a.

13 Calculation of the Stabilization Parameters for Compressible Flows and Shock-Capturing

The SUPG formulation for compressible flows was first introduced, in the context of conservation variables, in a 1982 NASA Technical Report [14] and a 1983 AIAA paper [2]. Here we will call that formulation "$(SUPG)_{82}$". After that, several SUPG-like methods for compressible flows were developed. Taylor–Galerkin method [15], for example, is very similar, and under certain conditions is identical, to one of the versions of $(SUPG)_{82}$. Another example of the subsequent SUPG-like methods for compressible flows in conservation variables is the streamline-diffusion method described in [16]. Later, following the work in [14, 2], the SUPG formulation for compressible flows was recast in entropy variables and supplemented with a shock-capturing term [17]. It was shown in a 1991 ASME paper [18] that, $(SUPG)_{82}$, when supplemented with a similar shock-capturing term, is very comparable in accuracy to the SUPG formulation that was recast in entropy variables. Later, 2D test computations for inviscid flows reported in [19] showed that the SUPG formulation in conservation and entropy variables yielded indistinguishable results.

Together with $(SUPG)_{82}$, the 1982 NASA Technical Report [14] and 1983 AIAA paper [2] introduced a set of stabilization parameters (τs) to be used in conjunction with that formulation. That set of τs will be called here "τ_{82}". The stabilized formulation introduced in [6] for advection–diffusion–reaction equations included a shock-capturing term and a τ definition that takes into account the interaction between the shock-capturing term and the SUPG term. That τ definition, for example, precludes "compounding" (i.e. augmentation of the SUPG effect by the shock-capturing effect when the advection and shock directions coincide). In the 1991 ASME paper [18], the τ used with $(SUPG)_{82}$ is a slightly modified version of τ_{82}, and a shock-capturing parameter, which we will call here "δ_{91}", is embedded in the shock-capturing term used. Subsequent minor modifications of τ_{82} took into account the interaction between the shock-capturing and the $(SUPG)_{82}$ terms in a fashion similar to how it was done in [6] for advection–diffusion–reaction equations. All these slightly modified versions of τ_{82} have always been used with the same δ_{91}, and we categorize them all under the label "$\tau_{82\text{-MOD}}$". The element-matrix-based τ definitions introduced in [9] were recently applied in [20] to $(SUPG)_{82}$, supplemented with the shock-capturing term (with δ_{91}) used in [18].

In this section, in the context of the $(SUPG)_{82}$ formulation and based on the ideas discussed in earlier sections, we propose alternative ways of calculating the stabilization parameters and defining the shock-capturing terms. For this we first define the conservation variables vector as $\mathbf{U} = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho e)$ (e is the total energy per unit volume), associate to it a test vector-function $\mathbf{W}$, define the acoustic speed as c, and define the unit vector $\mathbf{j}$ as

$$\mathbf{j} = \frac{\boldsymbol{\nabla}\rho^h}{\| \boldsymbol{\nabla}\rho^h \|} \; . \tag{113}$$

As the first alternative in computing τ_{SUGN1} for each component of the test vector-function $\mathbf{W}$, we propose to define ${\tau^\rho}_{\text{SUGN1}}$, ${\tau^u}_{\text{SUGN1}}$ and ${\tau^e}_{\text{SUGN1}}$ (associated with ρ, $\rho\mathbf{u}$ and ρe, respectively) by using the expression given by Eq. (63):

$${\tau^\rho}_{\text{SUGN1}} = {\tau^u}_{\text{SUGN1}} = {\tau^e}_{\text{SUGN1}} = \left(\sum_{a=1}^{n_{en}} |\mathbf{u}^h \cdot \boldsymbol{\nabla} N_a| \right)^{-1} \; . \tag{114}$$

As the second alternative, we propose to use the following definition:

$${\tau^\rho}_{\text{SUGN1}} = {\tau^u}_{\text{SUGN1}} = {\tau^e}_{\text{SUGN1}} = \left(\sum_{a=1}^{n_{en}} \left(c\, |\mathbf{j} \cdot \boldsymbol{\nabla} N_a| + |\mathbf{u}^h \cdot \boldsymbol{\nabla} N_a| \right) \right)^{-1} \; . \tag{115}$$

In computing τ_{SUGN2}, we propose to use the expression given by Eq. (57):

$${\tau^\rho}_{\text{SUGN2}} = {\tau^u}_{\text{SUGN2}} = {\tau^e}_{\text{SUGN2}} = \frac{\Delta t}{2} \; . \tag{116}$$

In computing τ_{SUGN3}, we propose to define ${\tau^u}_{\text{SUGN3}}$ by using the expression given by Eq. (88):

$${\tau^u}_{\text{SUGN3}} = \frac{h^2_{\text{RGN}}}{4\nu} \; . \tag{117}$$

We propose to define ${\tau^e}_{\text{SUGN3}}$ as

$$ {\tau^e}_{\text{SUGN3}} = \frac{({h^e}_{\text{RGN}})^2}{4\nu^e} \ , \tag{118} $$

where ν^e is the "kinematic viscosity" for the energy equation,

$$ {h^e}_{\text{RGN}} = 2 \left(\sum_{a=1}^{n_{en}} |\mathbf{r}^e \cdot \boldsymbol{\nabla} N_a| \right)^{-1} , \tag{119} $$

$$ \mathbf{r}^e = \frac{\boldsymbol{\nabla}\theta^h}{\| \boldsymbol{\nabla}\theta^h \|} \ , \tag{120} $$

and θ is the temperature. We define $(\tau^\rho_{\text{SUPG}})_{\text{UGN}}$, $(\tau^u_{\text{SUPG}})_{\text{UGN}}$ and $(\tau^e_{\text{SUPG}})_{\text{UGN}}$ by using the "*r-switch*" given in Section 4 :

$$ (\tau^\rho_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{({\tau^\rho}_{\text{SUGN1}})^r} + \frac{1}{({\tau^\rho}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}} , \tag{121} $$

$$ (\tau^u_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{({\tau^u}_{\text{SUGN1}})^r} + \frac{1}{({\tau^u}_{\text{SUGN2}})^r} + \frac{1}{({\tau^u}_{\text{SUGN3}})^r} \right)^{-\frac{1}{r}} , \tag{122} $$

$$ (\tau^e_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{({\tau^e}_{\text{SUGN1}})^r} + \frac{1}{({\tau^e}_{\text{SUGN2}})^r} + \frac{1}{({\tau^e}_{\text{SUGN3}})^r} \right)^{-\frac{1}{r}} . \tag{123} $$

In defining the shock-capturing term, we first define the "shock-capturing viscosity" ν_{SHOC}:

$$ \nu_{\text{SHOC}} = \tau_{\text{SHOC}} (u_{\text{int}})^2, \tag{124} $$

where

$$ \tau_{\text{SHOC}} = \frac{h_{\text{SHOC}}}{2u_{\text{cha}}} \left(\frac{\| \boldsymbol{\nabla}\rho^h \| \, h_{\text{SHOC}}}{\rho_{\text{ref}}} \right)^\beta , \tag{125} $$

$$ h_{\text{SHOC}} = h_{\text{JGN}} \ , \tag{126} $$

$$ h_{\text{JGN}} = 2 \left(\sum_{a=1}^{n_{en}} |\mathbf{j} \cdot \boldsymbol{\nabla} N_a| \right)^{-1} . \tag{127} $$

Here ρ_{ref} is a reference density (such as ρ^h at the inflow, or the difference between the estimated maximum and minimum values of ρ^h), u_{cha} is a characteristic velocity (such as u_{ref} or $\|\mathbf{u}^h\|$ or acoustic speed c), and u_{int} is an intrinsic velocity (such as u_{cha} or $\|\mathbf{u}^h\|$ or acoustic speed c). We propose to set $u_{\text{int}} = u_{\text{cha}} = u_{\text{ref}}$. The parameter β influences the smoothness of the shock-front. We set $\beta = 1.0$ for smoother shocks and $\beta = 2.0$ for sharper shocks (in return for tolerating possible overshoots and undershoots). Then the shock-capturing term is defined as

$$ S_{\text{SHOC}} = \sum_{e=1}^{n_{el}} \int_{\Omega^e} \boldsymbol{\nabla}\mathbf{W}^h : \left(\boldsymbol{\kappa}_{\text{SHOC}} \cdot \boldsymbol{\nabla}\mathbf{U}^h \right) d\Omega \ , \tag{128} $$

where $\boldsymbol{\kappa}_{\text{SHOC}}$ is defined as

$$\boldsymbol{\kappa}_{\text{SHOC}} = \nu_{\text{SHOC}}\ \mathbf{I}\ . \tag{129}$$

As a possible alternative, we propose

$$\boldsymbol{\kappa}_{\text{SHOC}} = \nu_{\text{SHOC}}\ \mathbf{jj}\ . \tag{130}$$

To preclude compounding, we propose to modify ν_{SHOC} as follows:

$$\nu_{\text{SHOC}}\ \leftarrow\ \nu_{\text{SHOC}} - switch\left(\tau_{\text{SUPG}}\ (\mathbf{j}\cdot\mathbf{u})^2,\ \tau_{\text{SUPG}}\ (|\mathbf{j}\cdot\mathbf{u}| - c)^2,\ \nu_{\text{SHOC}}\right)\ , \tag{131}$$

where the "*switch*" function is defined as the "*min*" function or as the "*r-switch*" given in Section 4. For viscous flows, the above modification would be made separately with each of $\tau^{\rho}_{\text{SUPG}}$, τ^{u}_{SUPG} and τ^{e}_{SUPG}, and this would result in ν_{SHOC} becoming a diagonal matrix.

Remark 8 *Remark 6 applies also to the calculation of* τ^{ρ}_{SUPG}, τ^{u}_{SUPG} *and* τ^{e}_{SUPG}, *and* ν_{SHOC}.

14 Examples of Test Computations

In this section, we present examples of test computations carried out to evaluate the performances of the stabilization parameters. For additional examples see [9, 10, 21].

2D incompressible flow past a cylinder – comparison of stabilization parameters. In this test computation, we compare the performances of the element-matrix-based and UGN-based stabilization parameters. The test problem we use, 2D incompressible flow past a cylinder at Re = 100, is a well-studied problem, with an easily identifiable Karman vortex shedding. We are using a quadrilateral mesh with increased refinement near the cylinder. In this computation $U_\infty \Delta t/R = 0.1$, where U_∞ is the free-stream velocity and R is the cylinder radius. Figure 1 shows the values of the stabilization parameters τ_{SUPG} and τ_{PSPG} along the vertical line passing through the cylinder center, starting from the upper cylinder surface. For more details on this test computation, see [9, 10].

2D incompressible flow past a cylinder – elements with high aspect ratios. In this test computation at Re = 100, for meshes containing elements with high aspect ratios, we evaluate the performance of the SUPG/PSPG formulation with UGN/RGN-based stabilization parameters. We are using a triangular mesh with increased refinement near the cylinder. Figure 2 shows the velocity vectors near the cylinder and in the boundary layer. Although the aspect ratio of the elements adjacent to the cylinder surface is 100, the SUPG/PSPG formulation with the UGN/RGN-based stabilization parameters performs very well. For more details on this test computation, see [21].

15 Concluding Remarks

We reviewed the calculation ways we developed for the stabilization parameters used in the streamline-upwind/Petrov-Galerkin (SUPG) and pressure-stabilizing/Petrov-Galerkin (PSPG) methods. These parameters were designed for the semi-discrete

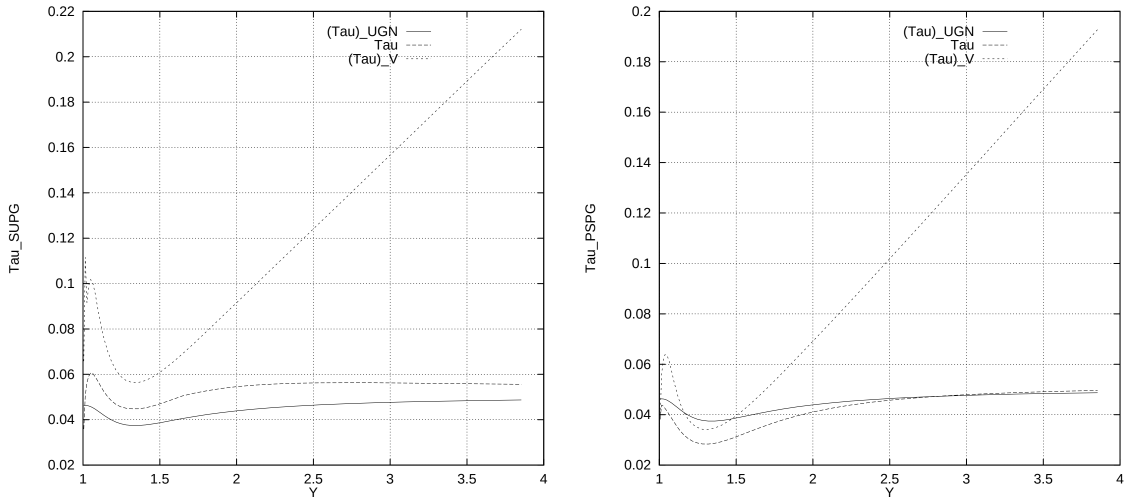

Figure 1: 2D incompressible flow past a cylinder – comparison of stabilization parameters. Re = 100. Stabilization parameters: τ_{SUPG} (left) and τ_{PSPG} (right). Here $(\tau)_{\mathrm{UGN}}$ represents the group of stabilization parameters $(\tau_{\mathrm{SUPG}})_{\mathrm{UGN}}$ (given by Eq. (59)) and $(\tau_{\mathrm{PSPG}})_{\mathrm{UGN}}$ (Eq. (60)); τ represents τ_{SUPG} (Eq. (20)) and τ_{PSPG} (Eq. (42)); and $(\tau)_{\mathrm{V}}$ represents $(\tau_{\mathrm{SUPG}})_{\mathrm{V}}$ (Eq. (24)) and $(\tau_{\mathrm{PSPG}})_{\mathrm{V}}$ (Eq. (46)).

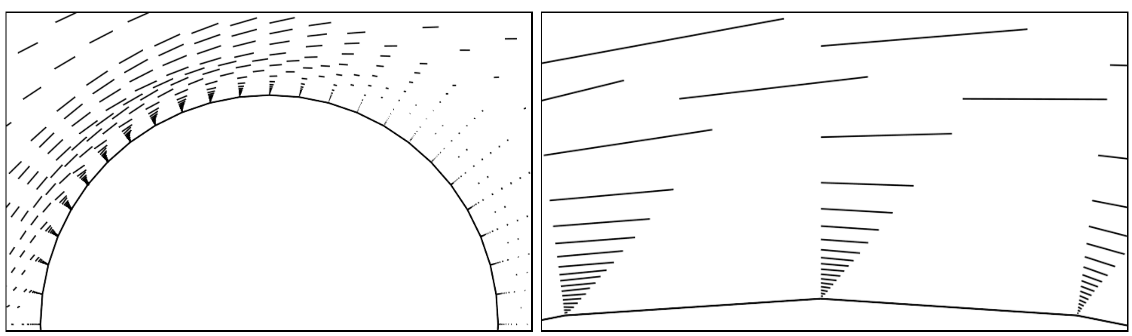

Figure 2: 2D incompressible flow past a cylinder – elements with high aspect ratios. Re = 100. Computed with SUPG/PSPG formulation with UGN/RGN-based stabilization parameters. Velocity vectors near the cylinder (left) and in the boundary layer (right).

and space–time formulations and for the advection–diffusion and Navier–Stokes equations. Some of the parameters are calculated based on the element-level matrices and vectors and are expressed in terms of the ratios of the norms of the matrices or vectors involved in the definitions. The local length scales, advection field, and the element-level Reynolds number are represented in these definitions, because they are contained in the element-level matrices and vectors. Based on these definitions, a τ can be calculated for each element, or for each element node or degree of freedom or element equation. Furthermore, based on these definitions, a τ can be calculated for each element integration point. Some other stabilization parameters are calculated by directly taking into account the flow velocity, viscosity, and the local length scales for the advection- and diffusion-dominated limits. With examples of the test problems we solved for the Navier–Stokes equations, we showed that the stabilization parameters described perform well, even for elements with high aspect ratios. The SUPG and PSPG methods are used extensively in many finite element formulations, including the interface-tracking and interface-capturing techniques we developed for computation of flow problems with moving boundaries and interfaces. Therefore the stabilization parameters described here will strengthen our computational techniques for a wide range of fluid mechanics applications.

ACKNOWLEDGMENT
This work was supported by the Army Natick Soldier Center and NASA JSC.

REFERENCES

[1] T.J.R. Hughes and A.N. Brooks, "A multi-dimensional upwind scheme with no crosswind diffusion", in T.J.R. Hughes, editor, *Finite Element Methods for Convection Dominated Flows*, AMD-Vol.34, 19–35, ASME, New York, 1979.

[2] T.E. Tezduyar and T.J.R. Hughes, "Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations", in *Proceedings of AIAA 21st Aerospace Sciences Meeting*, AIAA Paper 83-0125, Reno, Nevada, (1983).

[3] T.E. Tezduyar, "Stabilized finite element formulations for incompressible flow computations", *Advances in Applied Mechanics*, **28** (1992) 1–44.

[4] T.E. Tezduyar, "Finite element methods for flow problems with moving boundaries and interfaces", *Archives of Computational Methods in Engineering*, **8** (2001) 83–130.

[5] T. Tezduyar, "Interface-tracking and interface-capturing techniques for computation of moving boundaries and interfaces", in *Proceedings of the Fifth World Congress on Computational Mechanics*, On-line publication: http://wccm.tuwien.ac.at/, Paper-ID: 81513, Vienna, Austria, (2002).

[6] T.E. Tezduyar and Y.J. Park, "Discontinuity capturing finite element formulations for nonlinear convection-diffusion-reaction equations", *Computer Methods in Applied Mechanics and Engineering*, **59** (1986) 307–325.

[7] T.E. Tezduyar and D.K. Ganjoo, "Petrov-Galerkin formulations with weighting functions dependent upon spatial and temporal discretization: Applications to transient convection-diffusion problems", *Computer Methods in Applied Mechanics and Engineering*, **59** (1986) 49–71.

[8] L.P. Franca, S.L. Frey, and T.J.R. Hughes, "Stabilized finite element methods: I. Application to the advective-diffusive model", *Computer Methods in Applied Mechanics and Engineering*, **95** (1992) 253–276.

[9] T.E. Tezduyar and Y. Osawa, "Finite element stabilization parameters computed from element matrices and vectors", *Computer Methods in Applied Mechanics and Engineering*, **190** (2000) 411–430.

[10] T.E. Tezduyar, "Computation of moving boundaries and interfaces and stabilization parameters", *International Journal for Numerical Methods in Fluids*, **43** (2003) 555–575.

[11] T.E. Tezduyar, "Stabilized finite element formulations and interface-tracking and interface-capturing techniques for incompressible flows", in M.M. Hafez, editor, *Numerical Simulations of Incompressible Flows*, World Scientific, New Jersey, (2003) 221–239.

[12] T. Tezduyar, "Stabilization parameters and local length scales in SUPG and PSPG formulations", in *Proceedings of the Fifth World Congress on Computational Mechanics*, On-line publication: http://wccm.tuwien.ac.at/, Paper-ID: 81508, Vienna, Austria, (2002).

[13] T.E. Tezduyar, "Calculation of the stabilization parameters in SUPG and PSPG formulations", in *Proceedings of the First South-American Congress on Computational Mechanics (CD-ROM)*, Santa Fe–Parana, Argentina, (2002).

[14] T.E. Tezduyar and T.J.R. Hughes, "Development of time-accurate finite element techniques for first-order hyperbolic systems with particular emphasis on the compressible Euler equations", NASA Technical Report NASA-CR-204772, NASA, 1982, http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19970023187_1997034954.pdf.

[15] J. Donea, "A Taylor-Galerkin method for convective transport problems", *International Journal for Numerical Methods in Engineering*, **20** (1984) 101–120.

[16] C. Johnson, U. Navert, and J. Pitkäranta, "Finite element methods for linear hyperbolic problems", *Computer Methods in Applied Mechanics and Engineering*, **45** (1984) 285–312.

[17] T.J.R. Hughes, L.P. Franca, and M. Mallet, "A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multi-dimensional advective-diffusive systems", *Computer Methods in Applied Mechanics and Engineering*, **63** (1987) 97–112.

[18] G.J. Le Beau and T.E. Tezduyar, "Finite element computation of compressible flows with the SUPG formulation", in *Advances in Finite Element Analysis in Fluid Dynamics*, FED-Vol.123, ASME, New York, (1991) 21–27.

[19] G.J. Le Beau, S.E. Ray, S.K. Aliabadi, and T.E. Tezduyar, "SUPG finite element computation of compressible flows with the entropy and conservation variables formulations", *Computer Methods in Applied Mechanics and Engineering*, **104** (1993) 397–422.

[20] L. Catabriga, A.L.G.A. Coutinho, and T.E. Tezduyar, "Finite element SUPG parameters computed from local matrices for compressible flows", in *Proceedings of the 9th Brazilian Congress of Engineering and Thermal Sciences*, Caxambu, Brazil, (2002).

[21] T. Tezduyar and S. Sathe, "Stabilization parameters in SUPG and PSPG formulations", *Journal of Computational and Applied Mechanics*, **4** (2003) 71–88.