ELSEVIER

Comput. Methods Appl. Mech. Engrg. 174 (1999) 153–170

Computer methods in applied mechanics and engineering

A better consistency for low-order stabilized finite element methods

Kenneth E. Jansen[a,*,1], S. Scott Collis[b,2], Christian Whiting[a,1], Farzin Shakib[c]

[a]*Rensselaer Polytechnic Institute, Troy, NY, USA*
[b]*Rice University, Houston, TX, USA*
[c]*Acusim Software, Inc, Los Altos, CA, USA*

Received 31 October 1997

Abstract

The standard implementation of stabilized finite element methods with a piece-wise function space of order lower than the highest derivative present in the partial differential equation often suffers from a weak consistency that can lead to reduced accuracy. The popularity of these low-order elements motivates the development of a new stabilization operator which globally reconstructs the derivatives not present in the local element function space. This new method is seen to engender a stronger consistency leading to better convergence and improved accuracy. Applications to the Navier–Stokes equations are given which illustrate the improvement at a negligible additional cost.

1. Introduction

Over the last two decades, stabilized finite element methods have grown in popularity, especially in application to fluid dynamics. Starting with Brooks and Hughes [4] with the SUPG method, through Hughes' et al. [12] work on the Galerkin/least squares (GLS) method and up to recent work on multiscale methods of Hughes [11] and related work on residual free bubbles by Ruśso [16] and Brezzi et al. [3], a number of stabilized formulations have been proposed. Each formulation is cast in weighted residual form through the addition of an integral over the interior of each element. This weighted residual form is a necessary condition to attain the traditional notion of finite element consistency. The consistency emanates from the requirement that, should the exact solution replace the discrete solution, the weak form should be satisfied. This consistency is critical to the proofs of higher-order accuracy that these methods have been shown to attain.

As mentioned above, the consistency of these weak forms is guaranteed by construction. However, the consistency of low-order finite element implementations based on these formulations is less clear. Here, we define low-order to mean when the order of the piecewise polynomial is less than the highest derivative present in the problem (a more precise definition will be given later). For example, the Navier–Stokes equations have second derivatives in the diffusive terms and therefore piecewise-linear-in-space elements are termed a low-order method. Likewise, if we employ a piecewise-constant-in-time element, our temporal piecewise polynomial is less than the first-order derivative present in the unsteady Navier–Stokes equations.

The concern in these cases is that the stabilization term that is added to the Galerkin formulation, though based upon the complete residual of the weak form (when the function space was all of H^1), has no ability to approximate the residual in the chosen finite element space. Indeed, with this choice of basis functions, it is

*Corrresponding author.
[1]Supported by a grant from the AFOSR.
[2]Supported by a grant from NASA LaRC.

0045-7825/99/$ – see front matter
PII: S0045-7825(98)00284-9

possible to refine the mesh to infinitesimal scales and yet this residual would not be zero, since the local function space has no approximation power for derivatives higher than the polynomial order. It is important to point out here that the situation is not as grim as it might first seem since all of the stabilization methods described above employ a stabilization coefficient matrix, τ, (or weight factor of some form) which itself vanishes as we refine the mesh. In fact the analysis of the above methods foreshadows this problem by requiring the assumption that $\tau = O(h^m)$, $(m \geq 1)$ where $h = \min(\Delta x, c\,\Delta t)$. The message from the analysis is clear, convergence depends on a uniform refinement in space and time, driving h and subsequently τ to zero, thereby recovering the consistency of the method.

We characterize this as poor consistency since it depends so heavily on the stabilization coefficient. It is not very satisfying that the method becomes consistent due to the elimination of the stabilization term. Higher-order methods do not have this problem since the local residual itself goes to zero achieving consistency without the strict requirements on τ. In this paper we seek to develop improved, low-order stabilized methods by globally reconstructing the terms not seen by the local element residual. These methods recover the ability to approximate the residual in the stabilization term yielding a better consistency. The methods will be shown to also yield more accurate solutions at a negligible additional cost. The methods are also proposed to be a more cost-effective approach to higher-order implementations of the SUPG method since they eliminate the need to calculate the second derivative of the mapping.

2. Navier–Stokes equations

Consider the compressible Navier–Stokes equations (complete with continuity and total energy equation) after assuming Fourier heat conduction, Newtonian viscosity and Stokes bulk viscosity

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

where

$$\boldsymbol{U} = \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{Bmatrix} = \rho \begin{Bmatrix} 1 \\ u_1 \\ u_2 \\ u_3 \\ e_{\text{tot}} \end{Bmatrix}, \qquad \boldsymbol{F}^{\text{adv}}_i = u_i \boldsymbol{U} + p \begin{Bmatrix} 0 \\ \delta_{1i} \\ \delta_{2i} \\ \delta_{3i} \\ u_i \end{Bmatrix}, \qquad \boldsymbol{F}^{\text{diff}}_i = \begin{Bmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ \tau_{3i} \\ \tau_{ij}u_j - q_i \end{Bmatrix} \tag{2}$$

and

$$\tau_{ij} = \mu\left(S_{ij}(\boldsymbol{u}) - \frac{1}{3}S_{kk}(\boldsymbol{u})\delta_{ij}\right), \qquad q_i = -\kappa T_{,i}, \qquad \kappa = c_{\text{p}}\frac{\mu}{\text{Pr}} \tag{3}$$

$$S_{ij}(\boldsymbol{u}) = u_{i,j} + u_{j,i}, \qquad e_{\text{tot}} = e + \frac{u_i u_i}{2}, \qquad e = c_{\text{v}}T. \tag{4}$$

Here, and after, indicial notation is used where repeated indices are summed and inferior commas denote differentiation with respect to the variables that follow the comma.

The variables are: the velocity u_i, the pressure p, the density ρ, the temperature T and the total energy e_{tot}. The constitutive laws relate the stress, τ_{ij}, to the strain-rate, through a molecular viscosity, μ. Similarly, the heat flux, q_i, is proportional to the gradient of temperature with the proportionality constant given by the molecular conductivity, κ. The fluid is modelled as an ideal gas, $p = \rho RT$, where both the specific heat at constant pressure, c_{p}, and the specific heat at constant volume, c_{v}, are constant. Since all the calculations shown in this paper are at low Mach number, where temperature variations are small, we have also assumed a constant molecular viscosity and constant conductivity through a constant Prandtl number. Finally, $\mathscr{S}$ is a body force (or source) term which we leave general at this time.

For the specification of the methods that follow, it is helpful to define the quasi-linear operator related to (1) as

$$\mathscr{L} \equiv \boldsymbol{I}\frac{\partial}{\partial t} + \boldsymbol{A}_i \frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\boldsymbol{K}_{ij}\frac{\partial}{\partial x_j}\right) \tag{5}$$

from which $\mathscr{L}$ can be naturally decomposed into time, advective and diffusive portions

$$\mathscr{L} = \mathscr{L}_{\mathrm{t}} + \mathscr{L}_{\mathrm{adv}} + \mathscr{L}_{\mathrm{diff}}. \tag{6}$$

Here $\boldsymbol{A}_i = \boldsymbol{F}_{i,U}^{\mathrm{adv}}$ is the ith Euler Jacobian matrix, $\boldsymbol{K}_{ij}$ is the diffusivity matrix, defined such that $\boldsymbol{K}_{ij}\boldsymbol{U}_{,j} = \boldsymbol{F}_i^{\mathrm{diff}}$ and $\boldsymbol{I}$ is a (5×5) identity matrix. For a complete description of $\boldsymbol{A}_i$ and $\boldsymbol{K}_{ij}$, the reader is referred to Hauke and Hughes [10]. Using this, we can write (1) as simply $\mathscr{L}\boldsymbol{U} = \mathscr{S}$.

3. Weak forms

To proceed with the finite element discretization of the Navier–Stokes equations (1), we must make precise the approximation spaces with which we are dealing. First, let $\bar{\Omega} \subset \boldsymbol{R}^{n_{sd}}$ represent the closure of the physical spatial domain (i.e. $\Omega \cup \Gamma$ where Γ is the boundary) and $Q_n = \Omega \times I_n$ the nth space–time slab with boundary $\mathscr{P}_n = \Gamma \times I_n$, in n_{sd} space dimensions, only $n_{\mathrm{sd}} = 3$ is considered here. In addition, $H^1(\Omega)$ represents the usual Sobolev space of functions with square-integrable values and derivatives on Ω.

Next, Ω or Q_n is discretized into n_{el} finite elements, Ω^e or Q_n^e, depending on the type of discretization, respectively semi-discrete or space–time. With this, we can define the trial solution space for the semi-discrete formulations as

$$\mathscr{V}_h = \{\boldsymbol{v} | \boldsymbol{v}(\cdot, t) \in H^1(\Omega)^m, t \in [0, T], \boldsymbol{v}|_{x\in\Omega^e} \in P_k(\Omega^e)^m, \boldsymbol{v}(\cdot, t) = \boldsymbol{g} \text{ on } \Gamma_g\}, \tag{7}$$

and the weighting function space

$$\mathscr{W}_h = \{\boldsymbol{w} | \boldsymbol{w}(\cdot, t) \in H^1(\Omega)^m, t \in [0, T], \boldsymbol{w}|_{x\in\Omega^e} \in P_k(\Omega^e)^m, \boldsymbol{w}(\cdot, t) = \boldsymbol{0} \text{ on } \Gamma_g\}, \tag{8}$$

where $P_k(\Omega^e)$, is the space of all polynomials defined on Ω^e complete to order $k \geqslant 1$, m is the number of degrees of freedom ($m = 5$), and Γ_g is the portion of the boundary upon which the solution is prescribed. For the constant-in-time space–time formulations, the trial solution space is given by

$$\mathscr{V}_h^n = \{\boldsymbol{v} | \boldsymbol{v} \in H^1(Q_n^e)^m, \boldsymbol{v}|_{x\in Q_n^e} \in P_k(Q_n^e)^m, \boldsymbol{v} = \boldsymbol{g}(t) \text{ on } \mathscr{P}_{ng}\}, \tag{9}$$

with corresponding weight space

$$\mathscr{W}_h^n = \{\boldsymbol{w} | \boldsymbol{w} \in H^1(Q_n^e)^m, \boldsymbol{w}|_{x\in Q_n^e} \in P_k(Q_n^e)^m, \boldsymbol{w} = \boldsymbol{0} \text{ on } \mathscr{P}_{ng}\}, \tag{10}$$

where now $P_k(Q_n^e)$ is the space of polynomials defined on the nth space–time slab and $\mathscr{P}_{ng}$ is the portion of the space–time slab boundary upon which the solution is prescribed. In principle, the discretization Q_n may change from step to step thus the subscript n. Continuity is not enforced between successive space–time slabs.

Of particular interest to the present paper is the local polynomial approximation space, $P_k(\Omega^e)$, which allows for a precise definition of the term 'low-order method'. If the order of complete polynomial, k, is less than the order of the strong form of the differential equation (or, more generally, less than the highest order derivatives appearing in the associated weak form), then the method will be defined as low-order. In this case, the local approximation space will have no ability to represent the terms which are of higher order than k. This is precisely what happens in the case of the Navier–Stokes equations (1) approximated with piecewise-linear in space, $k = 1$, or constant-in-time elements. Clearly, these equations contain second derivatives in space and first derivatives in time, making these choices of approximation space low-order.

3.1. Semi-discrete GLS

To derive the weak form of (1), the entire equation is dotted from the left by a vector of smooth weight functions and integrated over the spatial domain. Integration by parts is then performed to move some of the derivatives onto the weight functions (reducing the continuity requirements when we approximate the smooth

weight functions by the finite collection of piecewise polynomials described in (8)). This process leads to the integral equation (often referred to as the variational equation): find $\boldsymbol{U} \in \mathcal{V}_h$[3] such that $\forall\ \boldsymbol{W} \in \mathcal{W}_h$,

$$0 = \int_{\Omega} (\boldsymbol{W}\cdot\boldsymbol{U}_{,t} - \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{diff}} + \boldsymbol{W}\cdot\mathcal{S})\,\mathrm{d}\Omega - \int_{\Gamma} \boldsymbol{W}\cdot(-\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{F}_i^{\text{diff}})n_i\,\mathrm{d}\Gamma + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \mathcal{L}^{\mathrm{T}}\boldsymbol{W}\cdot\boldsymbol{\tau}(\mathcal{L}\boldsymbol{U} - \mathcal{S})\,\mathrm{d}\Omega, \tag{11}$$

Here, Ω is the spatial domain of the problem with boundary Γ, discretized into n_{el} elements. The first line of (11) contains the Galerkin approximation (interior and boundary) and the second line contains the least-squares stabilization which is composed of the sum of integrals over element interiors. The integrand is the variation of the square of the residual and therefore contains the full residual of the partial differential equation (1), multiplied by a stabilization matrix $\boldsymbol{\tau}$, then dotted from the left by the partial differential operator acting on the weight function. Methods of this type are known as weighted residual methods since their Euler–Lagrange conditions are easily shown to be the product of some weight function multiplying a residual. The success of the method emanates from having the residual of the partial differential equation controlled (driven to a small, ultimately zero, value) on each element interior. The stabilization matrix $\boldsymbol{\tau}$ is an important ingredient in these methods and is well documented in Shakib et al. [17].

This formulation has been proven to be stable and higher-order accurate (the L_2 norm of the error converges at a rate proportional to h^{k+1}) for a linear advective-diffusive system (the closest model problem to (1)) in [12] and for the linearized incompressible Navier–Stokes equations in [7].

3.2. Semi-discrete SUPG

A predecessor to the GLS method is the SUPG method. The difference is that rather than having $\mathcal{L}$ operating on the weighting space, only its advective portion, $\mathcal{L}_{\text{adv}}$, acts there. It is important to note that the full operator still operates on the solution space preserving the weighted residual character of the formulation. The weak form may be stated in this case as: find $\boldsymbol{U} \in \mathcal{V}_h$ such that $\forall\ \boldsymbol{W} \in \mathcal{W}_h$,

$$0 = \int_{\Omega} (\boldsymbol{W}\cdot\boldsymbol{U}_{,t} - \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{diff}} + \boldsymbol{W}\cdot\mathcal{S})\,\mathrm{d}\Omega - \int_{\Gamma} \boldsymbol{W}\cdot(-\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{F}_i^{\text{diff}})n_i\,\mathrm{d}\Gamma + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \mathcal{L}_{\text{adv}}^{T}\boldsymbol{W}\cdot\boldsymbol{\tau}(\mathcal{L}\boldsymbol{U} - \mathcal{S})\,\mathrm{d}\Omega \tag{12}$$

3.3. Space–time SUPG

Both of the previous variational formulations, (11) and (12), are semi-discrete methods which require the further step of choosing a time integrator to advance the resulting system of ordinary differential equations in time. In this paper, no emphasis is placed upon semi-discrete time integrators since they have no bearing on our results. However, there exists another variational formulation which does warrant careful study to insure an implementation with improved consistency. This method is known as the discontinuous-in-time Galerkin method. It could be stabilized by GLS or SUPG (we will only show SUPG here for brevity). The weak form may be stated as: find $\boldsymbol{U} \in \mathcal{V}_h^n$, such that $\forall\ \boldsymbol{W} \in \mathcal{W}_h^n$.

$$0 = \int_{Q_n} (-\boldsymbol{W}_{,t}\cdot\boldsymbol{U} - \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{W}_{,i}\cdot\boldsymbol{F}_i^{\text{diff}} - \boldsymbol{W}\cdot\mathcal{S})\,\mathrm{d}Q - \int_{\mathcal{P}_n} \boldsymbol{W}\cdot(-\boldsymbol{F}_i^{\text{adv}} + \boldsymbol{F}_i^{\text{diff}})n_i\,\mathrm{d}\mathcal{P} + \int_{\Omega} (\boldsymbol{W}(t_{n+1}^-)\cdot\boldsymbol{U}(t_{n+1}^-) - \boldsymbol{W}(t_n^+)\cdot\boldsymbol{U}(t_n^-))\,\mathrm{d}\Omega + \sum_{e=1}^{(n_{\text{el}})_n} \int_{Q_n^e} \mathcal{L}_{\text{adv}}^{T}\boldsymbol{W}\cdot\boldsymbol{\tau}(\mathcal{L}\boldsymbol{U} - \mathcal{S})\,\mathrm{d}\Omega \tag{13}$$

[3] In practice, one may choose any linear independent variable set for the solution variables (i.e. $\boldsymbol{Y} \in \mathcal{V}_h$) as discussed in Hauke and Hughes [9]. This choice has no bearing on the principles of this paper and so only the most notationally simple choice is discussed herein.

where the integrals are now over the space–time slabs, Q_n, with space–time boundaries, $\mathcal{P}_n$. The stabilization integral is now the sum of integrals over the space–time element interiors, the eth of which is denoted by (Q_n^e). This, like the previous two weak forms, is a weighted residual method, the only difference being that now the residual is integrated over both space and time.

4. Incomplete residual implementations

All three formulations described in the previous section (11,12,13) are weighted residual methods and therefore enjoy the traditional finite element notion of consistency, where, if the discrete solution is replaced by the exact solution the equations are satisfied. However, common low-order implementations of these three approaches can lead to methods for which this notion of consistency is rather hollow. By this we mean that the 'residual' that the method controls has no ability to approximate the actual residual and therefore only becomes consistent as the stabilization parameter matrix $\boldsymbol{\tau}$ vanishes.

The problem occurs when the polynomial order used is lower than the highest derivative present in the system. For example, consider a piecewise linear-in-space implementation of either (11) or (12) with n_{en} basis functions, N_a. Common implementations would neglect contributions made by $\mathscr{L}_{\text{diff}}\boldsymbol{U}$ on the grounds that it is composed of second derivatives which, on the interior of a piecewise linear element, are zero by the usual finite element definition, viz.

$$\phi^h(\boldsymbol{x}_e)=\sum_{a=1}^{n_{\text{en}}} N_a\phi_a \qquad \phi^h_{,x_1}(\boldsymbol{x}_e)=\sum_{a=1}^{n_{\text{en}}} N_{a,x_1}\phi_a \qquad \phi^h_{,x_1x_1}(\boldsymbol{x}_e)=\sum_{a=1}^{n_{\text{en}}} N_{a,x_1x_1}\phi_a \tag{14}$$

where ϕ^h could be any component of $\boldsymbol{U}$, the discrete solution, and ϕ_a are the nodal values of the same. The problem is clear, for linear elements, $N_{a,x_1x_1}=0$.

The consequence of this implementation is that $\mathscr{L}$ gets replaced in (11) or (12) by $\mathscr{L}_{\text{t}}+\mathscr{L}_{\text{adv}}$ but

$$\mathscr{L}_{\text{t}}\boldsymbol{U}+\mathscr{L}_{\text{adv}}\boldsymbol{U}-\mathscr{S}\neq 0. \tag{15}$$

Therefore, the method's residual will only be a good approximation to the real residual when diffusion is small. There exist many interesting problems for which this is not the case. Furthermore, many of these problems are actually represented quite well with linear elements. For example, in Poiseuille flow the viscous term is exactly balanced by the pressure force. Linear elements are capable of obtaining a solution with a nodally exact velocity profile since this profile is only quadratic. This solution is spoiled by the low-order implementations described above. More discussion of these results will be given in the numerical results section 6.1.

A second example of residual incompleteness is observed for the space time formulation (13) when it is implemented with constant-in-time elements. The standard implementation of this formulation would observe that the solution space within an element does not depend on time and therefore $\mathscr{L}$ gets replaced in (13) or (12) by $\mathscr{L}_{\text{adv}}+\mathscr{L}_{\text{diff}}$ but

$$\mathscr{L}_{\text{adv}}\boldsymbol{U}+\mathscr{L}_{\text{diff}}\boldsymbol{U}-\mathscr{S}\neq 0 \tag{16}$$

This residual ignores all temporal variation, a significant error in unsteady flows.[4] This formulation also suffers the same pathologies described above regarding the diffusive term if implemented with piecewise linear-in-space functions.

The two incomplete residual implementations described above have been described herein as weakly consistent. This terminology deserves further discussion before we describe improvements that can be made. Before stabilization is added to Galerkin's method it can be shown by itself to be a weighted residual method. It is well known that the Galerkin terms drive the true residual to zero. Examining the Euler–Lagrange conditions of the Galerkin portion of (12) we obtain:

[4] It is worth admitting at this point that constant-in-time is not the best choice for time accurate calculations and that the problem automatically remedies itself for linear-in-time and higher but the authors have observed implementations of this type applied to unsteady problems justifying the attention given here.

$$0 = \sum_{e=1}^{n_{el}} \int_{\Omega^e} \boldsymbol{W} \cdot (\mathscr{L}\boldsymbol{U} - \mathscr{S})\, d\Omega - \sum_{e=1}^{n_{el}} \int_{\Gamma^e_{int}} \boldsymbol{W} \cdot \boldsymbol{F}_i^{diff} n_i\, d\Gamma + \int_{\Gamma_h} \boldsymbol{W} \cdot ((\boldsymbol{F}_i^{adv} - \boldsymbol{F}_i^{diff}) n_i - \mathscr{H})\, d\Gamma \tag{17}$$

where Γ^e_{int} is defined to be the faces of the e^{th} element excluding the boundaries of the physical domain (i.e. interior faces). With this definition, each interior face is encountered exactly twice with opposite signed normals giving a difference between the two element diffusive fluxes through the faces. It is this term which gives the Galerkin terms an ability to approximate the second derivative globally (even though the element integral defined in the first part of (17) cannot since $\mathscr{L}\boldsymbol{U}$ again will not contain a diffusive term on the element interior). Note that the last term in (17) accounts for the portion of the true boundary with a non-zero weight function, Γ_h, where the flux is weakly forced to match the prescribed flux, $\mathscr{H}$. In practice, various components of this flux are prescribed, while others are calculated. This issue is fully addressed in [17] and is omitted here in the interest of maintaining focus.

If we next consider the contribution of the stabilization term alone we see that it is already in its Euler–Lagrange form

$$\sum_{e=1}^{n_{el}} \int_{\Omega^e} \boldsymbol{P} \cdot (\mathscr{L}\boldsymbol{U} - \mathscr{S})\, d\Omega \tag{18}$$

with $\boldsymbol{P} = \boldsymbol{\tau}^{\mathrm{T}} \mathscr{L}^{\mathrm{T}}_{adv} \boldsymbol{W}$ the modified weighting space that multiples the $\mathscr{L}\boldsymbol{U} - \mathscr{S}$ which, for linear-elements, has already been shown to be potentially different from the true residual. The combination of these terms sets up a competition where the Galerkin term attempts to drive the real residual to zero (with the $\boldsymbol{W}$ weight) while the stabilization term tries to drive a different residual (with the $\boldsymbol{P}$ weight). Things are not as dire as they seem though since, the modified weighting space is proportional to $\boldsymbol{\tau}$ which vanishes as h goes to zero. As h vanishes the method defaults back to Galerkin's method and consistency is recovered.

Still, this situation is not completely satisfactory for two reasons. First, many popular definitions of the $\boldsymbol{\tau}$ matrix have a tendency to stagnate and not go to zero sufficiently fast. The problem here stems from the fact that h is supposed to be a measure of both space and time (Δx and Δt). The error analysis tells us that uniform convergence is attained only as both approach zero at the same rate. It is common practice, however, in many problems to refine only one leaving the other at some small but finite value. Such a practice will, with common definitions for $\boldsymbol{\tau}$, lead to stagnation and may therefore lead to apparent convergence to the wrong result.

The second, and perhaps more compelling reason to be unhappy with this notion of consistency, is that extremely fine meshes are needed to see consistent behavior. Indeed, stabilized methods were proposed so that good approximation properties could be attained on coarse meshes. It is not very satisfying that the method only becomes consistent after the stabilization parameter has become so small as to eliminate the stabilization term, leaving only the Galerkin term.

5. Residual completion

The above observations motivate the development of new stabilization operators that restore the stabilization term's ability to approximate the complete residual. The hope here is that improved consistency might lead to improved accuracy as well. In the next subsection we will discuss reconstruction of the spatial second derivative. Then, we will discuss the rather trivial reconstruction of the temporal first derivative.

5.1. Global reconstruction of spatial second derivative

Based on how we motivated the problem through Euler–Lagrange conditions, one might consider augmenting the stabilization term with internal face integrals like those found in (17). The new stabilization term might look like the following:

$$\sum_{e=1}^{n_{el}} \int_{\Omega^e} \boldsymbol{P} \cdot (\mathscr{L}\boldsymbol{U} - \mathscr{S})\, d\Omega - \sum_{e=1}^{n_{el}} \int_{\Gamma^e_{int}} \boldsymbol{P} \cdot \boldsymbol{F}_i^{diff} n_i\, d\Gamma \tag{19}$$

This approach is unattractive for two reasons: (1) it requires a face data structure (in addition to the element data

structure that we currently carry); (2) it requires the evaluation of modified weight function $\boldsymbol{P}$ (including the stabilization matrix, $\boldsymbol{\tau}$) over the faces of each element. Already, the evaluation of this quantity over the interior of the element is expensive and this would seem to be substantially more effort (factor of four increase in work for tetrahedra).

Instead, a different approach was pursued. Rather than mimic the Galerkin term, a new operator is proposed which has an ability to approximate the true residual. The new stabilization term can be written as follows:

$$\sum_{e=1}^{n_{\mathrm{el}}} \int_{\Omega_e} \mathscr{L}_{\mathrm{adv}}^{\mathrm{T}} \boldsymbol{W} \cdot \boldsymbol{\tau}(\hat{\mathscr{L}}\boldsymbol{U} - \mathscr{S})\, \mathrm{d}\Omega \tag{20}$$

where

$$\hat{\mathscr{L}}\boldsymbol{U} \equiv \frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}_i \frac{\partial \boldsymbol{U}}{\partial x_i} - \frac{\partial \boldsymbol{Q}_j}{\partial x_j} \tag{21}$$

and $\boldsymbol{Q}_j$ is the diffusive flux in the jth direction ($\boldsymbol{F}_j^{\mathrm{diff}}$) which must be reconstructed. The divergence of the reconstructed diffusive flux is given in terms of the the element shape functions as

$$\frac{\partial \boldsymbol{Q}_j}{\partial x_j}(\boldsymbol{x}_e) = \sum_{a=1}^{n_{\mathrm{en}}} N_{a,j}\, \boldsymbol{Q}_{aj} \tag{22}$$

where $\boldsymbol{Q}_{aj}$ is the diffusive flux in the jth direction at node a of element e. This quantity is not defined in the usual finite element sense due to the discontinuous nature of the derivatives of low-order piecewise polynomials. It can, however, be reconstructed to be a continuous variable using an L_2 projection operator. The procedure is described as follows:

$$\boldsymbol{M}\hat{\boldsymbol{Q}}_j = \boldsymbol{R}_j \tag{23}$$

$$\boldsymbol{M} = [\boldsymbol{M}_{AB}]; \qquad \boldsymbol{R}_j = \{\boldsymbol{R}_{Aj}\}; \qquad \hat{\boldsymbol{Q}}_j = \{\boldsymbol{Q}_{Bj}\} \tag{24}$$

$$\boldsymbol{M}_{AB} = \boldsymbol{I} \int_{\Omega} N_A N_B\, \mathrm{d}\Omega; \qquad \boldsymbol{R}_{Aj} = \int_{\Omega} N_A \boldsymbol{Q}_j(\boldsymbol{U}(\boldsymbol{x}))\, \mathrm{d}\Omega \tag{25}$$

where these matrices are computed in the usual finite element manner, assembling element level contributions to form the global counterparts. Since most solution procedures for the Navier–Stokes equations involve successive iterations of solving a linearization of the nonlinear problem, it is quite easy to lag the solution for $\boldsymbol{Q}_{aj}$ by one iteration. Furthermore, it is usually sufficient to replace $\boldsymbol{M}_{AB}$ by its lumped mass equivalent. We prefer to use the special lumping described in [13].

Before applying this new stabilization operator to numerical examples it is worthwhile to discuss the implications of its use. Because the reconstructed flux can no longer be simply related to the solution space $\boldsymbol{U}$, the use of $\boldsymbol{Q}_j$ constitutes a variational 'misdemeanor'. At the very least, this complicates the analysis of the method, an unhappy result since the rigorous analysis of these methods is one of their treasured attributes. However, the bounty of the 'crime' is a method which has the ability to approximate the true residual. That is to say that as we refine the mesh, not only does the stabilization parameter vanish, but the residual that it multiplies vanishes as well. So long as the the projection is accurate to the order of the Galerkin approximation of the diffusive term, consistency is restored. It is hoped that this *better* consistency will lead to *better* convergence. Finally, it is worth noting that this technique is only practical for SUPG since GLS would require reconstruction of the second derivative on the weight space which would involve significantly more effort (reconstruction of a sparse $(n_{sd} \times n_{np})^2$ matrix instead of a $n_{sd} \times n_{np}$ vector, where n_{np} is the total number of nodes in the problem).

It is worth noting that methods have been proposed by Behr et al. [1] which introduce additional variables for the diffusive fluxes which, when interpolated linearly, provide a non-vanishing gradient of the diffusive flux and therefore do not require the above method to complete the residual. This diffusive flux interpolation method, while yielding to a more straightforward analysis, can be unattractive for three-dimensional problems due to the large number of additional variables (7 additional fluxes yielding 12 degrees of freedom per node as opposed to the current system with 5 degrees of freedom per node).

5.2. Local reconstruction of spatial second derivative

With higher-order elements, the viscous term in the SUPG or GLS residual can be computed directly by using the second derivative of the basis, which requires the second derivative of the mapping. Unfortunately, this technique is both complicated to code and computationally expensive to use. In the previous section, a method was presented to globally reconstruct a continuous approximation to the diffusive flux for linear finite elements using an L_2 projection. Although this same method can also be used for higher-order elements, one is faced with additional computational expense since the use of mass lumping (even the special lumping of Hughes [13]) is not as effective for higher-order elements. However, an efficient and accurate technique can be developed based on a local projection of the diffusive flux to the nodes. Instead of forming a global mass matrix, a local L_2 projection is performed for each element. The procedure involves forming the element mass matrix and residual

$$\boldsymbol{M}_{ab} = \int_{\Omega^e} N_a N_b \, \mathrm{d}\Omega, \qquad \boldsymbol{R}_{aj} = \int_{\Omega^e} N_a \boldsymbol{Q}_j(\boldsymbol{U}(\boldsymbol{x})) \, \mathrm{d}\Omega \tag{26}$$

and solving the local projection problem

$$\boldsymbol{M}_{ab}\hat{\boldsymbol{Q}}_{bj} = \boldsymbol{R}_{aj} \tag{27}$$

for the reconstructed diffusive flux at the nodes, $\hat{\boldsymbol{Q}}_{bj}$. Here, the lower case a and b refer to element nodes. The Cholesky factorization of the matrix $\boldsymbol{M}_{ab}$ can be pre-computed and stored for each element type to make this a very efficient computation [6]. It should be noted that this local procedure does not work for linear elements since the diffusive flux is constant over the element interior.

5.3. Global reconstruction of temporal first derivative

Reconstruction of the temporal derivative for constant-in-time space–time formulations is a trivial task since most implementations have the solution at the previous time step available. Here

$$\hat{\mathscr{L}}\boldsymbol{U} \equiv \frac{\boldsymbol{U}(t_{n+1}^{+}) - \boldsymbol{U}(t_n^{+})}{\Delta t} + \boldsymbol{A}_i \frac{\partial \boldsymbol{U}}{\partial x_i} - \frac{\partial \boldsymbol{Q}_j}{\partial x_j} \tag{28}$$

is used in (20) to form a new stabilization operator.

It is worth mentioning again that this is only necessary for constant-in-time space time. Linear-in-time and higher have a complete representation of the temporal portion of the residual. Furthermore, semi-discrete methods do not need this correction so long as they are implemented properly with the mass matrix being modified by the addition of the $\mathscr{L}_{\mathrm{adv}}\tau\mathscr{L}_{\mathrm{t}}$ contribution.

6. Numerical examples

The formulation described in the previous section has been implemented in the three codes used by the various authors. Comparisons were made in each case between the original (incomplete-residual) implementation and the completed-residual implementation. As expected, on some flows there is a dramatic difference (where the second derivative terms (diffusion) are O(1)) while in other flows almost no difference was observed (where the effects of the second order terms (diffusion) are negligible). In each implementation the additional CPU time required per iteration was less than 5% of the total cost.

6.1. Parallel channel flow

The first flow considered is fully-developed, laminar flow in a channel between two flat plates of infinite width. Assuming the viscosity to be independent of temperature de-couples the energy equation allowing a closed-form, exact solution as follows:

$$u_1 = u_{\max}\left[1 - \left(\frac{x_2}{h}\right)^2\right]; \qquad u_2 = u_3 = 0 \tag{29}$$

$$T = T_w\left[1 + \frac{\mathrm{Pr}\,\hat{E}c}{3}\left\{1 - \left(\frac{x_2}{h}\right)^4\right\}\right]; \qquad p = p_o - \frac{2\rho_o u_{\max}^2}{\mathrm{Re}_h}\frac{x_1}{h} \tag{30}$$

where h is half the height of the channel, p_o is the pressure at $x_1 = 0$, ρ_o is the density at $x_1 = 0$, $u_{\max}$ is the centerline velocity, T_w is the prescribed wall temperature, $\mathrm{Pr} = \mu/(c_p\kappa)$ is the Prandtl number, $\mathrm{Re}_h = (u_{\max}h)/\nu$ is the Reynolds number based on the channel half-height, and $\hat{E}c = (u_{\max}^2)/(c_pT_w)$ is a modified Eckert number using the specific heat c_p. This solution holds for compressible or incompressible flow so long as the viscosity is assumed constant. It is a particularly interesting flow in our studies since the quadratic velocity profile allows a weighted residual method using piecewise-linear functions to obtain a nodally exact solution. The temperature field will not be exact since it is a quartic function but the constant viscosity prevents contamination of the velocity field.

This problem is also interesting for a second reason. Regardless of the Reynolds number, the velocity field is the same. This occurs because the advection term is identically zero due to the orthogonality of the velocity vector and its gradient. It is apparent from substituting the solution above into (1) that, for this flow, the momentum equation reduces to a balance of the pressure gradient with the viscous stress. It therefore stands to reason, that a method which neglects the viscous stress in the stabilization operator would neglect one of the terms in the balance leaving the other as an apparent spurious residual, even when the velocity field has a nodally exact solution. The effect of the incomplete-residual stabilization term is, in this case, to drive the flow away from the nodally exact solution that satisfies the Galerkin residual.

6.1.1. Inflow–outflow boundary conditions

There are two ways to prescribe the boundary conditions for parallel channel flow. The first is to prescribe the velocity and temperature profiles at the inflow and at the walls while prescribing the pressure at the outflow boundary. If a three-dimensional solver is used the w component of the velocity should also be set to zero on the side planes. This arrangement of boundary conditions is usually referred to as inflow–outflow for obvious reasons. The new formulation is compared to the standard implementation on a very course grid ($N_{x_1} = 10$, $N_{x_2} = 20$, $N_{x_3} = 2$) and at a Reynolds number of 100, Prandtl number of 0.7, Eckert number of 0.016 in Fig. 1. Here, the pressure field shows substantial oscillations when using the standard implementation while the

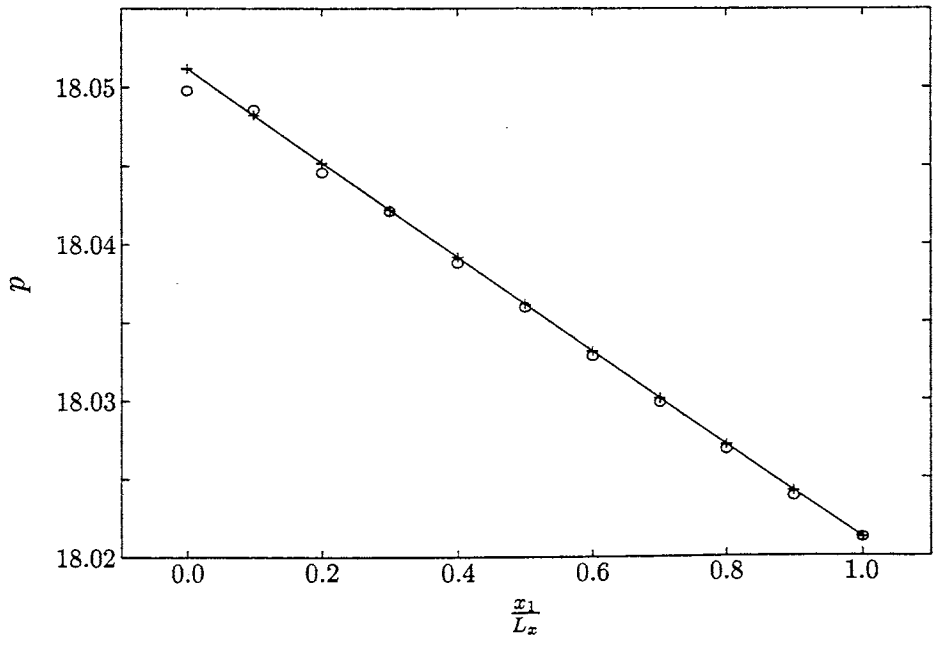

Fig. 1. Centerline pressure variation over length of the channel. Exact result ———, with residual completion +, without residual completion ○.

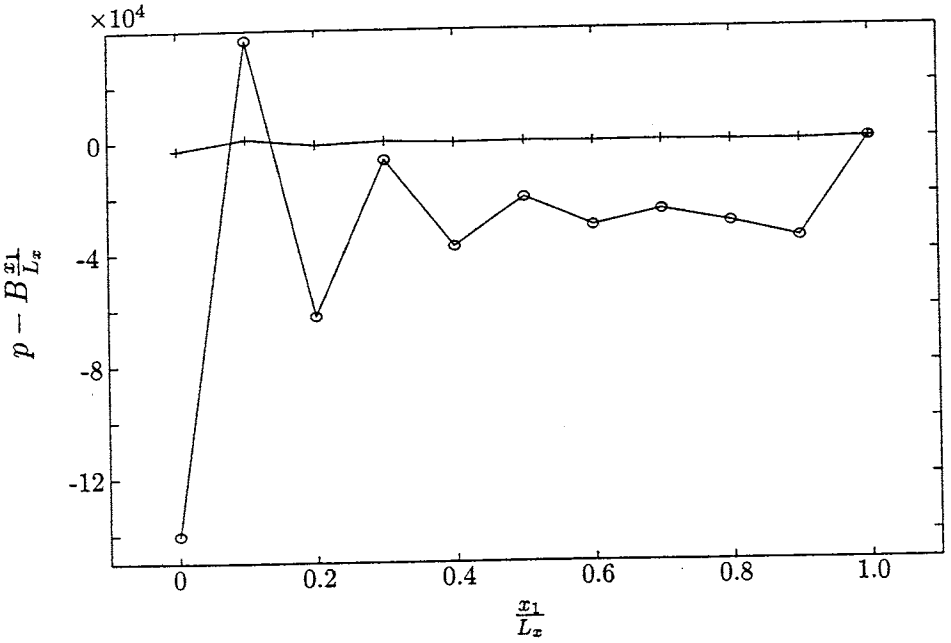

Fig. 2. Error in pressure variation over length of the channel. With residual completion +, without residual completion ○.

completed-residual method shows a dramatic improvement. Subtracting out the linear portion of the pressure gradient illustrates the error reduction of the new method versus the old as shown in Fig. 2. Based on our preliminary findings for small disturbance evolution described in section 6.3, Hauke [8] implemented the same global reconstruction described in 5.1 for the inflow–outflow parallel channel flow and observed a similar improvement.

6.1.2. Periodic boundary conditions

An alternative boundary condition prescription is obtained by forcing all variables on the outflow plane to be periodic with those on the inflow plane. To do this requires that the linear portion of the pressure be interpreted as a body force.

$$p = p_o + B\frac{x_1}{h} \tag{31}$$

Then, the constant portion, p_o can be assumed periodic along with the other variables, u_i and T, which are not functions of x_1. The linear portion ($B = -(2\rho_o u_{\max}^2)/\mathrm{Re}_h$ is a constant coefficient) of the pressure term becomes a source term. Under these boundary conditions, on the same equispaced mesh both the original method and the completed-residual method attain a nodally exact velocity field and a constant pressure field. While this result was expected for the completed-residual method it was somewhat surprising for the original method. To understand the origin of this fortuitous result requires careful examination of the stabilization term. Let $f = -(2\mu u_{\max})/h^2 = -(2\rho_o u_{\max}^2)/(\mathrm{Re}_h h) = B/h$, $\tau_m = \tau_m(x_2)$, $u_1 = u_1(x_2)$, then the x_1 momentum SUPG stabilization term reduces to:

$$\sum_{e=1}^{n_{\mathrm{el}}} \int_{\Omega_e} (N_{a,x_1} u_1 \tau_m (\mu u_{1,x_2x_2} + f))\, \mathrm{d}\Omega \tag{32}$$

At first sight one would expect the neglect of the viscous term (which should cancel exactly with the body force f) would create problems since the term would effectively be:

$$\sum_{e=1}^{n_{\mathrm{el}}} \int_{\Omega_e} (N_{a,x_1} u_1 \tau_m f)\, \mathrm{d}\Omega \tag{33}$$

however, under the assumptions above, in combination with a mesh of right hexahedra the stabilization term further simplifies to:

$$\sum_{e=1}^{n_{el}} \int_{x_3} \mathrm{d}x_3 \int_{x_2} u_1 \tau_m \, \mathrm{d}x_2 \int_{x_1} (N_{a,x_1} f) \, \mathrm{d}x_1 \tag{34}$$

If we further assume an equispaced grid in x it is easily shown that, for linear elements, the contribution to a node from the left element will be the exact negation of the contribution from the right element due to the changing sign on N_{a,x_1} (and a constant f). Therefore, even though each element has a large residual owing to the neglect of the viscous term, the assembled result enjoys a cancellation error. This fortuitous result is easily foiled by resolving the problem on a mesh without equispaced elements on which the completed-residual method enjoys a substantial accuracy advantage, see Fig. 3 where the mesh spacing has been altered to $3/4\Delta_{x_1}, 1/4\Delta_{x_1}, 3/4\Delta_{x_1}, \ldots$. The results in Fig. 3 indicate that the magnitude in the error without the completed-residual is more than 70 times greater than the error with the completed-residual. It should be pointed out that the completed-residual method could be made nodally exact here by the solution of a consistent mass matrix (currently we opt for the efficient lumped-mass reconstruction). The above analysis also elucidates the source of the error in the inflow–outflow case since, on the boundary, the above described cancellation does not occur. This is consistent with Fig. 1 where we see a large error emanating from the inflow boundary.

6.2. Vortex shedding behind a cylinder

To illustrate the effects of an incomplete residual in a constant-in-time space–time solution strategy, vortex shedding behind a cylinder is studied. A complete problem description can be found in variety of references [2,10,17]. We choose boundary conditions consistent with those of Shakib et al. [17]. At Reynolds number of 100, the salient feature of this flow is the periodic shedding of vortices from the cylinder creating a time varying lift determined by integrating the forces on the cylinder surface. The results of the three forms of residual completeness can be seen in Fig. 4. Without the time term in the residual, the amplitude is sharply lower and the period of the oscillation is significantly longer and farther away from the experimental value of Roshko [15]. In this case the method is effectively an artificial diffusion method since only the advective terms remain in the

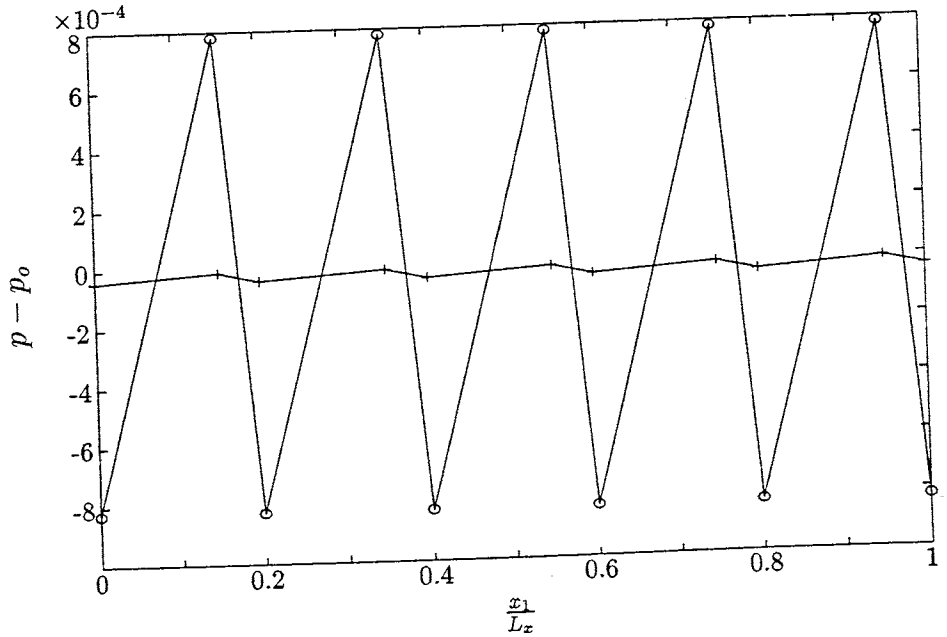

Fig. 3. Error in the pressure across the length of the channel. With residual completion +, and without residual completion ○.

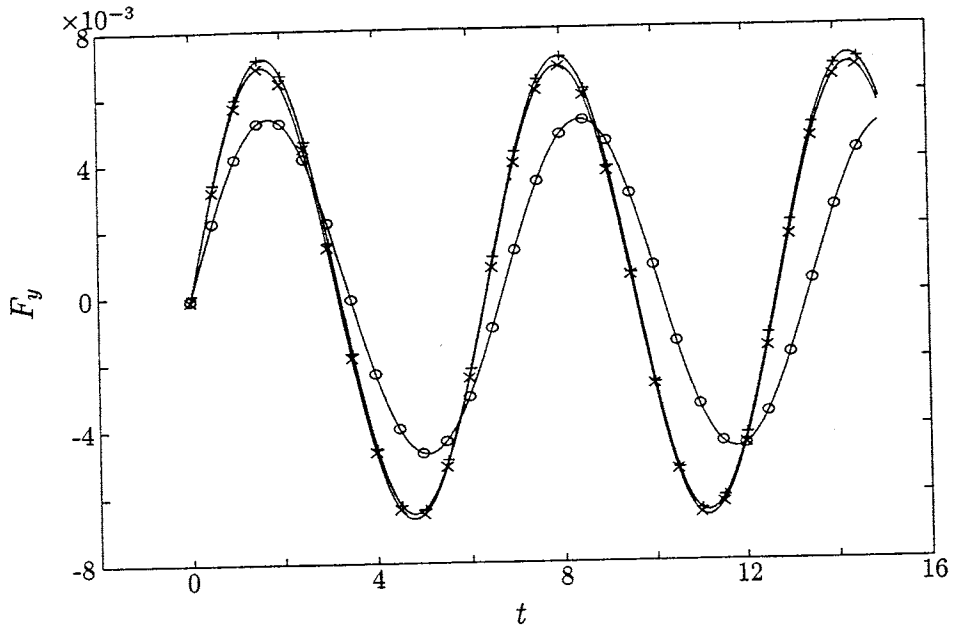

Fig. 4. Lift on the cylinder without residual completion ○, with residual completion of time +, and with residual completion of time and space ×.

incomplete residual. The other two curves (with the temporal reconstruction and with temporal and second-spatial derivative reconstruction) are not substantially different. This is not surprising since the lift forces are dominated by advection leaving little effect to the inclusion of diffusive residual. The time step used in these studies corresponds to 60 steps per period of the drag (120 steps per period of the lift). The nonlinear problem was converged four orders of magnitude in each step using three linearization passes, wherein the normalized residual of the linear problem was reduced to 5×10^{-3} in each pass. These results are quite insensitive to further convergence of the nonlinear problem.

6.3. Growth of small disturbances in a laminar boundary layer

A major feature of boundary layer transition from laminar to turbulent flow is the formation and growth of Tollmien–Schlichting (TS) instability waves. The mechanism which leads to the growth of TS waves is a viscous instability that makes the numerical simulation of TS waves a particularly sensitive test of a numerical method's ability to accurately represent viscous phenomena. In fact, it was through numerical experimentation with TS waves that the inconsistency of dropping the viscous term for linear finite elements within the SUPG residual was first identified and corrected using the methods of Section 5. In this section, we briefly describe a model problem for the simulation of TS waves and outline the modifications required to accurately solve this type of problem using an existing SUPG code. We then demonstrate that dropping the viscous term in the SUPG residual introduces an error which causes the method to converge to the incorrect instability growth rate. Finally, we show how the viscous term can be successfully and efficiently reconstructed using the methods of Section 5 so that convergence to the theoretical solution is retained.

6.3.1. Problem description

As a model problem for the growth of instability waves in a laminar boundary layer, we solve for the temporal growth of TS waves in a parallel boundary layer, i.e. a boundary layer which does not grow in the streamwise direction. Using a parallel boundary layer greatly simplifies the problem and allows direct comparison to theoretical analysis based on the parallel flow stability equations. By considering temporally growing waves, the computational domain can be limited to a single wavelength of the disturbance with periodic boundary conditions in the streamwise direction. This is in contrast to the more realistic, yet computationally

complex, spatial problem where the computational domain must be several wavelengths long with accurate inflow and outflow boundary conditions. A thorough discussion of both the temporal and spatial problems can be found in [5] in the context of high-order finite difference methods.

The mean boundary layer is a solution to the compressible Falkner–Skan–Cooke equations [5] for an unswept, zero pressure-gradient boundary layer. For simplicity we use an ideal, calorically perfect gas. In addition, the fluid properties such as the viscosity, μ, and thermal conductivity, κ, are assumed constant. The flow is characterized by the following non-dimensional parameters: Mach number $\mathrm{M} = u_{1_e}/c_e = 0.3$, Reynolds number $\mathrm{Re} = \rho_e u_{1_e}\delta^*/\mu = 1000$, and Prandtl number $\mathrm{Pr} = \mu c_p/\kappa = 1.0$. In these expressions a subscript 'e' refers to a quantity at the edge of the boundary layer, u_1 is the streamwise velocity, c is the speed of sound, ρ is the density, δ^* is the boundary layer displacement thickness, and c_p is the specific heat at constant pressure. The wall is assumed to be isothermal at temperature $T_w = T_0$ where $T_0 = T_e(1 + (\gamma - 1)/2\ M^2)$ is the freestream stagnation temperature,[5] $\gamma = c_p/c_v$, and c_v is the specific heat at constant volume.

Given the mean boundary layer, a Linear Stability Theory (LST) solver [5] based on Chebyshev collocation with 96 points in the wall normal direction is used to compute the reference growth-rate. A streamwise wavenumber of $\alpha = 0.308620690/\delta^*$ is used which gives a theoretical frequency of

$$\omega_{\mathrm{LST}} = (0.114678806 + 0.002384453i)\,\frac{u_{1_e}}{\delta^*} \tag{35}$$

and the growth-rate is given by the imaginary part, $\sigma_{\mathrm{LST}} = 0.002384453\ u_{1_e}/\delta^*$. We note that the eigenvalues from the LST calculation are believed to be accurate to the number of digits indicated. This has been verified by calculations with 128 Chebyshev points which show no change to 9 decimal places.

6.3.2. Implementation

The numerical simulation of boundary layer instabilities, at least in the linear regime, is particularly challenging since the wave amplitudes are several orders of magnitude smaller than the boundary layer edge velocity. Furthermore, the growth (or decay) of the waves is extremely sensitive to changes in the mean flow and to numerical errors including: truncation errors, errors in the initial condition, and errors due to approximate boundary conditions. These problems have typically been overcome by using high accuracy spectral or finite difference methods. Additionally, if one is only interested in the linear evolution of disturbances, the Navier–Stokes equations can be solved in disturbance form which removes the difficulty of simultaneously resolving both the mean flow and disturbance flows and prevents errors in the mean flow from corrupting the disturbance flow solution.

To address these issues within the context of a stabilized finite element method, we have modified the SUPG code of Shakib et al. [17] to solve for the evolution of TS waves in a parallel boundary layer. Modifications include second-order accurate midpoint time integration and periodic boundary conditions. Since our model problem uses a parallel mean boundary layer, which is not a steady solution to the Navier–Stokes equations, we have also added a source term which effectively makes the parallel boundary layer an exact solution to a forced set of equations.

To construct the appropriate source term, the solution to the discrete equations is written as

$$\boldsymbol{U} = \bar{\boldsymbol{U}} + \epsilon\tilde{\boldsymbol{U}} \tag{36}$$

where $\bar{\boldsymbol{U}}$ is an arbitrary base flow (i.e. the parallel boundary layer), $\tilde{\boldsymbol{U}}$ is the disturbance flow which consists of the instability waves in this case, and ϵ is the non-dimensional amplitude of the disturbances. Plugging this expression into the residual of the discrete Navier–Stokes equations leads to

$$\boldsymbol{R}(\bar{\boldsymbol{U}} + \epsilon\tilde{\boldsymbol{U}}) = \mathscr{B} \tag{37}$$

where the operator $\boldsymbol{R}$ is compact notation for the SUPG finite element method and $\mathscr{B}$ is the desired source term which ensures that the base flow remains stationary. This equation can be rewritten as

$$\boldsymbol{R}\bar{\boldsymbol{U}} + \epsilon\boldsymbol{R}_l[\bar{\boldsymbol{U}}]\tilde{\boldsymbol{U}} + \mathrm{O}(\epsilon^2)\boldsymbol{R}_n[\bar{\boldsymbol{U}}]\tilde{\boldsymbol{U}} = \mathscr{B} \tag{38}$$

[5] For $\mathrm{Pr} = 1$, T_0 is also the adiabatic wall temperature.

where $\boldsymbol{R}_l[\bar{\boldsymbol{U}}]$ denotes the discrete finite element equations linearized about the base flow, $\boldsymbol{R}_n[\bar{\boldsymbol{U}}]$ is an operator representing the nonlinear evolution of the disturbances, and $O(\epsilon^2)$ indicates that the amplitude of the nonlinear term is at least ϵ^2. If we set

$$\mathscr{B} = \boldsymbol{R}\bar{\boldsymbol{U}} \tag{39}$$

then when we solve (37) we are actually solving

$$\boldsymbol{R}_l[\bar{\boldsymbol{U}}]\tilde{\boldsymbol{U}} + O(\epsilon)\boldsymbol{R}_n[\bar{\boldsymbol{U}}]\,\tilde{\boldsymbol{U}} = 0 \tag{40}$$

which is exactly the equation that governs the (possibly nonlinear) evolution of disturbances about the stationary base flow $\bar{\boldsymbol{U}}$. In practice, we use $\epsilon = 10^{-4}$ such that nonlinearities are negligible and the disturbances evolve linearly according to

$$\boldsymbol{R}_l[\bar{\boldsymbol{U}}]\tilde{\boldsymbol{U}} = 0. \tag{41}$$

To implement this method into the existing finite element code, we first start the code with an initial condition composed of the base flow, $\bar{\boldsymbol{U}}$, and compute and store the residual $\boldsymbol{R}\bar{\boldsymbol{U}}$ for the first time-step. The code is then restarted using the base flow plus the disturbances as the initial condition and, after forming the complete residual at each time step, the base flow residual is subtracted yielding Eq. (40). It is important to note that all numerical parameters (including Δt and τ) must be the same when computing the base flow residual and the total residual. With the method properly implemented, the disturbance solution at any time step can be recovered using the expression $\tilde{\boldsymbol{U}} = (\boldsymbol{U} - \bar{\boldsymbol{U}})/\epsilon$.

6.3.3. Results

Using the modified SUPG code for the problem described in Section 6.3.1, we have performed numerical simulations of temporally growing TS waves using both bi-linear and bi-quadratic elements. For both element types, spatial convergence studies have been performed to determine the asymptotic growth rate predicted by the numerical simulation. Growth-rate results from the finite element calculations are based on the integrated, disturbance kinetic-energy

$$E_k' \equiv \frac{1}{2}\int_0^\infty (|\hat{u}_1'|^2 + |\hat{u}_2'|^2)\,\mathrm{d}y \tag{42}$$

where a hat denotes the streamwise Fourier transform of the disturbance velocities for wavenumber α. Given E_k' the growth rate is given by

$$\sigma = \frac{1}{2}\frac{1}{E_k'}\frac{\mathrm{d}E_k'}{\mathrm{d}t} \tag{43}$$

where the factor $1/2$ is required since E_k' is a quadratic quantity.

The base flow field is constructed by interpolating the Falkner–Skan–Cooke mean boundary layer profile onto the finite element mesh using cubic spline interpolation. The initial disturbance field is obtained by spectrally interpolating the linear stability theory eigenfunction onto the finite element mesh. Thus, the disturbance initial-condition is a nodally exact solution to the continuous problem. A uniform mesh is used in the streamwise direction along with periodic boundary conditions. A stretched mesh is used in the wall normal direction based on the transformation

$$x_2(\eta) = \frac{a\eta}{b-\eta} \tag{44}$$

where

$$a = \frac{x_{2_{\max}} x_{2_s}}{x_{2_{\max}} - 2x_{2_s}}, \qquad b = 1 + \frac{a}{x_{2_{\max}}}, \quad \text{and} \quad \eta \in [0, 1]. \tag{45}$$

For all the computations reported here, $x_{2_{\max}} = 80\delta^*$ and $x_{2_s} = \delta^*$. With this large value of $x_{2_{\max}}$, fixing the pressure on the upper boundary is an adequate boundary condition although a zero disturbance condition yields

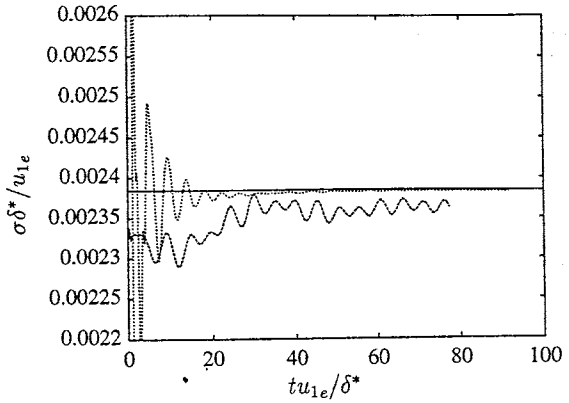

Fig. 5. Typical time history of the growth rate, σ: ——— linear stability theory, – – – 21×127 linear elements, - - - - - - 21×127 quadratic elements.

nearly identical results. The mesh resolution is specified as $N_1 \times N_2$ where N_1 is the number of node points in the streamwise direction, x_1, and N_2 is the number of nodes in the vertical direction, x_2.

Fig. 5 shows typical time histories of the growth rate for a 21×127 node mesh with both linear and quadratic elements using the reconstructed viscous flux (see Section 5). The time step for this and all the computations reported here is $\Delta t = 0.0478\delta^*/u_{1e}$ which corresponds to approximately $N_t = 1150$ time steps per period of TS wave oscillation. Previous experience [5] has shown that this time resolution is sufficient to reduce the relative temporal error in the growth rate to less than 10^{-6} which allows us to focus solely on spatial errors. As expected, Fig. 5 demonstrates that the asymptotic error in the quadratic element solution is considerably less than that in the linear element solution. As indicated above, the growth rate is a very sensitive measure of errors in a numerical solution. The oscillations seen in Fig. 5 are due to interpolation and discretization errors in the initial time step which initialize spurious acoustic waves in the domain. Typically there is a transient from $t = 0$ to $t \approx 40\delta^*/u_{1e}$ after which the average growth rate is constant. However, due to the slightly damped acoustic waves the growth rate always exhibits oscillations in time, although the oscillations are reduced as the resolution is increased. Typical of the results using quadratic elements, the initial oscillations in the growth rate are much more severe than those for linear elements, although these oscillations decay quickly yielding an improved average growth rate with smaller oscillations at large times compared to linear elements.

The asymptotic growth rate, $\bar{\sigma}$, is extracted from the time histories shown in Fig. 5 by computing the time-average growth-rate, starting after one period of the TS wave oscillation, $t \approx 55\delta^*/u_{1e}$, which ensures that the initial transient is complete. In the following convergence studies, the relative error in the asymptotic growth rate is determined using

$$E_{\bar{\sigma}} = \frac{|\bar{\sigma} - \sigma_{\mathrm{LST}}|}{\sigma_{\mathrm{LST}}}. \tag{46}$$

We begin by considering bi-linear and bi-quadratic finite element solutions which include the diffusive flux reconstruction described in Section 5. For bi-linear elements, global reconstruction with lumped mass is used, while local reconstruction is used for the bi-quadratic case. Fig. 6 shows the relative error in the computed growth-rate for both element types as N_1 is increased with N_2 fixed as 127. With the reconstructed viscous term in the SUPG residual, the bi-linear finite element solution converges to the exact growth rate at second order and the bi-quadratic converges at third order. Of course, the error will eventually saturate due to discretization errors in x_2 and t, but for the relatively high values of N_2 and N_t used here, we observe optimal convergence of the growth rate with increases in N_1.

However, if viscous reconstruction is not used for linear elements, then the convergence behavior is as shown in Fig. 7. In this case, the error initially drops but then *increases* as N_1 is increased. To further illustrate the error incurred by dropping the viscous term in the SUPG residual, Fig. 7 also shows results for quadratic elements where the SUPG viscous term has been artificially dropped. As N_1 is increased, the relative error also increases

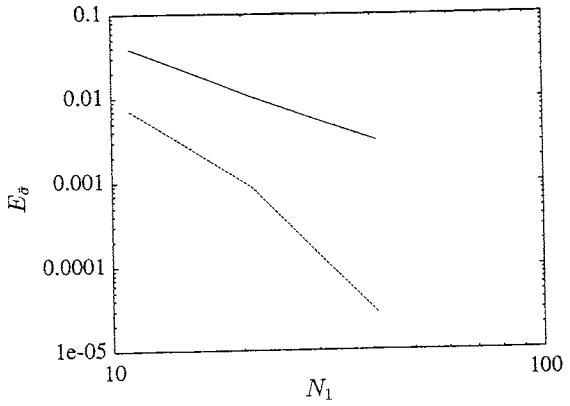

Fig. 6. Convergence of the temporal growth rate for ——— linear and – – – quadratic elements with viscous reconstruction in the SUPG residual.

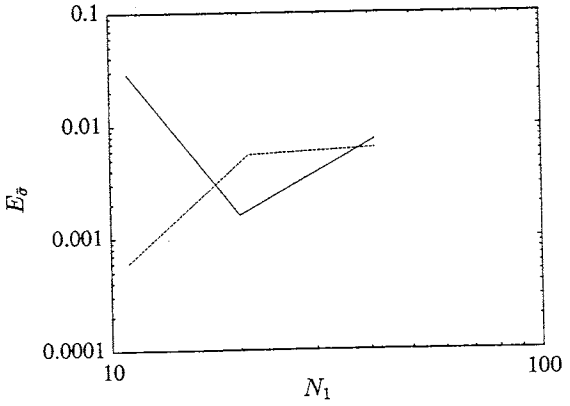

Fig. 7. Convergence of the temporal growth rate for ——— linear and – – – quadratic elements with no viscous term in the SUPG residual.

and eventually saturates at a value near 0.008. Thus, without the SUPG viscous term, the numerical solutions do not converge to the theoretical solution, but instead the error saturates at a fixed value of approximately 0.008. Although this error may eventually decrease for values of N_1 larger then those considered, depending on the definition of h in τ, it is clear that the behavior exhibited by using an incomplete SUPG residual is undesirable. Although SUPG for linear finite elements is a weighted residual method, we find that without reconstructing the viscous flux in the SUPG residual, the numerical solution inconsistent. In fact, similar behavior is seen if the SUPG viscous term is, arbitrarily, dropped for quadratic elements.

To determine the computational expense associated with forming the reconstructed viscous term, we have compared the average CPU time per time-step for the methods with and without viscous reconstruction. For linear elements, global reconstruction of the diffusive flux results in 3% greater CPU time per time-step. This is judged to be a negligible additional expense given the improvement in convergence behavior. For the local reconstruction used with quadratic elements, the additional expense is only 0.9% greater CPU time per time-step which is trivial compared with the expense of computing the diffusive flux using the second derivative of the mapping.

In conclusion, by including the reconstructed SUPG viscous term, bi-linear and bi-quadratic finite element solutions are shown to converge monotonically to the exact solution. However, without the viscous term in the SUPG residual, the error can increase as more elements are used and can eventually saturate at a fixed error relative to the exact solution. By adding a reconstructed viscous term to the SUPG residual, the optimal order convergence of the finite element method is preserved at negligible cost for both linear and quadratic elements.

6.4. Turbulent flow over a backward facing step

The final numerical example is that of flow over a backward facing step at a Reynolds number of 45,000. The Reynolds number and expansion ratio, $H/(H+h) = 1.5$, allow comparison with Kim et al. [14]. Here, H is the incoming channel flow height and h is the step height. The high Reynolds number makes a Reynolds-averaged Navier–Stokes simulation appropriate. The Spalart–Allmaras [18] eddy viscosity model was used. The mesh employed 7200 hexahedral elements. In Fig. 8, the streamlines of the solution are shown. In this flow, the re-attachment length ratio, x_r/h, is an important quantity to predict. The convergence of this quantity is shown in Fig. 9 for the incomplete residual method and the completed residual method. Both results are within the error range of the experiment; incomplete-residual $x_r/h = 7.08$, completed-residual $x_r/h = 7.03$, experiment $x_r/h = 7.0 \pm 0.5$. The only noticeable difference between the two approaches is that the completed residual method seems to converge slightly faster. Again, this flow is not affected strongly by the inclusion of the viscous term due to the high Reynolds number which reflects the dominance of advection over diffusion.

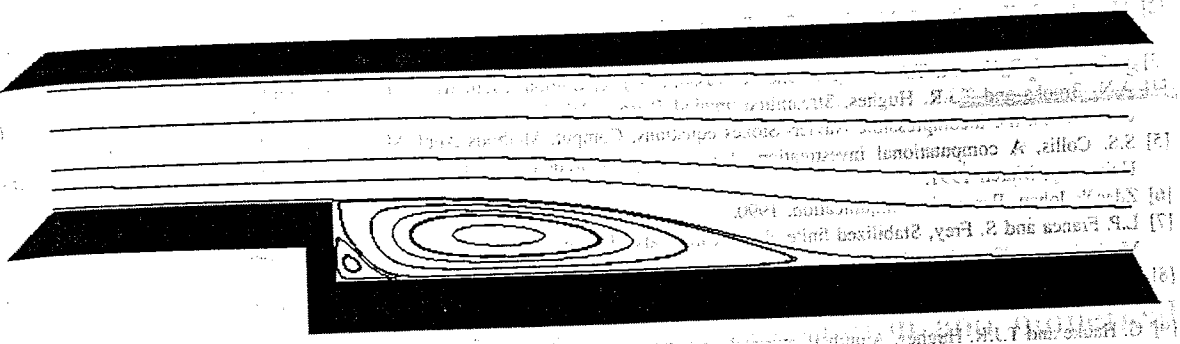

Fig. 8. Streamlines for flow over a backward facing step.

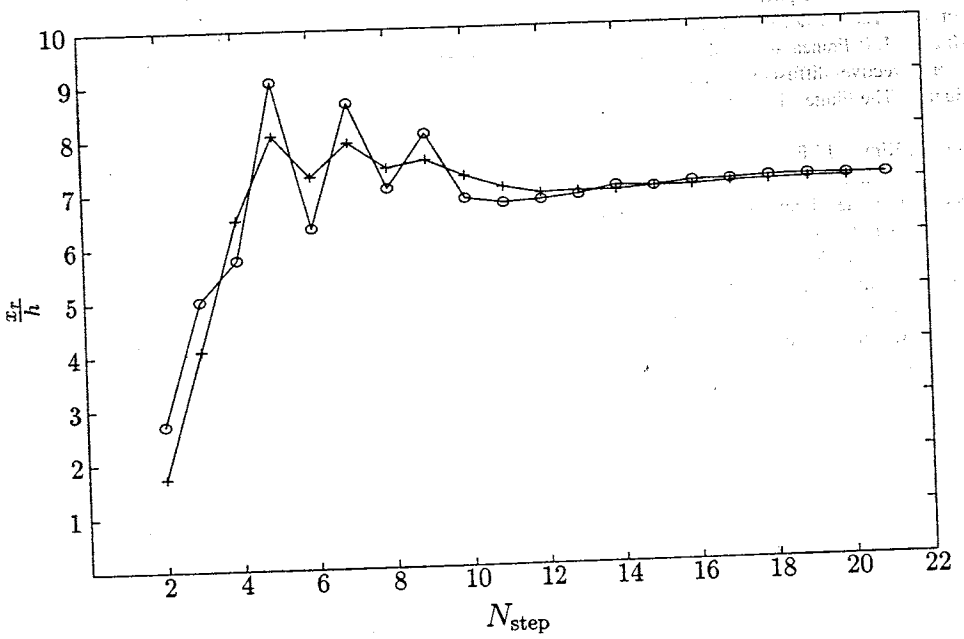

Fig. 9. Convergence of re-attachment length ratio, x_r/h, with time step. With residual completion +, without residual completion ○.

7. Conclusions

Stabilized finite element implementations with polynomial order lower than the highest-order derivative present were shown to possess a weak consistency which could be strengthened through a new form of the stabilization operator. The new stabilization operator accomplishes this feat by globally reconstructing the higher derivative information which is lost at the local element level. The performance of this new operator was tested and shown, in some cases, to greatly improve accuracy. The low cost of the approach (less than 5% of the total calculation) coupled with the fact that in no case did the new operator lead to a poorer solution leads us to conclude that the method is superior to previous low-order stabilized method implementations.

References

[1] M. Behr, L.P. Franca and T.E. Tezduyar, Stabilized finite element methods for the velocity–pressure–stress formulation of incompressible flows, Comput. Methods Appl. Mech. Engrg. 104 (1993) 31–38.

[2] M. Behr, D. Hastreiter, S. Mittal and T.E. Tezduyar, Incompressible flow past a circular cylinder: dependence of the computed flow field on the location of the lateral boundaries, Comput. Methods Appl. Mech. Engrg. 123 (1996) 309–316.
[3] F. Brezzi, L.P. Franca, T.J.R. Hughes and A. Russo, $b = \int g$, Comput Methods Appl. Mech. Engrg. 145 (1997) 329–339.
[4] A.N. Brooks and T.J.R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 32 (1982) 199–259.
[5] S.S. Collis, A computational investigation of receptivity in high-speed flow near a swept leading-edge, Ph.D. Thesis, Stanford University, March 1997.
[6] Zdeněk Johan, Personal communication, 1990.
[7] L.P. Franca and S. Frey, Stabilized finite element methods: II. The incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 99 (1992) 209–233.
[8] G. Hauke, A unified approach to compressible and incompressible flows and a new entropy-consistent formulation of the k-ϵ model, Ph.D. Thesis, Stanford University, 1995.
[9] G. Hauke and T.J.R. Hughes, A unified approach to compressible and incompressible flows, Comput. Methods Appl. Mech. Engrg. 113 (1994) 389–396.
[10] G. Hauke and T.J.R. Hughes, A comparative study of different sets of variables, Comput. Methods Appl. Mech. Engrg. 153 (1998) 1–44.
[11] T.J.R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Engrg. 127 (1995) 387–401.
[12] T.J.R. Hughes, L.P. Franca and G.M. Hulbert, A new finite element formulation for fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations, Comput. Methods Appl. Mech. Engrg. 73 (1989) 173–189.
[13] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis (Prentice Hall, Englewood Cliffs, NJ, 1987).
[14] S.J. Klein, J.J. Kim and J.P. Johnston, Investigation of separation and reattachment of a turbulent shear layer, Technical Report MD-37, Dept. of Mech. Eng., Stanford University, 1978.
[15] A. Roshko, On the development of turbulent wakes and vortex streets, Technical Report 1191, NACA, 1954.
[16] A. Ruśso, Bubble stabilization of the finite element methods for the linearized incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 132 (1996) 335–343.
[17] F. Shakib, T.J.R. Hughes and Z. Johan, A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 89 (1991) 141–219.
[18] P.R. Spalart and S.R. Allmaras, A one-equation turbulence model for aerodynamic flows, in: AIAA Paper 92-439, 1992.