INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **57**:1189–1209
Published online 22 January 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1743

Stabilized formulations for incompressible flows with thermal coupling

Tayfun E. Tezduyar*, †, Srinivas Ramakrishnan and Sunil Sathe

Mechanical Engineering, Rice University-MS 321, 6100 Main Street, Houston, TX 77005, U.S.A.

SUMMARY

We present applications of the stabilized finite element formulations developed for incompressible flows with thermal coupling to 2D and 3D test problems. The stabilized formulations are based on the streamline-upwind/Petrov–Galerkin and pressure-stabilizing/Petrov–Galerkin stabilizations and are supplemented with discontinuity capturing (DC), including the discontinuity-capturing directional dissipation. The stabilization and DC parameters associated with these formulations are also presented. The coupled fluid mechanics and temperature equations are solved with a direct coupling technique. The test problems computed include 2D and 3D natural convection, as well as a simplified 3D model of air circulation in a small data center. Copyright © 2008 John Wiley & Sons, Ltd.

Received 15 October 2007; Revised 1 December 2007; Accepted 5 December 2007

KEY WORDS: incompressible flow; thermal coupling; finite elements; SUPG/PSPG formulations; discontinuity capturing

1. INTRODUCTION

Stabilized formulations are now very commonly used in finite element computation of flow problems. These formulations bring numerical stability in flow problems with high Reynolds or Mach numbers and shocks or thin boundary layers, without introducing excessive numerical dissipation. They also bring numerical stability in incompressible flow computations when using equal-order interpolation functions for velocity and pressure. Some of the earliest stabilized formulations are the streamline-upwind/Petrov–Galerkin (SUPG) formulation for incompressible flows [1, 2] and the SUPG formulation for compressible flows [3, 4]. The stabilized formulations introduced in [5] for advection–diffusion equations and in [6] for advection–diffusion–reaction equations

*Correspondence to: Tayfun E. Tezduyar, Mechanical Engineering, Rice University-MS 321, 6100 Main Street, Houston, TX 77005, U.S.A.
†E-mail: tezduyar@rice.edu

Contract/grant sponsor: NSF; contract/grant number: CNS-0421109

included discontinuity-capturing (DC) terms. The formulation in [6] accounted for the interaction between the DC and SUPG terms by precluding the augmentation of the SUPG effect by the DC effect when the advection and discontinuity directions coincide. The formulation in [6] also included stabilization terms, which were called the 'DRD' terms, to counter the numerical instabilities seen in reaction-dominated problems. The formulations introduced in [6] were applied in [6] for the computation of coupled chemical and thermal transport problems governed by three coupled advection–diffusion–reaction equations—one governing the temperature and the other two governing the concentrations of two chemical species. These formulations were extended in [7] for the computation of time-dependent versions of this class of coupled chemical and thermal transport problems. The pressure-stabilizing/Petrov–Galerkin (PSPG) formulation for incompressible flows [8, 9] assures numerical stability while allowing us to use equal-order interpolation functions for velocity and pressure. An earlier version of this stabilized formulation for Stokes flow was introduced in [10].

In these stabilized formulations, an embedded stabilization parameter most commonly known as 'τ' plays an important role. It involves representation of the local length scales (also known as 'element length') and other parameters such as the local flow velocity and the time-step size. Various element lengths and τ's were proposed starting with those in [1–4], followed by the one introduced in [6]. The τ definition introduced in [6] automatically yields lower values for higher-order elements. Later, other τ definitions that are applicable to higher-order elements were proposed in [11] in the context of advective–diffusive systems. Calculating the τ's based on the element-level matrices and vectors was introduced in [12] in the context of the advection–diffusion equation and the Navier–Stokes equations of incompressible flows. These definitions are expressed in terms of the ratios of the norms of the matrices or vectors. They automatically take into account the local length scales, advection field and the element Reynolds number. These definitions were extended in [13, 14] to compressible flows.

The discontinuity-capturing directional dissipation (DCDD) stabilization was introduced in [15, 16] to be used with the SUPG/PSPG formulation of incompressible flows, in flow problems with sharp gradients. The DCDD stabilization involves a second element length scale, which was also introduced in [15, 16] and is based on the solution gradient. This new element length scale is used together with the element length defined in [6]. Recognizing this second element length as a diffusion length scale, new stabilization parameters for the diffusive limit were introduced in [16, 17]. The DCDD stabilization was originally conceived in [15, 16] as an alternative to the least-squares on incompressibility constraint (LSIC) stabilization. The DCDD takes effect where there is a sharp gradient in the velocity field and introduces dissipation in the direction of that gradient. In some versions of the DCDD stabilization, the way the DCDD is added to the formulation precludes augmentation of the SUPG effect by the DCDD effect when the advection and discontinuity directions coincide. Recently, the SUPG/PSPG formulation supplemented with the DCDD stabilization was extended in [18] to turbulent flow computations. It was shown in [18], that this DCDD-supplemented formulation is very comparable in numerical performance to the SUPG/PSPG formulation supplemented with the Smagorinsky turbulence model. Also recently, the 'DRD' stabilization introduced in [6] and its improved versions were applied in [19] to turbulent flow computations, and the results obtained show a good potential for this approach.

Various forms of stabilized methods have also been applied to incompressible flows with thermal coupling. For example, in finite element simulations of incompressible flows and coupled reaction–convection–diffusion processes reported in [20], the SUPG formulation is used for the temperature and species transport. In [20], the fluid mechanics equations are discretized with the Galerkin

DOI: 10.1002/fld

formulation, and the incompressibility constraint is enforced with a penalty formulation. The SUPG/PSPG formulation of incompressible flows with thermal coupling and the associated stabilization and DC parameters were presented in [21] in the context of the ALE formulation [22]. The DC stabilizations presented in [21] included the extension of the DCDD stabilization to the temperature equation. In this paper, we apply those formulations to a number of 2D and 3D test problems. The test problems do not involve any moving boundaries or interfaces; therefore, the formulations described in the context of the ALE formulation are used with zero mesh velocity. The coupled fluid mechanics and temperature equations are solved with a direct coupling technique (see [21] for terminology). The test problems computed include 2D and 3D natural convection problems, as well as a simplified 3D model of air circulation and cooling in a small data center. We describe the governing equations in Section 2. The stabilized formulations, stabilization parameters and the DC parameters are described in Sections 3–5. Test computations and concluding remarks are presented in Sections 6 and 7.

2. GOVERNING EQUATIONS

Let $\Omega_t \subset \mathbb{R}^{n_{sd}}$ be the spatial domain with boundary Γ_t at time t. The subscript t allows for the time dependence of the domain. The coupled incompressible flow and thermal transport equations are expressed on Ω_t as

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}-(1-\beta_{\mathrm{T}}(T-T_{\mathrm{ref}}))\mathbf{a}_{\mathrm{GRAV}}\right)-\nabla\cdot\boldsymbol{\sigma}=0 \tag{1}$$

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

$$\rho C_p\left(\frac{\partial T}{\partial t}+\mathbf{u}\cdot\nabla T\right)-\nabla\cdot(\kappa\nabla T)=0 \tag{3}$$

Here, ρ and $\mathbf{u}$ are the density and velocity, β_{T} is the coefficient of thermal expansion, T is the temperature, T_{ref} is a reference temperature, $\mathbf{a}_{\mathrm{GRAV}}$ is the gravitational acceleration, C_p is the constant-pressure specific heat and κ is the thermal conductivity. The stress tensor $\boldsymbol{\sigma}$ is defined as $\boldsymbol{\sigma}(p,\mathbf{u})=-p\mathbf{I}+2\mu\boldsymbol{\varepsilon}(\mathbf{u})$, where p is the pressure, $\mathbf{I}$ is the identity tensor, $\mu=\rho\nu$ is the viscosity, ν is the kinematic viscosity and $\boldsymbol{\varepsilon}(\mathbf{u})=((\nabla\mathbf{u})+(\nabla\mathbf{u})^{\mathrm{T}})/2$ is the strain-rate tensor. For ideal gases, $\beta_{\mathrm{T}}=1/T$, with T in the expression representing the absolute temperature. As it was pointed out in [21], this expression is not valid for water, and the β_{T} values need to be extracted from tabulated data. In computations, it was proposed in [21] to use a polynomial representation of that data, expressed as a function of temperature, for the expected temperature range $T_1\leqslant T\leqslant T_2$. A simple way, proposed in [21], would be to use a quadratic polynomial:

$$\beta_{\mathrm{T}}(T)=(\beta_{\mathrm{T}})_1+b_1(T-T_1)+b_2(T-T_1)^2 \tag{4}$$

where $(\beta_{\mathrm{T}})_1=\beta_{\mathrm{T}}(T_1)$, and the coefficients b_1 and b_2 are determined by a least-squares fit to the data tabulated for the range $T_1\leqslant T\leqslant T_2$. The essential and natural boundary conditions for Equation (1) are represented as $\mathbf{u}=\mathbf{g}$ on $(\Gamma_t)_g$ and $\mathbf{n}\cdot\boldsymbol{\sigma}=\mathbf{h}$ on $(\Gamma_t)_{\mathrm{h}}$, where $(\Gamma_t)_g$ and $(\Gamma_t)_{\mathrm{h}}$ 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. The essential and natural boundary conditions for Equation (3) are represented as $T=g_{\mathrm{T}}$ on $((\Gamma_t)_g)_{\mathrm{T}}$ and

DOI: 10.1002/fld

$\mathbf{n}\cdot\kappa\nabla T = h_{\mathrm{T}}$ on $((\Gamma_t)_{\mathrm{h}})_{\mathrm{T}}$, where $((\Gamma_t)_g)_{\mathrm{T}}$ and $((\Gamma_t)_{\mathrm{h}})_{\mathrm{T}}$ are complementary subsets of the boundary Γ_t, and g_{T} and h_{T} are given functions.

3. FINITE ELEMENT FORMULATIONS

The finite element trial function spaces $\mathscr{S}_{\mathbf{u}}^h$ for velocity and $\mathscr{S}_p^h$ for pressure, and the test function spaces $\mathscr{V}_{\mathbf{u}}^h$ and $\mathscr{V}_p^h = \mathscr{S}_p^h$ are defined by using, over Ω_t, first-order polynomials. Similarly, the trial and test functions spaces $\mathscr{S}_{\mathrm{T}}^h$ and $\mathscr{V}_{\mathrm{T}}^h$ for temperature are defined by using first-order polynomials. Although the test computations reported in this paper do not involve moving boundaries or interfaces, for generality we express the stabilized formulations in the context of the ALE method. The formulation corresponding to Equations (1) and (2) is expressed, from [8, 16, 21], as follows: find $\mathbf{u}^h \in \mathscr{S}_{\mathbf{u}}^h$ and $p^h \in \mathscr{S}_p^h$ such that $\forall \mathbf{w}^h \in \mathscr{V}_{\mathbf{u}}^h$ and $\forall q^h \in \mathscr{V}_p^h$:

$$\begin{aligned}
&\int_{\Omega_t} \mathbf{w}^h \cdot \rho \left(\left.\frac{\partial \mathbf{u}^h}{\partial t}\right|_{\boldsymbol{\xi}} + (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla\mathbf{u}^h - (1-\beta_{\mathrm{T}}(T^h - T_{\mathrm{ref}}))\mathbf{a}_{\mathrm{GRAV}} \right) \mathrm{d}\Omega \\
&\quad + \int_{\Omega_t} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h)\, \mathrm{d}\Omega - \int_{(\Gamma_t)_{\mathrm{h}}} \mathbf{w}^h \cdot \mathbf{h}^h \, \mathrm{d}\Gamma + \int_{\Omega_t} q^h \nabla\cdot\mathbf{u}^h \, \mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{n_{el}} \int_{\Omega_t^e} \frac{1}{\rho} [\tau_{\mathrm{SUPG}} \rho (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla\mathbf{w}^h + \tau_{\mathrm{PSPG}} \nabla q^h] \\
&\quad \cdot [Ł(p^h, \mathbf{u}^h) - \rho(1-\beta_{\mathrm{T}}(T^h - T_{\mathrm{ref}}))\mathbf{a}_{\mathrm{GRAV}}]\, \mathrm{d}\Omega \\
&\quad + \sum_{e=1}^{n_{el}} \int_{\Omega_t^e} \nu_{\mathrm{LSIC}} \nabla\cdot\mathbf{w}^h \rho \nabla\cdot\mathbf{u}^h \mathrm{d}\Omega + S_{\mathrm{DC}} = 0
\end{aligned} \tag{5}$$

where $\boldsymbol{\xi}$ is the vector of element (parent-domain) coordinates, $\mathbf{v}^h$ is the mesh velocity, and

$$Ł(q^h, \mathbf{w}^h) = \rho \left(\left.\frac{\partial \mathbf{w}^h}{\partial t}\right|_{\boldsymbol{\xi}} + (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla\mathbf{w}^h \right) - \nabla\cdot\boldsymbol{\sigma}(q^h, \mathbf{w}^h) \tag{6}$$

Here, τ_{SUPG}, τ_{PSPG} and ν_{LSIC} are the SUPG, PSPG and LSIC stabilization parameters. We provide in Section 4 the definitions used for the computations reported in this paper. The symbol S_{DC} represents the DC term, which we find helpful in computations with thermal coupling, especially for the equation governing the temperature. For the Navier–Stokes equations, the DC term is given from [16, 17, 23] as

$$S_{\mathrm{DC}} = \sum_{e=1}^{n_{el}} \int_{\Omega_t^e} \rho \nabla\mathbf{w}^h : (\boldsymbol{\nu}_{\mathrm{DC}} \cdot \nabla\mathbf{u}^h)\, \mathrm{d}\Omega \tag{7}$$

where $\boldsymbol{\nu}_{\mathrm{DC}}$ is the DC parameter. For examples of ways of calculating this parameter, including the DCDD parameter, see [16, 17, 23]. We note that the DCDD stabilization was originally introduced as an alternative to the LSIC stabilization; therefore, normally only one of these stabilizations would be retained. How we calculate $\boldsymbol{\nu}_{\mathrm{DC}}$ is described in Section 5.

DOI: 10.1002/fld

The stabilized formulation corresponding to Equation (3) is expressed from [5, 6, 21] as follows: find $T^h \in \mathscr{S}^h_{\mathrm{T}}$ such that $\forall w^h \in \mathscr{V}^h_{\mathrm{T}}$:

$$\int_{\Omega_t} w^h \rho C_p \left(\left.\frac{\partial T^h}{\partial t}\right|_{\xi} + (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla T^h \right) \mathrm{d}\Omega + \int_{\Omega_t} \nabla w^h \cdot \kappa \nabla T^h \, \mathrm{d}\Omega$$
$$- \int_{((\Gamma_t)_{\mathrm{h}})_{\mathrm{T}}} w^h \mathrm{h}^h \, \mathrm{d}\Gamma + \sum_{e=1}^{n_{el}} \int_{\Omega_t^e} (\tau_{\mathrm{SUPG}})_{\mathrm{T}} (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla w^h$$
$$\times \left(\rho C_p \left(\left.\frac{\partial T^h}{\partial t}\right|_{\xi} + (\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla T^h \right) - \nabla\cdot(\kappa \nabla T^h) \right) \mathrm{d}\Omega + (S_{\mathrm{DC}})_{\mathrm{T}} = 0 \tag{8}$$

where $(\tau_{\mathrm{SUPG}})_{\mathrm{T}}$ is the SUPG stabilization parameter, and from [5, 6]:

$$(S_{\mathrm{DC}})_{\mathrm{T}} = \sum_{e=1}^{n_{el}} \int_{\Omega_t^e} \nabla w^h \cdot \boldsymbol{\kappa}_{\mathrm{DC}} \nabla T^h \, \mathrm{d}\Omega \tag{9}$$

For early examples of ways of calculating $\boldsymbol{\kappa}_{\mathrm{DC}}$, see [5, 6]. Newer examples of ways of calculating $\boldsymbol{\kappa}_{\mathrm{DC}}$ are those based on the DCDD stabilization [16, 17, 23] and '$YZ\beta$ shock-capturing' [17, 23] techniques (the 'β' in $YZ\beta$ shock-capturing is not related to the β_{T} representing the coefficient of thermal expansion). How we calculate $\boldsymbol{\kappa}_{\mathrm{DC}}$ is described in Section 5.

4. STABILIZATION PARAMETERS

For the Navier–Stokes equations, we provide the stabilization parameters defined in [16, 21]:

$$\tau_{\mathrm{SUPG}} = \left(\frac{1}{\tau_{\mathrm{SUGN1}}^2} + \frac{1}{\tau_{\mathrm{SUGN2}}^2} + \frac{1}{\tau_{\mathrm{SUGN3}}^2} \right)^{-1/2} \tag{10}$$

$$\tau_{\mathrm{SUGN1}} = \left(\sum_{a=1}^{n_{en}} |(\mathbf{u}^h - \mathbf{v}^h)\cdot\nabla N_a| \right)^{-1}, \quad \tau_{\mathrm{SUGN2}} = \frac{\Delta t}{2}, \quad \tau_{\mathrm{SUGN3}} = \frac{h_{\mathrm{RGN}}^2}{4\nu} \tag{11}$$

$$h_{\mathrm{RGN}} = 2\left(\sum_{a=1}^{n_{en}} |\mathbf{r}\cdot\nabla N_a| \right)^{-1}, \quad \mathbf{r} = \frac{\nabla\|\mathbf{u}^h\|}{\|\nabla\|\mathbf{u}^h\|\|} \tag{12}$$

$$\tau_{\mathrm{PSPG}} = \tau_{\mathrm{SUPG}} \tag{13}$$

$$\nu_{\mathrm{LSIC}} = \tau_{\mathrm{SUPG}} \|\mathbf{u}^h - \mathbf{v}^h\|^2 \tag{14}$$

where Δt is the time-step size, n_{en} is the number of element nodes and N_a is the shape function associated with node a. We note again that although the test computations reported in this paper do not involve moving boundaries or interfaces, for generality we provide the definitions of the stabilization parameters in the context of the ALE method. For more ways of calculating τ_{SUPG}, τ_{PSPG} and ν_{LSIC}, see [12, 16, 17, 23].

DOI: 10.1002/fld

For the temperature equation, we provide the stabilization parameters defined in [21]:

$$(\tau_{\mathrm{SUPG}})_{\mathrm{T}}=\left(\frac{1}{((\tau_{\mathrm{SUGN1}})_{\mathrm{T}})^2}+\frac{1}{((\tau_{\mathrm{SUGN2}})_{\mathrm{T}})^2}+\frac{1}{((\tau_{\mathrm{SUGN3}})_{\mathrm{T}})^2}\right)^{-1/2} \tag{15}$$

$$(\tau_{\mathrm{SUGN1}})_{\mathrm{T}}=\tau_{\mathrm{SUGN1}},\quad (\tau_{\mathrm{SUGN2}})_{\mathrm{T}}=\tau_{\mathrm{SUGN2}},\quad (\tau_{\mathrm{SUGN3}})_{\mathrm{T}}=\frac{((h_{\mathrm{RGN}})_{\mathrm{T}})^2}{4(\kappa/(\rho C_p))} \tag{16}$$

$$(h_{\mathrm{RGN}})_{\mathrm{T}}=2\left(\sum_{a=1}^{n_{en}}|\mathbf{r}_{\mathrm{T}}\cdot\nabla N_a|\right)^{-1},\quad \mathbf{r}_{\mathrm{T}}=\frac{\nabla T^h}{\|\nabla T^h\|} \tag{17}$$

5. DC PARAMETERS

For the Navier–Stokes equations, we can define $\boldsymbol{\nu}_{\mathrm{DC}}$ as $\boldsymbol{\nu}_{\mathrm{DC}}=\nu_{\mathrm{DC}}\mathbf{I}$, $\boldsymbol{\nu}_{\mathrm{DC}}=\nu_{\mathrm{DC}}\mathbf{rr}$, $\boldsymbol{\nu}_{\mathrm{DC}}=\nu_{\mathrm{DC}}[\mathbf{rr}-(\mathbf{r}\cdot\mathbf{s})^2\mathbf{ss}]$ (where $\mathbf{s}=\mathbf{u}/\|\mathbf{u}\|$), or in some more complex way, such as by using '*switch*' functions, as described in [17, 23]. Based on the DCDD stabilization, ν_{DC} can be calculated by using the expression:

$$\nu_{\mathrm{DC}}=\nu_{\mathrm{DCDD}}=\frac{1}{2}\left(\frac{\|\mathbf{u}^h-\mathbf{v}^h\|}{u_{\mathrm{ref}}}\right)^2(h_{\mathrm{RGN}})^2\|\nabla\|\mathbf{u}^h\|\| \tag{18}$$

where 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 $\|\mathbf{u}^h\|$).

For the temperature equation, we can define $\boldsymbol{\kappa}_{\mathrm{DC}}$ as $\boldsymbol{\kappa}_{\mathrm{DC}}=\rho C_p(\nu_{\mathrm{DC}})_{\mathrm{T}}\mathbf{I}$, $\boldsymbol{\kappa}_{\mathrm{DC}}=\rho C_p(\nu_{\mathrm{DC}})_{\mathrm{T}}\mathbf{r}_{\mathrm{T}}\mathbf{r}_{\mathrm{T}}$, or in some more complex way, such as by using '*switch*' functions, as described in [17, 23]. Based on $YZ\beta$ shock-capturing (with $\beta=2$), $(\nu_{\mathrm{DC}})_{\mathrm{T}}$ can be calculated, as proposed in [21], by using the following expression:

$$(\nu_{\mathrm{DC}})_{\mathrm{T}}=(\nu_{YZ\beta})_{\mathrm{T}}=\left|Y^{-1}\left(\left.\frac{\partial T^h}{\partial t}\right|_{\xi}+(\mathbf{u}^h-\mathbf{v}^h)\cdot\nabla T^h\right)\right|\left(\frac{(h_{\mathrm{RGN}})_{\mathrm{T}}}{2}\right)^2 \tag{19}$$

where Y is a scaling value for T, which can be selected as $Y=T_{\max}-T_{\min}$.

Based on the DCDD stabilization, $(\nu_{\mathrm{DC}})_{\mathrm{T}}$ can be calculated by using one of the following expressions:

$$(\nu_{\mathrm{DC}})_{\mathrm{T}}=((\nu_{\mathrm{DCDD}})_{\mathrm{T}})_1=\|\mathbf{u}^h-\mathbf{v}^h\|\left(\frac{(h_{\mathrm{RGN}})_{\mathrm{T}}}{2}\right)\frac{\|\nabla T^h\|(h_{\mathrm{RGN}})_{\mathrm{T}}}{(\Delta T)_{\mathrm{ref}}} \tag{20}$$

$$(\nu_{\mathrm{DC}})_{\mathrm{T}}=((\nu_{\mathrm{DCDD}})_{\mathrm{T}})_2=\frac{\|\mathbf{u}^h-\mathbf{v}^h\|^2}{u_{\mathrm{ref}}}\left(\frac{(h_{\mathrm{RGN}})_{\mathrm{T}}}{2}\right)\frac{\|\nabla T^h\|(h_{\mathrm{RGN}})_{\mathrm{T}}}{(\Delta T)_{\mathrm{ref}}} \tag{21}$$

where $(\Delta T)_{\mathrm{ref}}$ is a reference value for ΔT, which can be selected as $(\Delta T)_{\mathrm{ref}}=Y$. The expression given by Equation (20) was proposed in [21].

We note that the DC expressions given by Equations (19)–(21) are all quadratic in h_{RGN} and $(h_{\mathrm{RGN}})_{\mathrm{T}}$. Using expressions that are linear in those local length scales would lead to more dissipative DC stabilization and that might be preferable in some cases. For example, it was shown

DOI: 10.1002/fld

in [24] that $YZ\beta$ shock-capturing with $\beta=1$ (leading to a DC expression that is linear in h_{RGN}) was the best compromise between accuracy and robustness for drug delivery problems.

6. TEST COMPUTATIONS

All computations are carried out in a parallel computing environment, using PC clusters. In all cases, the fully discretized, coupled fluid mechanics and temperature equations are solved with a direct coupling (see [21] for terminology). In solving the linear equation system at every nonlinear iteration, the GMRES search technique [25] is used with a diagonal preconditioner. The LSIC term is dropped, and the stabilization parameters are based on Equations (10)–(13) and (15)–(17). Unless stated otherwise, the τ_{SUGN2} and $(\tau_{\mathrm{SUGN2}})_{\mathrm{T}}$ terms are dropped in Equations (10) and (15). The DC terms are included only when it is explicitly stated so, with the definitions $\mathbf{v}_{\mathrm{DC}}=\nu_{\mathrm{DC}}\mathbf{I}$ and $\boldsymbol{\kappa}_{\mathrm{DC}}=\rho C_p(\nu_{\mathrm{DC}})_{\mathrm{T}}\mathbf{I}$ and with the DC parameters calculated based on Equations (18) and (20). In time integrations, all terms involving β_{T} are treated implicitly. In natural convection, the Rayleigh number, based on a characteristic length d, is used for parameterizing the flow. It is defined as

$$Ra=\frac{\|\mathbf{a}_{\mathrm{GRAV}}\|\beta_m(T_{\mathrm{h}}-T_{\mathrm{c}})d^3}{\nu\alpha} \tag{22}$$

Here, α is the thermal diffusivity, T_{h} and T_{c} are the hot and cold temperature boundary conditions, respectively, β_m is a reference value for β_{T}, set as $\beta_m=2/(T_{\mathrm{h}}+T_{\mathrm{c}})$, and the magnitude of the gravitational acceleration is set to $9.81\,\mathrm{m/s^2}$. In all computations the fluid is air, with $\nu=1.5\times 10^{-5}\,\mathrm{m^2/s}$, and the Prandtl number, defined as $Pr=\nu/\alpha$, is set to 0.71. Computations are carried out with both the actual (spatially and temporally varying) thermal-expansion coefficient β_{T} and the reference thermal-expansion coefficient β_m, and the results are compared. The computed data we report include Nu_{h}, which is the surface-averaged value of the Nusselt number at the hot wall. The Nusselt number is defined here as $Nu=(\mathbf{n}\cdot\nabla T)d/(T_{\mathrm{h}}-T_{\mathrm{c}})$.

6.1. Natural convection in a 2D square cavity

The problem setup is shown in Figure 1. The two meshes used are shown in Figures 2 and 3. Mesh-1 has 3721 nodes and 3600 four-node quadrilateral elements and is used for $Ra=10^3$, 10^4 and 10^5. Mesh-2 has 10 201 nodes and 10 000 four-node quadrilateral elements and is used for

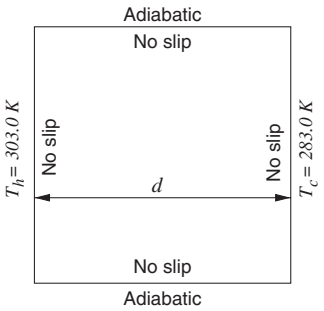

Figure 1. Natural convection in a 2D square cavity. Problem setup.

DOI: 10.1002/fld

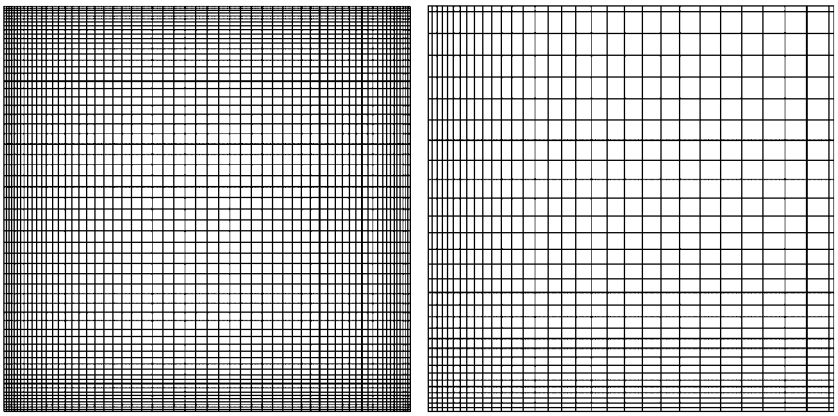

Figure 2. Natural convection in a 2D square cavity. Mesh-1, used for $Ra = 10^3$, 10^4 and 10^5, has 3721 nodes and 3600 four-node quadrilateral elements.

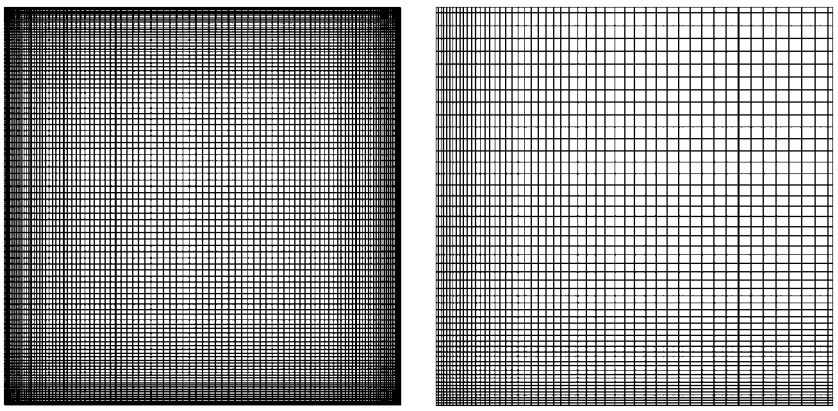

Figure 3. Natural convection in a 2D square cavity. Mesh-2, used for $Ra = 10^6$, 10^7 and 10^8, has 10 201 nodes and 10 000 four-node quadrilateral elements.

$Ra = 10^6$, 10^7 and 10^8. The time-step size is 0.003 s. The number of nonlinear iterations per time step is 3, and the number of inner GMRES iterations per nonlinear iteration is 50.

In Table I, the Nu_h values obtained with the actual thermal-expansion coefficient β_T and reference thermal-expansion coefficient β_m are compared with the Nu_h values reported in [26, 27]. The Nu_h values obtained with β_m closely match the values reported in [26, 27], whereas the values obtained with β_T are slightly lower. This suggests that it might be a common practice to treat β_T as a constant. Such a treatment introduces some error in the buoyancy-term evaluation, which explains the difference between the Nu_h values obtained with β_m and β_T. With β_m, the buoyancy term in Equation (1) becomes a linear function of the temperature, when in reality the temperature dependence is nonlinear. Furthermore, the treatment of β_T as a constant leads to a range of somewhat arbitrary choices for that constant. Therefore, we believe that the actual value of β_T should be used in computations and not its reference value. Temperature contours obtained with the actual value of β_T are shown in Figures 4 and 5.

DOI: 10.1002/fld

Table I. Natural convection in a 2D square cavity.

	Nu_h		
Ra	β_m	β_T	References [26, 27]
10^3	1.118	1.112	1.118
10^4	2.243	2.220	2.243
10^5	4.515	4.472	4.519
10^6	8.816	8.742	8.799
10^7	16.52	16.38	16.52
10^8	30.14	29.91	30.23

The Nu_h values obtained with the actual thermal-expansion coefficient β_T and reference thermal-expansion coefficient β_m are compared with the Nu_h values reported in [26, 27].

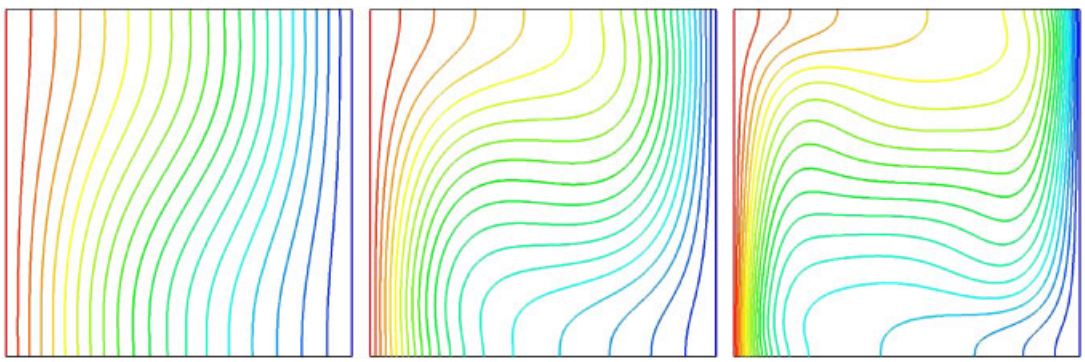

Figure 4. Natural convection in a 2D square cavity. Temperature contours obtained for $Ra = 10^3$ (left), 10^4 (middle) and 10^5 (right).

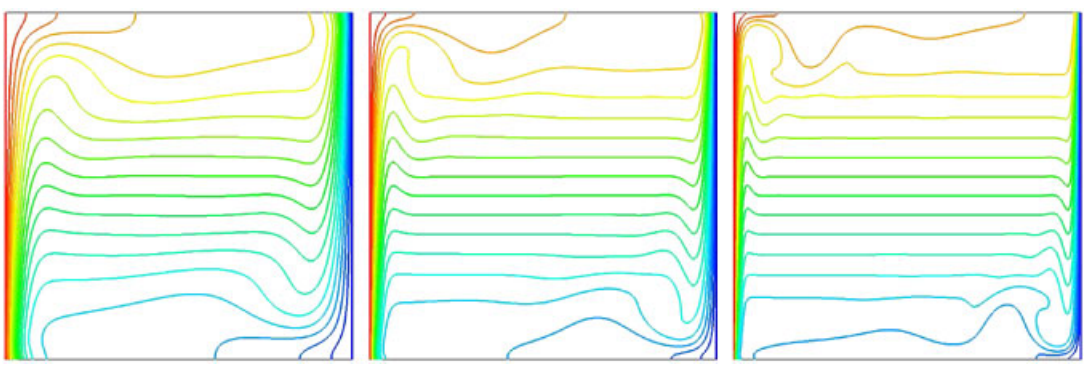

Figure 5. Natural convection in a 2D square cavity. Temperature contours obtained for $Ra = 10^6$ (left), 10^7 (middle) and 10^8 (right).

DOI: 10.1002/fld

The results presented in Table I are obtained without DC stabilization. The effect of using DC is studied for $Ra = 10^6$ and 10^8. The Nu_h values obtained with different combinations of active and inactive ν_{DC} and $(\nu_{DC})_T$ are shown in Table II. We observe that the Nu_h values are rather close. Temperature profiles near the hot and cold walls obtained for $Ra = 10^8$ with different combinations of active and inactive ν_{DC} and $(\nu_{DC})_T$ are shown in Figure 6. As expected, including a DC term for the temperature equation results in a slight decrease in the steepness of the temperature

Table II. Natural convection in a 2D square cavity.

$Ra = 10^6$			$Ra = 10^8$		
ν_{DC}	$(\nu_{DC})_T$	Nu_h	ν_{DC}	$(\nu_{DC})_T$	Nu_h
Off	Off	8.742	Off	Off	29.91
On	Off	8.731	On	Off	29.75
Off	On	8.790	Off	On	30.73
On	On	8.779	On	On	30.52

The Nu_h values obtained with different combinations of active and inactive ν_{DC} and $(\nu_{DC})_T$.

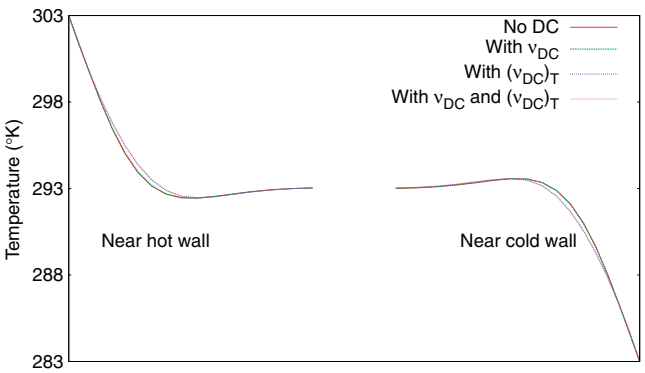

Figure 6. Natural convection in a 2D square cavity. Temperature profiles near the hot and cold walls obtained for $Ra = 10^8$ with different combinations of active and inactive ν_{DC} and $(\nu_{DC})_T$.

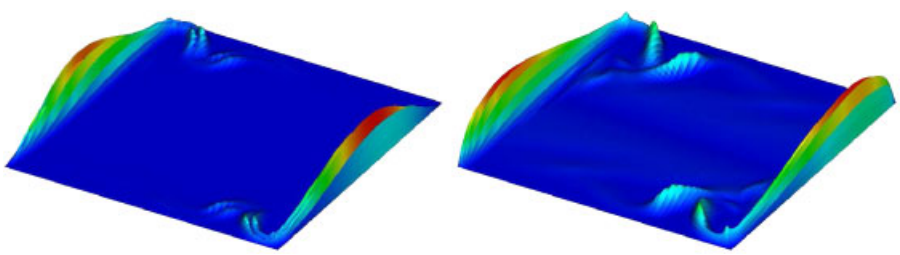

Figure 7. Natural convection in a 2D square cavity. Elevation plots for ν_{DC} (left) and $(\nu_{DC})_T$ (right) for $Ra = 10^8$.

DOI: 10.1002/fld

profile in the boundary layer. However, the differences are rather small, and we believe that this is because the mesh resolution is rather high where we have these temperature layers. Figure 7 shows the elevation plots for the DC parameters for $Ra=10^8$. As expected, the values are higher near the walls.

6.2. Natural convection in a 2D square cavity with an enclosed square body

The problem setup is shown in Figure 8. Computations are carried out for $Ra=10^4$ and 10^6. The two meshes used are shown in Figures 9 and 10. Mesh-1 has 3360 nodes and 3200 four-node quadrilateral elements and is used for $Ra=10^4$. Mesh-2 has 13 120 nodes and 12 800 four-node quadrilateral elements and is used for $Ra=10^6$. The time-step size is 0.008 s, the number of nonlinear iterations per time step is 3 and the number of inner GMRES iterations per nonlinear iteration is 50 for $Ra=10^4$ and 30 for $Ra=10^6$. The various boundary conditions imposed on the enclosed body are described in a dimensionless form with the expression $\theta_b=(T_b-T_c)/(T_h-T_c)$, where T_b is the temperature prescribed on the enclosed body.

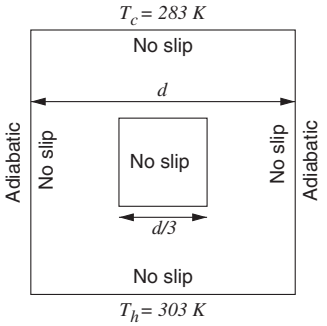

Figure 8. Natural convection in a 2D square cavity with an enclosed square body: problem setup.

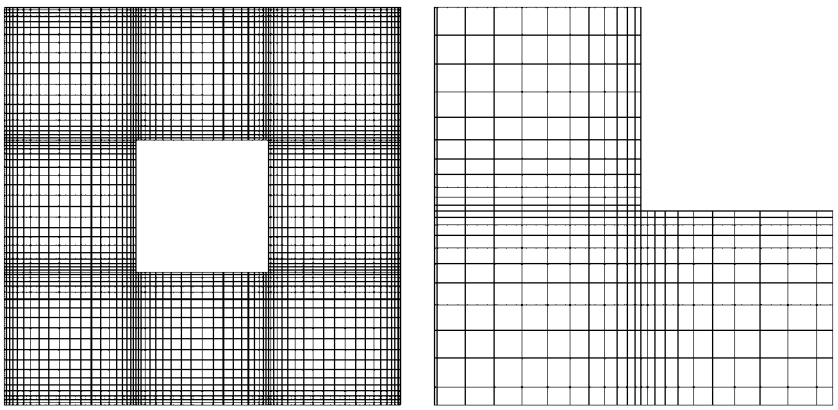

Figure 9. Natural convection in a 2D square cavity with an enclosed square body. Mesh-1, used for $Ra=10^4$, has 3360 nodes and 3200 four-node quadrilateral elements.

DOI: 10.1002/fld

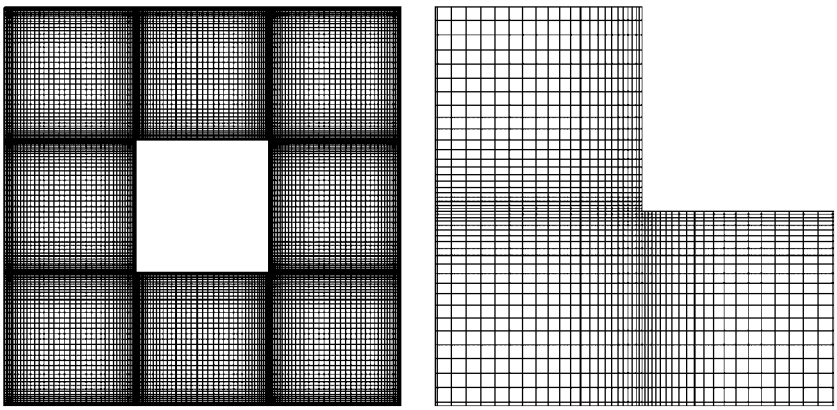

Figure 10. Natural convection in a 2D square cavity with an enclosed square body. Mesh-2, used for $Ra = 10^6$, has 13 120 nodes and 12 800 four-node quadrilateral elements.

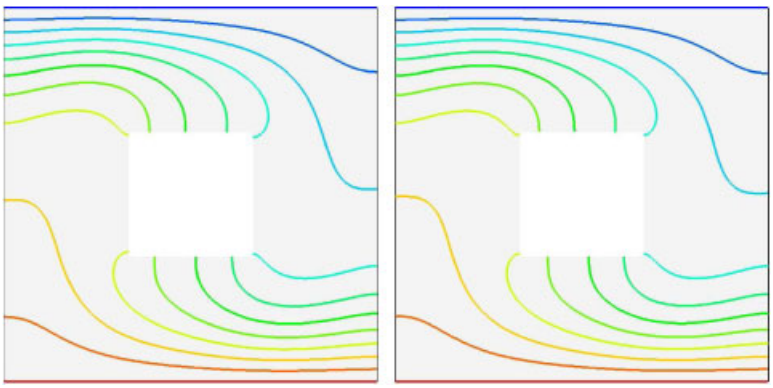

Figure 11. Natural convection in a 2D square cavity with an enclosed square body. $Ra = 10^4$, adiabatic body. Temperature contours obtained with β_m (left) and β_T (right).

6.2.1. $Ra = 10^4$, adiabatic body. We use this setup to compare the results obtained with β_m and β_T. The Nu_h values obtained with β_m and β_T are 2.335 and 2.308, respectively. These values are close to those reported in [28]. Figure 11 shows the temperature contours obtained with β_m and β_T. They are in good agreement with those shown in [28].

6.2.2. $Ra = 10^4$, isothermal body. Computations are carried out for the non-dimensional body-temperature values of $\theta_b = 0.0$, 0.5 and 1.0. These cases are computed with β_T. Figure 12 shows the temperatures contours obtained for $\theta_b = 0.0$, 0.5 and 1.0.

6.2.3. $Ra = 10^6$, isothermal body. At $Ra = 10^6$ the convection is unsteady. Computations are carried out for $\theta_b = 1.0$, with β_m and β_T. The time-averaged temperature contours are shown in Figure 13. Time histories of Nu_h obtained with β_m and β_T are shown in Figure 14. The time histories are similar and compare well with those reported in [28].

DOI: 10.1002/fld

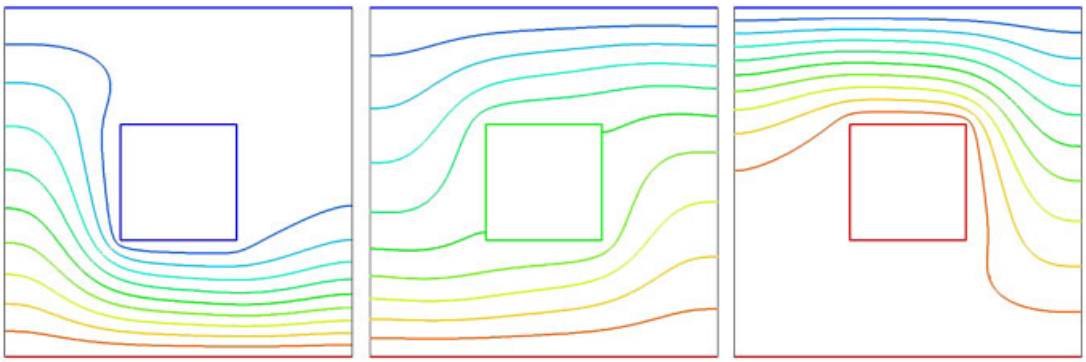

Figure 12. Natural convection in a 2D square cavity with an enclosed square body. $Ra = 10^4$, isothermal body. Temperature contours obtained for $\theta_b = 0.0$ (left), 0.5 (middle) and 1.0 (right).

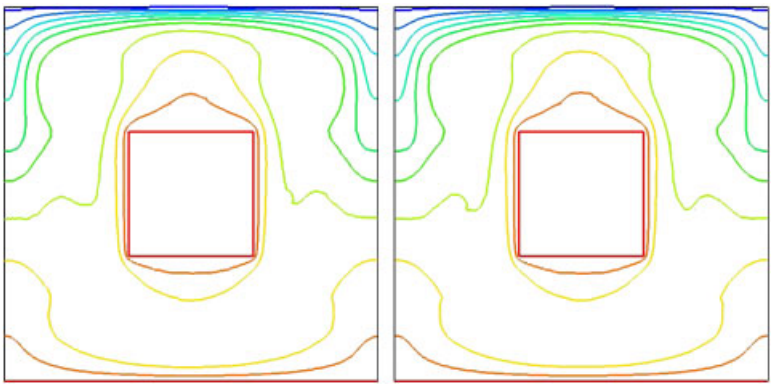

Figure 13. Natural convection in a 2D square cavity with an enclosed square body. $Ra = 10^6$, isothermal body. Time-averaged temperature contours obtained for $\theta_b = 1.0$ with β_m (left) and β_T (right).

6.3. Natural convection in a 3D cubical cavity

The problem setup for the 3D case is an extension of the 2D case shown in Figure 1. The boundary conditions imposed on the two additional faces are no-slip conditions for the fluid mechanics equations and adiabatic condition for the temperature equation. The mesh has 29 791 nodes and 27 000 eight-node hexahedral elements and is shown in Figure 15. The time-step size is 0.005 s. The number of nonlinear iterations per time step is 3, and the number of inner GMRES iterations per nonlinear iteration is 50. The temperature distribution and contours for $Ra = 10^6$ at steady state are shown in Figure 16. The 3D effects are most prominent in the vicinity of the two added faces. Figure 16 also shows the temperature contours on the mid-plane perpendicular to the axis along the added dimension. These contours, as expected, are similar to those seen in the 2D case.

DOI: 10.1002/fld

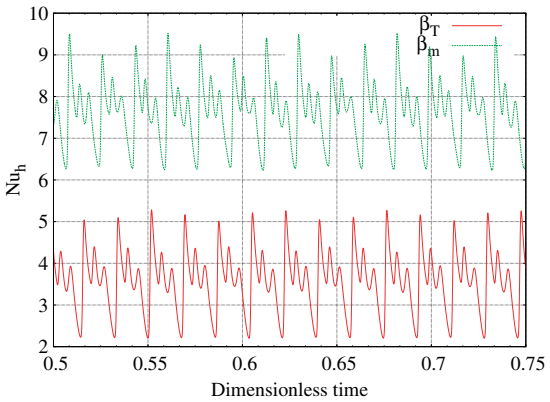

Figure 14. Natural convection in a 2D square cavity with an enclosed square body. $Ra = 10^6$, isothermal body. Time histories of Nu_h obtained with β_m and β_T. The dimensionless time used in the plots is defined as $t^* = t\alpha/d^2$. The curve for the Nu_h obtained with β_m has been shifted up by 4.0 for the purpose of illustration.

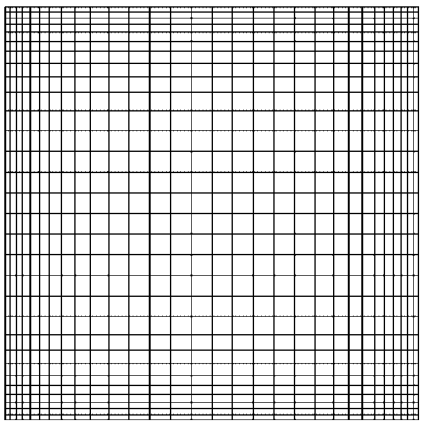

Figure 15. Natural convection in a 3D cubical cavity. Two-dimensional view of the mesh, which has 29 791 nodes and 27 000 eight-node hexahedral elements.

6.4. Simplified model of air circulation and cooling in a small data center

The room is 5.3 m long, 4.2 m wide and 3.05 m high. The room houses two computer racks, each 2.2 m long, 1.0 m wide and 1.6 m high (see Figure 17). The cold air is pumped into the room from a rectangular floor inlet with dimensions $1.0\,\text{m} \times 2.0\,\text{m}$. The inlet is located between the two racks. The air is vented out of the room through four ceiling vents, each with dimensions $1.0\,\text{m} \times 0.5\,\text{m}$. There are various types of boundary conditions specified in this problem (see Figure 17). For the inlet on the floor, we prescribe the flow velocity to be in the normal direction, with magnitude 1.0 m/s, and the temperature to be 288 K. For the sides of the racks facing the floor inlet, appropriate

DOI: 10.1002/fld

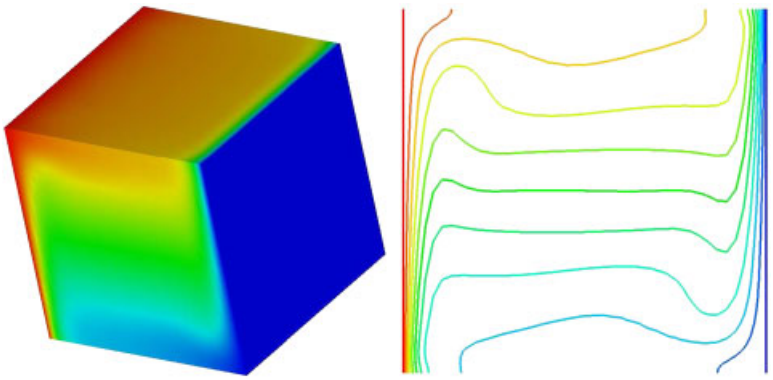

Figure 16. Natural convection in a 3D cubical cavity. Temperature distribution on the boundaries of the 3D domain (left) and temperature contours on the mid-plane perpendicular to the axis along the added dimension (right).

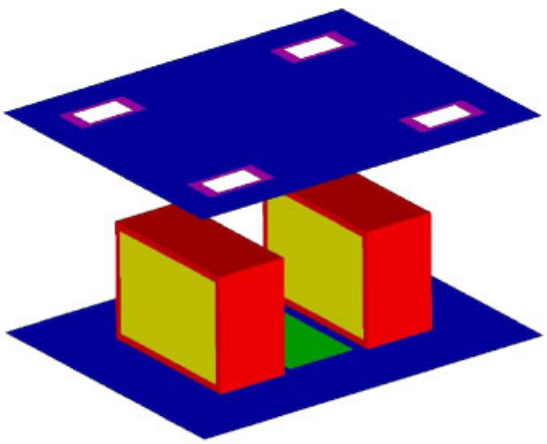

Figure 17. Simplified model of air circulation and cooling in a small data center. A partial view of the model surfaces: floor inlet (green), outer zones of the ceiling vents (magenta), rack (red), rack inflow and outflow (yellow) and floor and ceiling (blue). The four enclosing walls and the inner zones of the ceiling vents are not shown. The volume mesh has 361 440 nodes and 383 688 eight-node hexahedral elements.

boundary conditions would be the fluid stresses representing the suction generated by the cooling fans in the racks. Instead, for simplicity, we specify a velocity of 0.25 m/s in the normal direction, and zero stress in the other two directions. For the sides of the racks facing away from the inlet, we specify the velocity to be in the normal direction, with magnitude 0.25 m/s and the temperature to be 12 K above the mean temperature over the sides facing the floor inlets. This temperature differential is based on a heat generation rate of 400 W per processor, with 12 processors per unit length of the rack. For the ceiling vents, we partition each vent area into two zones: a rectangular inner zone and the outer zone. In the inner zone, we specify the stresses to be zero. In the outer zone, we specify the velocity in the normal direction and zero stress in the other two directions.

DOI: 10.1002/fld

This combination of boundary conditions ensures proper mass balance and prevents any reverse flow from the vents. At all other boundaries, we specify no-slip conditions for the fluid mechanics equations and adiabatic condition for the temperature equation. The volume mesh used in the computation has 361 440 nodes and 383 688 eight-node hexahedral elements. The time-step size is 0.0045 s. The number of nonlinear iterations per time step is 3, and the number of inner GMRES iterations per nonlinear iteration is 30. Unlike the test cases in the preceding sections, the highly transient nature of this problem requires τ_{SUGN2} and $(\tau_{SUGN2})_T$ to be active. Furthermore, to prevent possible oscillations in regions of high gradients that may be relatively under-resolved, we include the DC terms.

Figure 18 shows the temperature distribution on three mutually perpendicular planes in the room. On the vertical plane cutting through the racks, we see, as expected, hot regions in the vicinity of the hot faces of the racks. We also see that the rising air creates pockets of hot air above the racks. Table III provides the average temperature values at various locations in the room. Figure 19 shows the velocity vectors on half of the vertical plane bisecting the racks.

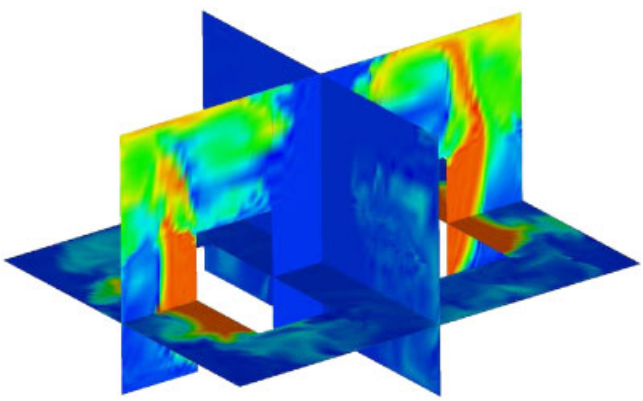

Figure 18. Simplified model of air circulation and cooling in a small data center. Temperature distribution on three mutually perpendicular planes in the room.

Table III. Simplified model of air circulation and cooling in a small data center. Average temperature at various locations in the room.

Location	Average temperature (K)
Room	290.9
Ceiling	293.8
Floor	288.1
Ceiling vents	293.7
Horizontal plane bisecting the racks	289.9

DOI: 10.1002/fld

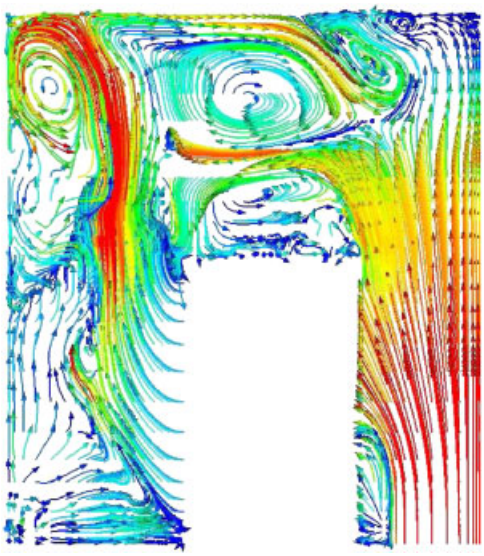

Figure 19. Simplified model of air circulation and cooling in a small data center. Velocity vectors on half of the vertical plane bisecting the racks.

7. CONCLUDING REMARKS

We presented applications of the stabilized finite element formulations developed earlier for incompressible flows with thermal coupling to 2D and 3D test problems. The formulations are based on the SUPG and PSPG stabilizations and are supplemented with DC. We also described the stabilization and DC parameters to be used with these formulations. Although the fluid mechanics and temperature equations have a common advective length scale, they have their individual diffusive length scales. We provided extensions of the DCDD to the fluid mechanics and temperature equations. For computational robustness, we solve the coupled fluid mechanics and temperature equations with a direct coupling technique. The test problems computed included a number of 2D and 3D natural convection problems and a simplified 3D model of air circulation and cooling in a small data center. In some of the cases, we were able to compare our results to those reported in the literature. The favorable nature of the comparisons in those cases and the reasonable nature of the results in the other cases increased our confidence in and demonstrated a good potential for the formulations developed.

APPENDIX A: QUADRATIC REPRESENTATION OF β_T

For ideal gases, $\beta_T = 1/T$, with T in the expression representing the absolute temperature. As it was pointed out in [21], this expression is not valid for water, and the β_T values need to be extracted from tabulated data. Table AI provides the values of β_T for water at various temperatures. As mentioned in Section 2, it was proposed in [21] to use in computations a polynomial representation of such data, expressed as a function of temperature, for the expected temperature range $T_1 \leqslant T \leqslant T_2$.

DOI: 10.1002/fld

Table AI. Values of β_T for water at various temperatures.

Temperature (°C)	β_T (K^{-1})
0	−8.530e−05
5	5.200e−06
10	8.211e−05
15	1.480e−04
20	2.070e−04
25	2.590e−04
30	3.060e−04
35	3.490e−04
40	3.890e−04
45	4.270e−04
50	4.620e−04
55	4.960e−04
60	5.290e−04
65	5.601e−04
70	5.900e−04
75	6.190e−04
80	6.470e−04
85	6.750e−04
90	7.020e−04
95	7.280e−04

Table AII. Curve fitting coefficients b_1 and b_2 for water for various temperature ranges.

Temperature (°C)	b_1 (K^{-2})	b_2 (K^{-3})
0–95	1.446e−05	−6.519e−08
10–95	1.148e−05	−4.755e−08
20–95	9.545e−06	−3.582e−08
30–95	8.273e−06	−2.811e−08
40–95	7.414e−06	−2.318e−08
50–95	6.795e−06	−1.994e−08
60–95	6.232e−06	−1.563e−08
5–75	1.359e−05	−7.156e−08
15–75	1.095e−05	−5.320e−08
25–75	9.153e−06	−3.989e−08
25–65	9.306e−06	−4.530e−08
0–40	1.761e−05	−1.471e−07
0–30	1.827e−05	−1.766e−07
30–60	8.684e−06	−4.237e−08

Also as mentioned in Section 2, a simple way, proposed in [21], would be to use a quadratic polynomial:

$$\beta_T(T) = (\beta_T)_1 + b_1(T - T_1) + b_2(T - T_1)^2 \tag{A1}$$

where $(\beta_T)_1 = \beta_T(T_1)$, and the coefficients b_1 and b_2 are determined by a least-squares fit to the data tabulated for the range $T_1 \leqslant T \leqslant T_2$.

DOI: 10.1002/fld

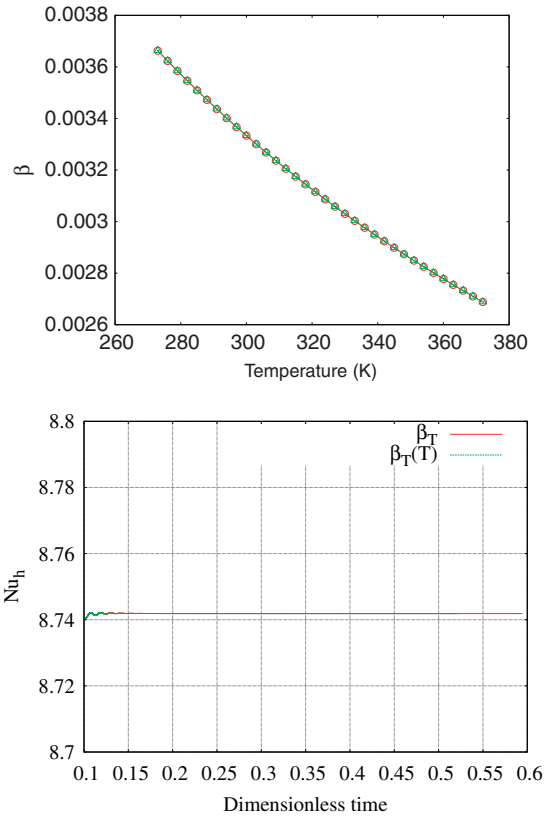

Figure A1. Comparison of the quadratic and $1/T$ representations of β_T. The β_T values (top) and the time histories of the Nu_h values (bottom) for the 2D square cavity at $Ra = 10^6$. In the top frame, the symbol $\circ$ denotes the $1/T$ data, and the symbol $\triangle$ denotes the data obtained from the quadratic representation.

The data from Table AI are used here for determining the values of b_1 and b_2 for various temperature ranges. The results from this curve fitting exercise are tabulated in Table AII. Overall, the quadratic model is able to represent β_T rather well for reasonable lengths ($\leqslant 50$ K) of temperature ranges.

We tested this quadratic-representation concept in modeling β_T for air (perfect gas). The coefficients b_1 and b_2 obtained for air for the temperature range of 0–100°C are -1.295 and 3.196, respectively. As seen in Figure A1, the quadratic and $1/T$ representations of β_T are very close. We used this quadratic representation of β_T in re-computing the case of 2D square cavity at $Ra = 10^6$ (see Section 6.1). As seen in Figure A1, the Nu_h values obtained with the quadratic and $1/T$ representations of β_T are very close.

ACKNOWLEDGEMENTS

This work was supported in part by the Rice Computational Research Cluster funded by NSF under Grant CNS-0421109, and a partnership between Rice University, AMD and Cray.

DOI: 10.1002/fld

REFERENCES

1. Hughes TJR, Brooks AN. A multi-dimensional upwind scheme with no crosswind diffusion. *Finite Element Methods for Convection Dominated Flows*, Hughes TJR (ed.), AMD-vol. 34. ASME: New York, 1979; 19–35.
2. Brooks AN, Hughes TJR. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.
3. Tezduyar TE, Hughes TJR. Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations. *Proceedings of AIAA 21st Aerospace Sciences Meeting*, Reno, Nevada, 1983; AIAA Paper 83-0125.
4. Hughes TJR, Tezduyar TE. Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Computer Methods in Applied Mechanics and Engineering* 1984; **45**:217–284.
5. Hughes TJR, Mallet M, Mizukami A. A new finite element formulation for computational fluid dynamics: II. Beyond SUPG. *Computer Methods in Applied Mechanics and Engineering* 1986; **54**:341–355.
6. Tezduyar TE, Park YJ. Discontinuity capturing finite element formulations for nonlinear convection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering* 1986; **59**:307–325.
7. Tezduyar TE, Park YJ, Deans HA. Finite element procedures for time-dependent convection–diffusion–reaction systems. *International Journal for Numerical Methods in Fluids* 1987; **7**:1013–1033.
8. Tezduyar TE. Stabilized finite element formulations for incompressible flow computations. *Advances in Applied Mechanics* 1992; **28**:1–44.
9. Tezduyar TE, Mittal S, Ray SE, Shih R. Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity–pressure elements. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**:221–242.
10. Hughes TJR, Franca LP, Balestra M. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: a stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering* 1986; **59**:85–99.
11. Franca LP, Frey SL, Hughes TJR. Stabilized finite element methods: I. Application to the advective–diffusive model. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**:253–276.
12. Tezduyar TE, Osawa Y. Finite element stabilization parameters computed from element matrices and vectors. *Computer Methods in Applied Mechanics and Engineering* 2000; **190**:411–430.
13. Catabriga L, Coutinho ALGA, Tezduyar TE. Compressible flow SUPG parameters computed from element matrices. *Communications in Numerical Methods in Engineering* 2005; **21**:465–476.
14. Catabriga L, Coutinho ALGA, Tezduyar TE. Compressible flow SUPG parameters computed from degree-of-freedom submatrices. *Computational Mechanics* 2006; **38**:334–343.
15. Tezduyar TE. Adaptive determination of the finite element stabilization parameters. *Proceedings of the ECCOMAS Computational Fluid Dynamics Conference 2001*, Swansea, Wales, U.K., 2001 (CD-ROM).
16. Tezduyar TE. Computation of moving boundaries and interfaces and stabilization parameters. *International Journal for Numerical Methods in Fluids* 2003; **43**:555–575.
17. Tezduyar TE. Finite element methods for fluid dynamics with moving boundaries and interfaces. In *Encyclopedia of Computational Mechanics*, *Volume 3*: *Fluids*, Stein E, De Borst R, Hughes TJR (eds), Chapter 17. Wiley: New York, 2004.
18. Rispoli F, Corsini A, Tezduyar TE. Finite element computation of turbulent flows with the discontinuity-capturing directional dissipation (DCDD). *Computers and Fluids* 2007; **36**:121–126.
19. Corsini A, Rispoli F, Santoriello A, Tezduyar TE. Improved discontinuity-capturing finite element techniques for reaction effects in turbulence computation. *Computational Mechanics* 2006; **38**:356–364.
20. Valli AMP, Carey GF, Coutinho ALGA. Control strategies for timestep selection in finite element simulation of incompressible flows and coupled reaction–convection–diffusion processes. *International Journal for Numerical Methods in Fluids* 2005; **47**:201–231.
21. Tezduyar TE, Sathe S. Modeling of fluid–structure interactions with the space–time finite elements: solution techniques. *International Journal for Numerical Methods in Fluids* 2007; **54**:855–900.
22. Hughes TJR, Liu WK, Zimmermann TK. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering* 1981; **29**:329–349.
23. Tezduyar TE. Finite elements in fluids: stabilized formulations and moving boundaries and interfaces. *Computers and Fluids* 2007; **36**:191–206.

DOI: 10.1002/fld

24. Bazilevs Y, Calo VM, Tezduyar TE, Hughes TJR. YZβ discontinuity-capturing for advection-dominated processes with application to arterial drug delivery. *International Journal for Numerical Methods in Fluids* 2007; **54**: 593–608.
25. Saad Y, Schultz M. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**:856–869.
26. Le Quéré P. Accurate solutions to the square thermally driven cavity at high Rayleigh number. *Computers and Fluids* 1991; **20**:29–41.
27. de Vahl Davis G. Natural convection of air in a square cavity: a benchmark numerical solution. *International Journal of Numerical Methods in Fluids* 1983; **3**:249–264.
28. Ha MY, Kim I, Yoon HS, Yoon KS, Lee JR, Balachandar S, Chu HH. Two-dimensional and unsteady natural convection in a horizontal enclosure with a square body. *Numerical Heat Transfer*, *Part A* 2002; **41**:183–210.

DOI: 10.1002/fld