



# Multigrid solution of the Navier–Stokes equations at low speeds with large temperature variations

Peter M. Sockol \*

*Turbomachinery and Propulsion System, NASA Glenn Research Center, Mail Stop 5-10, Cleveland, OH 44135, USA*

Received 6 November 2002; received in revised form 22 July 2003; accepted 22 July 2003

---

## Abstract

Multigrid methods for the Navier–Stokes equations at low speeds and large temperature variations are investigated. The compressible equations with time-derivative preconditioning and preconditioned flux-difference splitting of the inviscid terms are used. Three implicit smoothers have been incorporated into a common multigrid procedure. Both full coarsening and semi-coarsening with directional fine-grid defect correction have been studied. The resulting methods have been tested on four 2D laminar problems over a range of Reynolds numbers on both uniform and highly stretched grids. Two of the three methods show efficient and robust performance over the entire range of conditions. In addition, none of the methods has any difficulty with the large temperature variations.

© 2003 Published by Elsevier B.V.

*Keywords:* Multigrid; Implicit smoothers; Time-derivative preconditioning; Flux-difference splitting; Defect correction; Navier–Stokes

---

## 1. Introduction

In a now classic work, Brandt [1] defined optimal multigrid convergence as reduction in the discrete system error to the level of the truncation error in a computational effort that is a small multiple of the effort to evaluate the discrete system residual. Today this is termed textbook multigrid efficiency (TME). While it is well known that TME can be achieved for fully elliptic problems, for the equations of fluid dynamics this performance has only been approached in a few simplified problems. Brandt and Yavneh [2] solved the incompressible Navier–Stokes equations for high-Reynolds number entering flows on a uniform rectangular grid. Thomas et al. [3] extended this work and applied it on a stretched rectangular grid to the boundary layer over a finite flat plate. Roberts et al. [4] solved the incompressible Euler equations on both unstructured triangular and structured quadrilateral grids for flows through a simple channel and over a symmetric airfoil. Roberts et al. [5] added nonlocal farfield boundary conditions and again applied it on a quadrilateral grid to the flow over a symmetric airfoil. A common feature of each of these works is that the fluid equations are separated into an advection part (advection–diffusion part for the viscous problem) and

---

\* Tel.: +1-216-433-5821; fax: +1-216-433-58202.

E-mail address: [psockol@nasa.gov](mailto:psockol@nasa.gov).

an elliptic part. The advection part is solved by marching in the downstream direction and the elliptic part is solved by multigrid. However, at this juncture it is not evident how these approaches are to be extended to general viscous flows with large separated regions.

Hence much effort continues to improve more general multigrid solvers and to extend them to a variety of problems. We note a number of recent efforts in the following. Pierce and Giles [6] examined methods for improving the performance of Runge–Kutta based solvers for compressible turbulent flows on stretched meshes. Steelant et al. [7] did the same for a number of implicit solvers applied to low Mach number laminar and turbulent flows. Amaladas and Kamath [8] investigated the performance of several implicit time-marching methods together with a number of upwind schemes for 2D inviscid and viscous compressible flows. Three papers have dealt with multigrid methods for the 3-D incompressible Navier–Stokes equations. Drikakis et al. [9] introduced a characteristics-based upwinding and used a smoother based on artificial compressibility (AC) and Runge–Kutta time-marching. Yuan [10] investigated three implicit time-marching smoothers together with AC. Montero et al. [11] evaluated both alternating-plane smoothers with standard coarsening and plane-implicit smoothers with semi-coarsening for the steady equations on staggered grids.

We next take note of a number of efforts to extend or improve the capability of multigrid solvers to treat a variety of physical phenomena. Epstein et al. [12] presented a 3-D compressible Navier–Stokes solver that used ENO differencing with defect correction together with Runge–Kutta time-marching for transonic turbulent flows. Sheffer et al. [13] presented a 2-D solver for compressible reacting flows that used a point implicit treatment of the chemical source terms, upwind differencing, and explicit time-marching. Gerlinger et al. [14] addressed the same problem, used point implicit treatment of the source terms with central differencing and matrix dissipation, added a two-equation turbulence model, and employed an implicit LUSGS solver as a smoother. As a final example we note the use of multigrid by Caughey [15] in a temporal subiteration process to converge the equations at each time step in the unsteady flow past both fixed and moving cylinders of square cross-section at moderate Reynolds numbers.

The present work is directed toward the development of efficient robust multigrid solvers for the compressible Navier–Stokes equations at low speed with large temperature variations. This is important for internal flows with heat transfer or combustion. To this end we employ time-derivative preconditioning to remove stiffness at low Mach numbers and preconditioned flux-difference splitting for the inviscid terms so that the implicit operator will be diagonally dominant. Three promising implicit smoothers have been incorporated in a common multigrid procedure. The resulting methods have been tested on four 2D laminar problems over a range of Reynolds numbers with both uniform and highly stretched grids. Convergence results from each of these studies are used to evaluate the relative strengths and weaknesses of the three solvers.

## 2. Governing equations

The governing equations are written in terms of nondimensional variables defined as follows (starred quantities are dimensional and subscript r denotes a reference value):

$$x = \frac{x^*}{L_r}, \quad y = \frac{y^*}{L_r}, \quad t = \frac{t^*}{L_r/V_r}, \quad p = \frac{p^*}{\rho_r V_r^2}, \quad u = \frac{u^*}{V_r},$$

$$v = \frac{v^*}{V_r}, \quad T = \frac{T^*}{T_r}, \quad \rho = \frac{\rho^*}{\rho_r}, \quad \mu = \frac{\mu^*}{\mu_r}, \quad \kappa = \frac{\kappa^*}{\kappa_r},$$

where  $x$  and  $y$  are space dimensions,  $t$ ,  $p$ ,  $u$ ,  $v$ ,  $T$ ,  $\rho$ ,  $\mu$  and  $\kappa$  are time, pressure,  $x$  and  $y$  velocities, temperature, density, viscosity and thermal conductivity, respectively. A perfect gas with constant specific heats is assumed. The primary nondimensional parameters are given by

$$Ma = \frac{V_r}{c_r}, \quad Re = \frac{\rho_r V_r L_r}{\mu_r}, \quad Pr = \frac{\mu_r C_{pr}}{\kappa_r}, \quad Fr = \frac{V_r^2}{gL_r}, \quad Ec = \frac{V_r^2}{C_{pr} T_r} = (\gamma - 1) Ma^2,$$

which are, respectively, the Mach, Reynolds, Prandtl, Froude and Eckert numbers where  $c_r = \sqrt{\gamma R T_r}$  is the reference speed of sound,  $\gamma$  is the ratio of specific heats,  $R$  is the gas constant,  $C_{pr}$  is the constant pressure specific heat, and  $g$  is the gravitational constant. Finally, for low Mach number flows, it is convenient to subtract off the large background pressure by setting

$$p = \bar{p} + p' \quad (1)$$

with  $\bar{p} = (\gamma Ma^2)^{-1}$ .

The two-dimensional compressible Navier–Stokes equations with time-derivative preconditioning are written in vector, conservation-law form

$$P_c^{-1} \frac{\partial U}{\partial t} + \frac{\partial}{\partial x} (E - E_v) + \frac{\partial}{\partial y} (F - F_v) = H, \quad (2)$$

where

$$\begin{aligned} U &= (\rho, \rho u, \rho v, e')^T, \\ E &= (\rho u, \rho u^2 + p', \rho uv, \rho u h_t)^T, \\ F &= (\rho v, \rho uv, \rho v^2 + p', \rho v h_t)^T, \\ E_v &= R^{xx} \frac{\partial Q}{\partial x} + R^{xy} \frac{\partial Q}{\partial y}, \\ F_v &= R^{yx} \frac{\partial Q}{\partial x} + R^{yy} \frac{\partial Q}{\partial y}, \\ H &= Fr^{-1} (0, 0, -\rho, -Ec\rho v)^T, \\ Q &= (p', u, v, T)^T, \end{aligned} \quad (3)$$

$P_c$  is the preconditioning matrix,  $e' = \rho h_t - Ec p'$  is the total energy, and  $h_t = T + (1/2)Ec(u^2 + v^2)$  is the total enthalpy. Note that the source vector  $H$  contains a gravitational body force in the negative  $y$ -direction and also that subtraction of the constant background pressure  $\bar{p}$  does not change the basic form of the equations.

The density is given by the equation of state for an ideal gas

$$\rho = (1 + p'/\bar{p})/T, \quad (4)$$

and the diffusion coefficient matrices are given by

$$\begin{aligned} R^{xx} &= Re^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{4}{3}\mu & 0 & 0 \\ 0 & 0 & \mu & 0 \\ 0 & \frac{4}{3}Ec\mu u & Ec\mu v & Pr^{-1}\kappa \end{bmatrix}, & R^{xy} &= Re^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3}\mu & 0 \\ 0 & \mu & 0 & 0 \\ 0 & Ec\mu v & -\frac{2}{3}Ec\mu u & 0 \end{bmatrix}, \\ R^{yx} &= Re^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 \\ 0 & -\frac{2}{3}\mu & 0 & 0 \\ 0 & -\frac{2}{3}Ec\mu v & Ec\mu u & 0 \end{bmatrix}, & R^{yy} &= Re^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 \\ 0 & 0 & \frac{4}{3}\mu & 0 \\ 0 & Ec\mu u & \frac{4}{3}Ec\mu v & Pr^{-1}\kappa \end{bmatrix}. \end{aligned} \quad (5)$$

The preconditioning matrix  $P_c$  is adapted from the work of Turkel [16]. First consider the inviscid equations written in terms of the primitive variables  $V = (p', u, v, S')^T$ , with  $dS' = dp' - c^2 d\rho$  and  $c^2 = Ma^{-2}T$ . These are written as

$$P_p^{-1} \frac{\partial V}{\partial t} + A_p \frac{\partial V}{\partial x} + B_p \frac{\partial V}{\partial y} = 0, \tag{6}$$

where

$$A_p = \begin{bmatrix} u & \rho c^2 & 0 & 0 \\ \rho^{-1} & u & 0 & 0 \\ 0 & 0 & u & 0 \\ 0 & 0 & 0 & u \end{bmatrix}, \quad B_p = \begin{bmatrix} v & 0 & \rho c^2 & 0 \\ 0 & v & 0 & 0 \\ \rho^{-1} & 0 & v & 0 \\ 0 & 0 & 0 & v \end{bmatrix}. \tag{7}$$

For this system the simplest form of Turkel’s preconditioner is written as

$$P_p^{-1} = \begin{bmatrix} m^{-2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{8}$$

where  $m^2 = \beta^2/c^2$  and  $\beta^2$  is chosen to bring the system eigenvalues closer together at low Mach numbers. The eigenvalues of  $P_p A_p$  are given by

$$\lambda = u - u_0 - c_0, u, u, u - u_0 + c_0 \tag{9}$$

with

$$\begin{aligned} c_0^2 &= u_0^2 + \beta^2, \\ u_0 &= \frac{1}{2}(1 - m^2)u. \end{aligned} \tag{10}$$

The conservation-law form is then given by  $P_c = MP_p M^{-1}$  with  $M = \partial U / \partial V$ . Finally, following Choi and Merkle [17], we express Eq. (2) in terms of the “viscous” variables  $Q$

$$P_Q^{-1} \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}(E - E_v) + \frac{\partial}{\partial y}(F - F_v) = H, \tag{11}$$

where  $P_Q^{-1} = MP_p^{-1}N$  and  $N = \partial V / \partial Q$ .

For most flows it is sufficient to set  $\beta^2$  equal to the square of the local speed together with lower and upper limits. However, for some low Reynolds number cases, convergence is improved by introducing the viscous limit of Choi and Merkle [17]. Hence we take

$$\beta^2 = \min[\max(\beta_0^2, u^2 + v^2, \beta_v^2), c^2], \tag{12}$$

where  $\beta_0^2$  is a global user specified lower limit and

$$\beta_v^2 = \frac{\alpha(\alpha - 1)u^2}{1 + (\alpha - 1)u^2/c^2}. \tag{13}$$

Here  $\alpha = RCV/Re_{ds}$ , RCV is the ratio of CFL to von Neumann numbers (also user specified) and  $Re_{ds} = Re|u|ds$  is the local cell Reynolds number with  $u$  the convective velocity and  $ds$  the spacing between cell centers.

### 3. Discrete formulation

The governing equations are solved on a stretched rectangular grid with unknowns stored at cell centers. While extension to a general curvilinear coordinate system is straightforward, the simpler system is sufficient for our current purposes. Integration of Eq. (11) over a cell together with Euler implicit time differencing gives

$$P_Q^{-1}(\text{dx dy}/\text{dt})\Delta Q = -R^{n+1}, \quad (14)$$

where dx and dy are cell dimensions, dt is the time step,  $\Delta Q = Q^{n+1} - Q^n$ , and the area-weighted residual  $R$  is given by

$$R = \delta_x(E - R^{xx} \text{dx}^{-1} \delta_x Q - R^{xy} \text{dy}^{-1} \delta_y Q) \text{dy} + \delta_y(F - R^{yx} \text{dx}^{-1} \delta_x Q - R^{yy} \text{dy}^{-1} \delta_y Q) \text{dx} - H \text{dx dy}. \quad (15)$$

Here  $\delta_x$  and  $\delta_y$  are central difference operators in  $x$  and  $y$ , respectively.

The cell-face inviscid fluxes  $E$  and  $F$  in Eq. (15) are approximated by preconditioned flux-difference splitting [18]. This choice is normally dictated by the need for a more accurate capturing of discontinuities such as shocks or contact surfaces. In the present case, however, since we intend to employ implicit methods, this choice is made as a means of obtaining positive inviscid contributions to the global matrix to improve the efficiency of the relaxation schemes. It should be noted that with time-derivative preconditioning all the wave speeds are of the same order, see Eq. (9), and the matrix condition number is significantly reduced at low Mach numbers. Hence we write

$$E = \frac{1}{2}[E(U^-) + E(U^+)] - \frac{1}{2}\hat{P}_c^{-1}|\hat{P}_c\hat{A}_c|(U^+ - U^-), \quad (16)$$

where  $U^-$  and  $U^+$  are, respectively, left and right states,  $A_c = \partial E/\partial U$ ,  $\hat{A}_c$  denotes evaluation at an average state, and  $P_c^{-1}|P_c A_c| = MP_p^{-1}|P_p A_p|M^{-1}$ . Here  $|P_p A_p|$  is computed using the eigenvectors and absolute eigenvalues of  $P_p A_p$ . Eq. (16) is rewritten in terms of the “viscous” variables  $Q$

$$E = \frac{1}{2}[E(Q^-) + E(Q^+)] - \frac{1}{2}\hat{A}^a(Q^+ - Q^-), \quad (17)$$

where  $A^a = MP_p^{-1}|P_p A_p|N$ . The left and right states  $Q^\mp$  are given by an upwind-biased interpolation using van Leer’s MUSCLE approach [19]

$$Q_{i\pm\frac{1}{2},j}^\mp = Q_{i,j} \pm \frac{1}{4}\sigma_x[(1 \mp \kappa_x)\delta_x^- + (1 \pm \kappa_x)\delta_x^+]Q_{i,j}, \quad (18)$$

where  $Q_{i,j}$  represents a cell-averaged value,  $\delta_x^-$  and  $\delta_x^+$  are backward and forward differences in  $x$ , respectively, and for  $\sigma_x = 1$ ,  $\kappa_x$  determines the spatial accuracy. Setting  $\kappa_x = -1$  corresponds to a fully upwind scheme,  $\kappa_x = +1$  to a central difference scheme, and  $\kappa_x = 1/3$  to a third-order upwind-biased scheme. Setting  $\sigma_x = 0$  reduces the scheme to first order. In the present implementation the average state  $\hat{Q}$  is taken as a simple average of  $Q^-$  and  $Q^+$  and the use of limiters is not required. The cell-face flux  $F$  in the  $y$ -direction is approximated by expressions similar to Eqs. (16) and (17) with  $A_c$  replaced by  $B_c = \partial F/\partial U$  and Eq. (18) for  $Q_{i\pm 1/2,j}^\mp$  is replaced by one for  $Q_{i,j\pm 1/2}^\mp$  with interpolation in  $y$  instead of  $x$ .

Eq. (14) is linearized about known values at level  $n$  and only first-order contributions from  $E$  and  $F$  are retained on the left-hand side

$$\begin{aligned} & \left[ P_Q^{-1} \text{dx dy}/\text{dt} + \{ \delta_x^-(S_x^+ \hat{A}^+) + \delta_x^+(S_x^- \hat{A}^-) - \delta_x R^{xx} \text{dx}^{-1} \delta_x \} \text{dy} \right. \\ & \left. + \{ \delta_y^-(S_y^+ \hat{B}^+) + \delta_y^+(S_y^- \hat{B}^-) - \delta_y R^{yy} \text{dy}^{-1} \delta_y \} \text{dx} - D \text{dx dy} \right] \Delta Q = -R^n, \end{aligned} \quad (19)$$

where  $S_x^\pm$  and  $S_y^\pm$  are half cell shift operators in  $\pm x$  and  $\pm y$ , respectively, and

$$A^\pm = \frac{1}{2}(A \pm A^a), \quad B^\pm = \frac{1}{2}(B \pm B^a), \quad (20)$$

with  $A = MA_p N$ ,  $B = MB_p N$  and  $D = \partial H / \partial Q$ . Note that  $A$  and  $B$  are evaluated at the average state and that the variations of  $A^a$  and  $B^a$  with  $Q$  have been neglected.

The discrete formulation is completed by the specification of the time step  $dt$ . We set

$$dx dy / dt = [r(P_p A_p) dy + r(P_p B_p) dx] / CFL, \quad (21)$$

where the spectral radii  $r$  are given by

$$\begin{aligned} r(P_p A_p) &= |u - u_{0x}| + c_{0x}, \\ r(P_p B_p) &= |v - u_{0y}| + c_{0y}. \end{aligned} \quad (22)$$

Here  $c_{0x}$  and  $u_{0x}$  are given by Eq. (10), while  $c_{0y}$  and  $u_{0y}$  are given by the same expressions with  $u$  replaced by  $v$ .

#### 4. Boundary conditions

With the implicit relaxation methods used in this work, very simple boundary condition treatments have proven quite effective. Boundary values are located on the physical boundary, one half cell from the nearest interior point. The surface fluxes are obtained directly from Eqs. (3) with the normal derivatives in the viscous fluxes approximated by one-sided second-order differences. At an inlet  $u$ ,  $v$ , and  $T$  are specified, while  $p'$  is obtained by linear extrapolation from the interior. At a solid wall  $u$  and  $v$  are specified,  $T$  is either specified or obtained from a second-order one-sided adiabatic wall condition, and  $p'$  is again found by linear extrapolation. Finally, at an outlet surface  $p'$  is specified and  $u$ ,  $v$ , and  $T$  are obtained by linear extrapolation from the interior. On the left-hand side of Eq. (19), boundary conditions on  $\Delta Q$  are treated implicitly. Thus, where a component of  $Q$  is specified, the corresponding component of  $\Delta Q$  is set to zero. On the other hand, where an element of  $Q$  is found by linear extrapolation or adiabatic wall condition, that element of  $\Delta Q$  is set equal to its value at the interior point one half cell away. These conditions are represented by a simple transfer relation

$$\Delta Q_b = L_b \Delta Q, \quad (23)$$

where  $\Delta Q_b$  is the boundary value,  $\Delta Q$  is the value at the interior point, and  $L_b$  is a diagonal matrix. At an inlet or solid wall with  $T$  specified,  $L_b = \text{Diag}(1, 0, 0, 0)$ ; at an adiabatic wall,  $L_b = \text{Diag}(1, 0, 0, 1)$ ; and at an outlet,  $L_b = \text{Diag}(0, 1, 1, 1)$ .

#### 5. Smoothing methods

It is generally recognized that the smoothing or relaxation method is the most important part of a multigrid solution process. In the present work we investigate three implicit methods as part of a common multigrid process. Each of these can be found in the literature, hence only a brief description is given here. The descriptions are presented in terms of the global matrix form of the discrete system of equations. Expansion of Eq. (19) gives

$$\bar{D}_{i,j}\Delta Q_{i-1,j} + \bar{A}_{i,j}\Delta Q_{i,j-1} + \bar{B}_{i,j}\Delta Q_{i,j} + \bar{C}_{i,j}\Delta Q_{i,j+1} + \bar{E}_{i,j}\Delta Q_{i+1,j} = -R_{i,j}, \quad (24)$$

where the block matrix coefficients,  $\bar{A}$  to  $\bar{E}$ , are given in Appendix A. When the cell  $(i, j)$  is located adjacent to a boundary, Eq. (23) is used to eliminate the boundary value, the coefficient of the boundary value is set to zero, and the central coefficient  $\bar{B}_{i,j}$  is modified accordingly. An example of such a modified coefficient is also given in Appendix A.

The first smoother is Line Block Gauss–Seidel (LBGS). For sweeps in the  $+x$ -direction, this is written as

$$\left. \begin{aligned} \bar{B}'_{i,j} &= \bar{B}_{i,j} - \bar{A}_{i,j}\bar{C}'_{i,j-1} \\ \bar{B}'_{i,j}\bar{C}'_{i,j} &= \bar{C}_{i,j} \\ \bar{B}'_{i,j}\Delta Q^*_{i,j} &= -R_{i,j} - \bar{D}_{i,j}\Delta Q_{i-1,j} - \bar{A}_{i,j}\Delta Q^*_{i,j-1} \\ \Delta Q_{i,j} &= \Delta Q^*_{i,j} - \bar{C}'_{i,j}\Delta Q_{i,j+1} \end{aligned} \right\} \begin{array}{l} \text{for } j = 1, \dots, n_y, \\ \text{for } i = 1, \dots, n_x, \\ \text{for } j = n_y, \dots, 1, \end{array} \quad (25)$$

where  $\bar{C}'_{i,j}$  and  $\Delta Q^*_{i,j}$  are stored along the line and there is one LU decomposition of  $\bar{B}'_{i,j}$  per cell during the first  $j$ -sweep. For cases with a predominant flow in the  $x$ -direction, LBGS is implemented by alternately sweeping in the  $+x$ - and  $-x$ -directions. For predominantly recirculating flows, we use a symmetric pattern of sweeps in the  $+x$ -,  $+y$ -,  $-y$ -, and  $-x$ -directions.

The second smoother is LU Symmetric Gauss–Seidel (LUSGS). This is written as

$$\left. \begin{aligned} \bar{B}_{i,j}\bar{C}'_{i,j} &= \bar{C}_{i,j} \\ \bar{B}_{i,j}\bar{E}'_{i,j} &= \bar{E}_{i,j} \\ \bar{B}_{i,j}\Delta Q^*_{i,j} &= -R_{i,j} - \bar{A}_{i,j}\Delta Q^*_{i,j-1} - \bar{D}_{i,j}\Delta Q^*_{i-1,j} \\ \Delta Q_{i,j} &= \Delta Q^*_{i,j} - \bar{C}'_{i,j}\Delta Q_{i,j+1} - \bar{E}'_{i,j}\Delta Q_{i+1,j} \end{aligned} \right\} \begin{array}{l} \text{for } i = 1, \dots, n_x, \quad j = 1, \dots, n_y, \\ \text{for } i = n_x, \dots, 1, \quad j = n_y, \dots, 1, \end{array} \quad (26)$$

where  $\bar{C}'_{i,j}$ ,  $\bar{E}'_{i,j}$  and  $\Delta Q^*_{i,j}$  are stored over the field and there is one LU decomposition of  $\bar{B}_{i,j}$  per cell in the first sweep. Note that this form of LUSGS requires less global storage and one less LU decomposition per cell than the usual form. For cases with a predominant flow in the  $x$ -direction, LUSGS is implemented by alternately switching the directions of the first and second sweeps.

The third and final smoother is Block Incomplete LU decomposition (BILU). This is written as

$$\left. \begin{aligned} \bar{B}'_{i,j} &= \bar{B}_{i,j} - \bar{A}_{i,j}\bar{C}'_{i,j-1} - \bar{D}_{i,j}\bar{E}'_{i-1,j} \\ \bar{B}'_{i,j}\bar{C}'_{i,j} &= \bar{C}_{i,j} \\ \bar{B}'_{i,j}\bar{E}'_{i,j} &= \bar{E}_{i,j} \\ \bar{B}'_{i,j}\Delta Q^*_{i,j} &= -R_{i,j} - \bar{A}_{i,j}\Delta Q^*_{i,j-1} - \bar{D}_{i,j}\Delta Q^*_{i-1,j} \\ \Delta Q_{i,j} &= \Delta Q^*_{i,j} - \bar{C}'_{i,j}\Delta Q_{i,j+1} - \bar{E}'_{i,j}\Delta Q_{i+1,j} \end{aligned} \right\} \begin{array}{l} \text{for } i = 1, \dots, n_x, \quad j = 1, \dots, n_y, \\ \text{for } i = n_x, \dots, 1, \quad j = n_y, \dots, 1, \end{array} \quad (27)$$

where  $\bar{C}'_{i,j}$ ,  $\bar{E}'_{i,j}$  and  $\Delta Q^*_{i,j}$  are stored over the field and one LU decomposition of  $\bar{B}'_{i,j}$  per cell is performed during the first sweep. Again for cases with a predominant flow in the  $x$ -direction, BILU is implemented by alternately switching the directions of the first and second sweeps.

In each of the three smoothers, the interior unknowns are updated by  $Q_{i,j}^{n+1} = Q_{i,j}^n + \Delta Q_{i,j}$  only after  $\Delta Q_{i,j}$  has been obtained for the entire field. Following this the boundary values of  $Q_{i,j}^{n+1}$  are computed based on the new interior values.

### 6. Multigrid and defect correction

Relaxation methods, such as those of the previous section, are in general much more efficient at reducing short-wavelength error components on a given grid than those of longer wavelength. Multigrid seeks to overcome this problem by transferring the long-wave components of the solution to a sequence of coarser grids where relaxation is more effective and much cheaper. Since the FAS–FMG (full approximation scheme–full multigrid) technique used in this work has been well documented in the literature, the present description of the multigrid process will be brief. The focus will, instead, be on the current implementation and in particular on those aspects that were found to be important in achieving a fast Navier–Stokes solver.

Introduce a sequence of grids,  $k = m$  to 1, where  $m$  is the finest and each grid  $k < m$  is coarser than grid  $k + 1$  in one or both directions. For  $k = m$  we seek the solution of the nonlinear system

$$N(Q^k) = 0 \tag{28}$$

with  $N$  representing the area-weighted residual given by Eq. (15). On any grid  $k$  let  $S^k$  denote a single smoothing pass of the system

$$\tilde{N}(Q^k) = F^k. \tag{29}$$

Here  $\tilde{N}$  is an approximation to  $N$  in which one or both convective operators *may* be replaced by a first-order discretization and  $F^k$  is defined below. Next introduce a fine-to-coarse restriction operator  $\hat{I}_{k+1}^k$  for unknowns, a restriction operator  $I_{k+1}^k$  for residuals,  $R^k = F^k - \tilde{N}(Q^k)$ , and a coarse-to-fine prolongation operator  $I_k^{k+1}$  for corrections. With these definitions, the FAS multigrid cycle  $M^k$  for improving an approximation  $Q^k$  is defined recursively as follows:

For  $k = m$ ,  $F^k \leftarrow \tilde{N}(Q^k) - N(Q^k)$ .

For  $k = 1$ , solve Eq. (29) by several smoothing sweeps.

For  $k > 1$ , do these steps:

(a) smooth on grid  $k$ ,

$$Q^k \leftarrow (S^k)^{v_1} Q^k,$$

(b) restrict  $Q^k$  to grid  $k - 1$ ,

$$Q^{k-1} \leftarrow \hat{I}_k^{k-1} Q^k,$$

(c) restrict  $R^k$  to grid  $k - 1$  and form  $F^{k-1}$ ,

$$F^{k-1} \leftarrow \tilde{N}(Q^{k-1}) + I_k^{k-1} R^k,$$

(d) perform  $\gamma$  multigrid cycles on  $Q^{k-1}$ ,

$$Q^{k-1} \leftarrow (M^{k-1})^\gamma Q^{k-1},$$

(e) prolong corrections to grid  $k$ ,

$$Q^k \leftarrow Q^k + I_{k-1}^k (Q^{k-1} - \hat{I}_k^{k-1} Q^k),$$

(f) smooth on grid  $k$ ,

$$Q^k \leftarrow (S^k)^{v_2} Q^k.$$

In general  $v_1$  and  $v_2$  can be considered functions of the grid level  $k$ .



For  $\gamma = 1$ , a  $V$ -cycle,  $V(v_1, v_2)$ , is obtained. For  $\gamma = 2$  the more robust  $W$ -cycle,  $W(v_1, v_2)$ , is obtained. We also introduce the  $F$ -cycle, which provides coarse grid performance that is intermediate between that of the  $V$ - and  $W$ -cycles. This is defined recursively as a  $\gamma = 2$  cycle in which the first iteration at grid  $k$  is an  $F$ -cycle and the second is a  $V$ -cycle. Finally, we obtain the full FAS–FMG technique by starting the computation on a very coarse grid, iterating to ‘convergence’ with the FAS process, and interpolating the result to obtain initial values on the next finest grid. In this way the first approximation on the finest grid is already close in much of the domain, an important consideration in nonlinear problems. Convergence criteria for each stage in the FAS–FMG process as used in the present work are described in the next section.

The sequence of grids,  $k = m$  to 1, is given by either full or semi-coarsening. In full coarsening, each cell on grid  $k$  is obtained by combining four cells on grid  $k + 1$  such that there are half as many cells in each direction on grid  $k$  as on  $k + 1$ . This tends to work well for flows that are predominantly recirculating and where the grid is only moderately stretched. For flows with a predominant flow direction and strong grid stretching in the cross-stream direction, which is common in many engineering applications, a more robust albeit more expensive solver can be obtained by using semi-coarsening. In this case each cell on grid  $k$  is obtained by combining two cells in the flow direction on grid  $k + 1$  such that there are half as many cells in the flow direction and the same number in the cross-stream direction on grid  $k$  as on  $k + 1$ . Both of these situations will be illustrated with sample problems.

The restriction operators,  $\hat{I}_{k+1}^k$  for unknowns and  $I_{k+1}^k$  for residuals, both use full weighting. For  $\hat{I}_{k+1}^k$  interior values are given by an area-weighted average over the fine grid cells from which the coarse cell is constructed. The unknown coarse-grid boundary values are then updated using the methods of section four. Since the residuals are conservatively differenced and area weighted,  $I_{k+1}^k$  is obtained by summing the fine-grid fluxes over the boundaries of the coarse grid cell and then adding the contributions from the fine-grid source terms. The prolongation operator,  $I_k^{k+1}$  for corrections, is given by bilinear interpolation in the equally spaced computational domain. Once the interior points have been corrected, the unknown boundary values are again updated by the methods of section four. The same method is also used to interpolate converged results from a coarse grid to initialize values on the next finest grid.

Another way to increase solver robustness is through the use of fine-grid defect correction. Normally this involves using first-order convective operators in both directions in  $\tilde{N}$ . This produces a large fine-grid source term  $F^m$  and substantially slows convergence. In the current work it has been found that using a first-order operator in the flow direction while retaining a higher order in the cross-stream direction gives an efficient yet quite robust solver. The source term  $F^m$  is updated every one to four fine-grid sweeps.

The multigrid solvers employed in this work have been coded to permit  $V$ -,  $W$ -, or  $F$ -cycles, including the possibility of  $k$ -dependent  $nu_1$  and  $v_2$ . While all the computations reported here were performed with  $W(1, 0)$  from the fine grid, as these were the most robust for all methods on all problems, higher values of  $v_1$  were sometimes used during the starting process on coarser grids. Finally, we note that the different sweeping patterns introduced in Section 5 in conjunction with each of the smoothers have been interleaved with the multigrid process. A sweep counter is established for every grid level and on each visit to that level the next direction in the sweep pattern for that grid is performed. This proved to be sufficient to give all the convergence benefits of the sweeping pattern.

## 7. Convergence criteria

In establishing convergence criteria for an FAS–FMG procedure it is useful to employ a measure that does not change by a large amount from one grid to the next. To this end we introduce the following  $L_2$  residual norm

$$L_2 R^k = \left\{ \sum_{i,j=1}^{n_x^k, n_y^k} |R_{i,j}^k|^2 / 4n_x^k n_y^k \right\}^{\frac{1}{2}} / d\bar{A}^k, \quad (30)$$

where  $n_x^k$  and  $n_y^k$  are the number of cells on grid  $k$  in  $x$  and  $y$ , respectively, and  $d\bar{A}^k$  is the average cell area on level  $k$ . Then on the finest grid  $m$  convergence is taken as

$$L_2 R^m < 10^{-6}. \quad (31)$$

In most cases this produces at least four significant figures in the solution. For intermediate grids  $k > 1$  in the FAS–FMG process, convergence before interpolating to the next finer grid is taken as

$$L_2 R^k < 10^{-2}, \quad (32)$$

with a maximum of 50 sweeps on level  $k$ , and for the coarsest grid  $k = 1$  ‘solution’ is given by

$$L_2 R^1 < L_2 R^k / 10, \quad (33)$$

where now  $L_2 R^k$  is the most recent error on the current finest grid and a maximum of four sweeps is taken on the coarsest grid.

## 8. Results and discussion

Four numerical test problems have been chosen to evaluate the efficiency and robustness of the multigrid method with each of the three smoothers. In three of the problems, existing numerical results are used to check the accuracy of the current approach. In two of the problems, large temperature variations are used to test the ability of the solvers in this area. For each problem the behavior of the solvers is tested over a range of Reynolds numbers  $Re$  on uniform grids. In addition, although uniform grids are adequate to resolve the physics in each case, the solvers are also tested over a range of stretched grids for each problem. Finally, since each of these cases is in the incompressible range, the reference Mach number  $Ma$  is set to  $10^{-4}$  for all computations.

For each test problem and smoother the parameters,  $\beta_0^2$ , RCV, and CFL, have been chosen to be nearly optimum over the range of Reynolds numbers presented. At higher values of  $Re$  it may be necessary to change one or more of these parameters. For example for the case of the lid-driven cavity at  $Re = 7500$  (not included below), good convergence is obtained for CFL = 10 on grids 5 and 6 while CFL = 40 is used for all cases presented at  $Re = 5000$  and below.

Note all computations were performed on an SGI Octane workstation with an R12000 (300 MHz) CPU.

### 8.1. Lid-Driven cavity

The lid-driven cavity is a common benchmark for recirculating flows. The second-order streamfunction-vorticity results of Ghia et al. [20], computed on a uniform  $256 \times 256$  grid, are generally accepted as standard. Isothermal flow is set up in a square cavity with a top lid that moves to the right at constant speed  $u = 1$ . In the current work, for this flow, the residual  $R^k$  is evaluated with third-order upwind differences in both directions on all grids. Streamfunction contours, computed on a uniform  $128 \times 128$  grid, for  $Re = 1000$  and 5000, are shown in Fig. 1. Profiles of  $u$  on the vertical centerline, computed on uniform  $128 \times 128$  and  $256 \times 256$  grids, for  $Re = 100, 1000$  and 5000, are compared with the standard results in Fig. 2. The agreement on both grids for all three Reynolds numbers is very good.

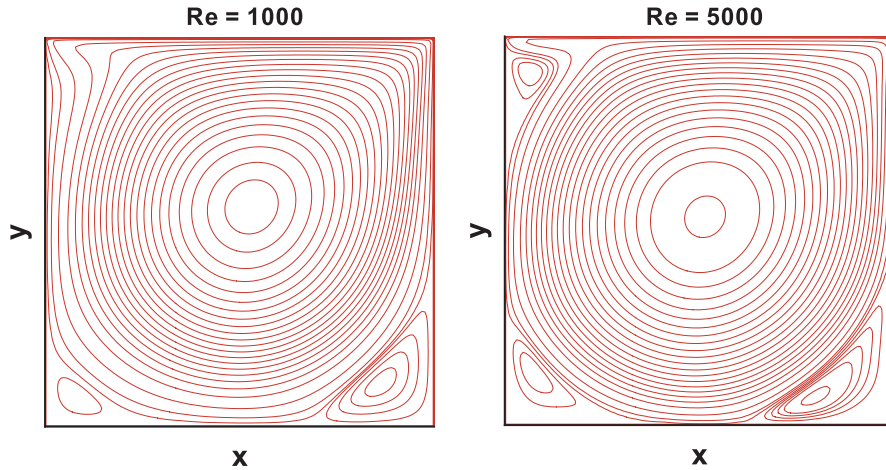


Fig. 1. Streamfunction contours for lid-driven cavity at  $Re = 1000$  and  $5000$ .

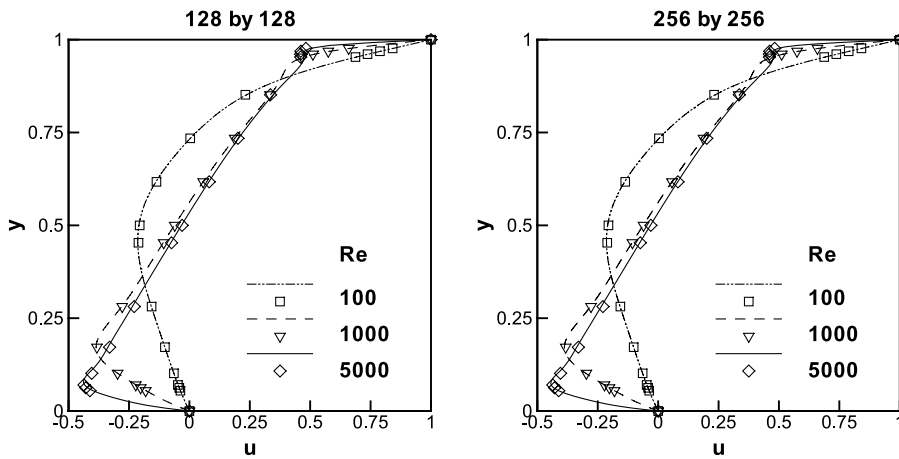


Fig. 2. Profiles of  $u$  on vertical centerline for lid-driven cavity on two uniform grids at three Reynolds numbers. Curves are present results; symbols are from Ghia et al. [20] on a  $256 \times 256$  grid.

The first set of performance results is for a uniform  $128 \times 128$  grid with  $Re$  varying from 100 to 5000. Six grids are used in the full coarsening multigrid process with computation starting on grid 2. Computations are performed with  $\beta_0^2 = 0.4$ ,  $R_{CV} = 2.2$  and  $CFL = 40.0$  for each method on all grids. Convergence plots of the mean fine grid residual  $L_2 R^6$  vs. work units are shown in Fig. 3 for all three smoothers. One work unit is defined as the CPU time required to evaluate the residual of the discrete equations on the fine grid, 0.088 s on the SGI for this case. Note that in the residual evaluation, which is the same for all three methods, the flux across each cell boundary is computed only once. In this figure each symbol on a plot corresponds to a single fine-grid sweep and the horizontal offset of the first point from the origin is the initialization time on the coarser grids. For these computations all three smoothers are comparable with the faster convergence of LUSGS and BILU compensated for by the shorter cpu time per sweep for LBGS.

The second set of results is for a stretched  $128 \times 128$  grid at  $Re = 1000$  with the grid spacing at all four walls varying from  $ds_0 = 0.002$  to  $0.00002$ . One-dimensional grid stretching in each direction is performed

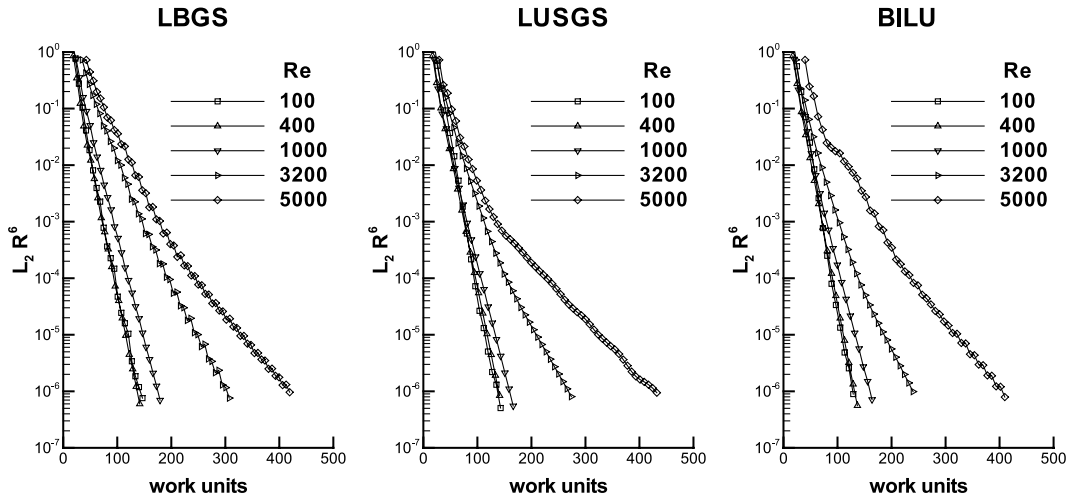


Fig. 3. Lid-driven cavity: convergence plots for each method on a uniform  $128 \times 128$  grid.

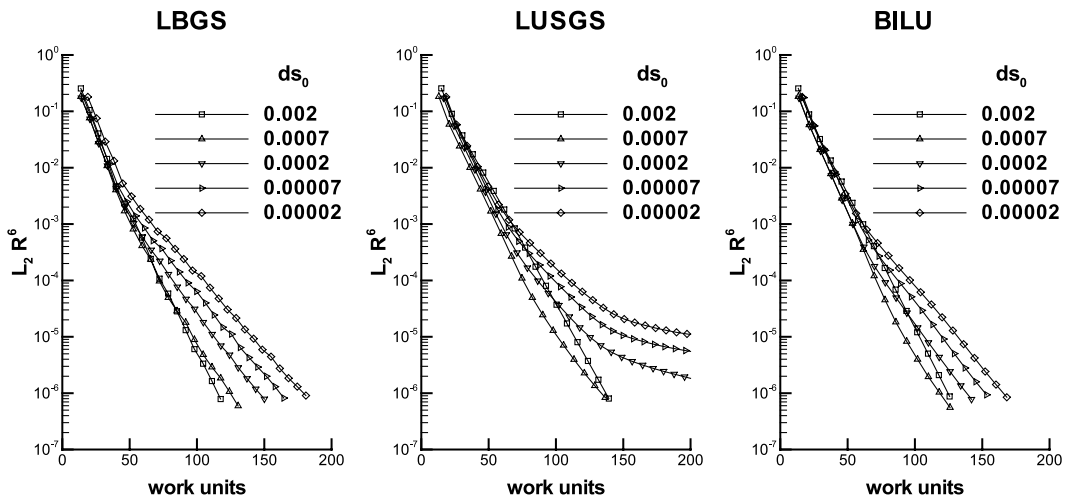


Fig. 4. Lid-driven cavity: convergence plots for each method on a stretched  $128 \times 128$  grid for  $Re = 1000$ .

by a procedure that permits specification of the grid spacing at each end of the interval. This is described in Appendix B. Six grids are used in the multigrid process with computation starting on grid 2. The values of  $\beta_0^2$ , RCV and CFL are the same as in the previous set. Convergence plots of fine grid  $L_2 R^6$  vs. work units are shown in Fig. 4 for all three smoothers. For the stretched grid LBGS and BILU are again comparable. The performance of LUSGS, however, rapidly degrades as the stretching increases.

### 8.2. Buoyancy-driven cavity

This is a flow in a square cavity with insulated horizontal walls, vertical walls with constant temperatures  $T_h$  on the left and  $T_c$  on the right, and gravity force in the negative  $y$ -direction. We consider the particular case of  $T_h/T_c = 4$  and varying Rayleigh number  $Ra$ , where  $Ra = \Delta T \rho_r^2 g C_{pr} L_r^3 / \mu_r \kappa_r$  with  $\Delta T = 2(T_h - T_c) / (T_h + T_c)$ . Yu et al. [21] have reported detailed results for this problem, which were obtained with a

least-squares finite-element method. The computations were performed on a  $129 \times 129$  moderately stretched grid with bi-quadratic elements. In order to compare with these results we use constant viscosity  $\mu$  and thermal conductivity  $\kappa$  and set  $Fr = 1.2$ ,  $Pr = 0.7$ . In the present work the residual  $R^k$  is again evaluated with third-order upwind differences in both directions on all grids. Streamfunction contours, computed on a uniform  $128 \times 128$  grid, for  $Ra = 10^5$  and  $10^6$ , are shown in Fig. 5. Profiles of  $u$  on the vertical centerline, computed on uniform 128 by 128 and  $256 \times 256$  grids, for  $Ra = 10^3$ ,  $10^5$  and  $10^6$ , are compared with those of Yu et al. [21] in Fig. 6. The agreement on both grids at  $Ra = 10^3$  and  $10^5$  is very good, both with each other and with the reference, but those at  $10^6$ , while agreeing with each other, disagree substantially with those of the reference. In [21] the streamfunction plot for this case shows a somewhat different structure with two additional secondary vortices. The present calculations cast this finding into doubt. Note our computations on a  $512 \times 512$  grid agree with our results on the two coarser grids.

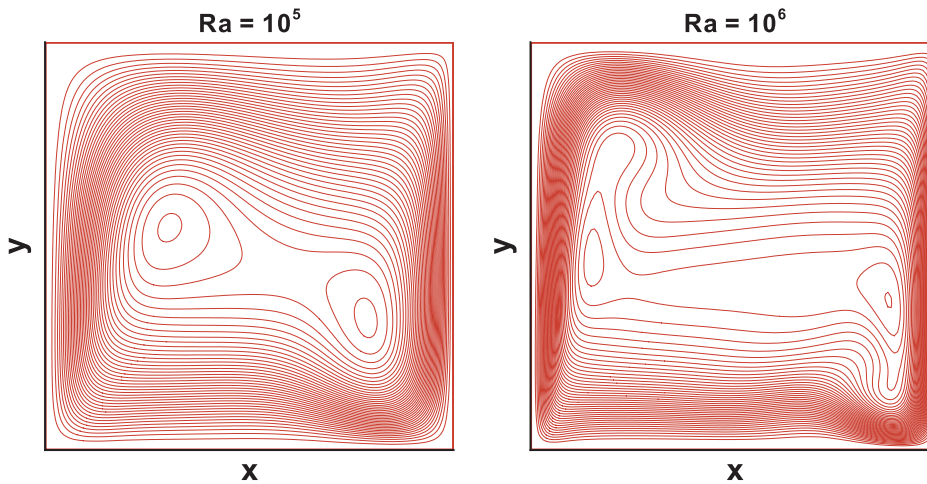


Fig. 5. Streamfunction contours for buoyancy-driven cavity at  $Ra = 10^5$  and  $10^6$ .

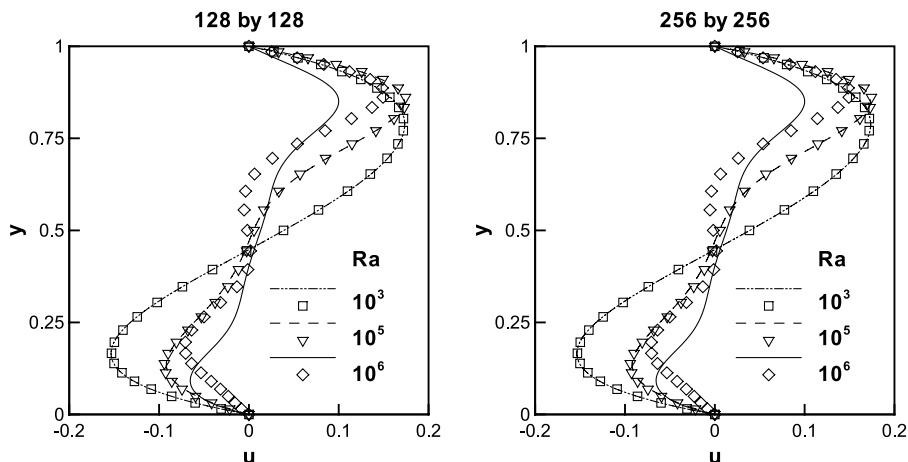
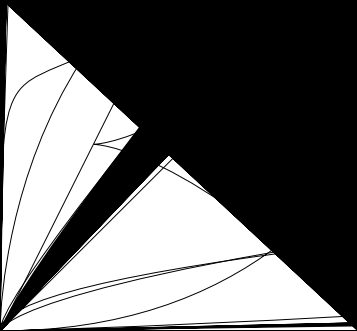


Fig. 6. Profiles of  $u$  on vertical centerline for buoyancy-driven cavity on two uniform grids at three Rayleigh numbers. Curves are present results; symbols are from Yu et al. [21] on a  $129 \times 129$  grid. See text for discussion of disagreement at  $Ra = 10^6$ .



grid  $L_2R^6$  vs. work units are shown in Fig. 8 for all three smoothers. As with the driven cavity, LBGS and BILU show comparable performance, while LUSGS rapidly degrades with increasing grid stretching.

### 8.3. Backward-facing step

We consider isothermal flow over a backward-facing step with a step height of half the downstream channel height and a parabolic inlet velocity  $u(y) = 24y(0.5 - y)$  for  $0 \leq y \leq 0.5$  specified at the step. The outflow boundary is taken at 15 channel heights downstream of the step. Gartling [22] set up this problem as a test for outflow boundary conditions. Numerical results were obtained for  $Re = 800$  with the outflow boundary at 30 channel heights, and used a Galerkin-based finite element method with  $800 \times 40$  biquadratic velocity elements and linear discontinuous pressure elements. For this flow the residual  $R^k$  is evaluated with third-order upwind differences in the  $y$ -direction on all grids while in the  $x$ -direction we use first-order upwind with defect correction to third-order on the fine grid only. Streamfunction contours, computed on a uniform  $480 \times 64$  grid, for  $Re = 400$  and  $800$ , are shown in Fig. 9. Note the contour levels are the same as those used by Gartling [22]. Profiles of  $u$  computed on uniform  $480 \times 64$  and  $960 \times 128$  grids at  $x = 7$  and  $15$  channel heights for  $Re = 800$  are compared with Gartling's [22] results in Fig. 10. The agreement at both locations for both grids is excellent. Note that  $x = 15$  channel heights is the exit boundary for the current computations and the value of  $u$  is obtained by linear extrapolation from the interior.

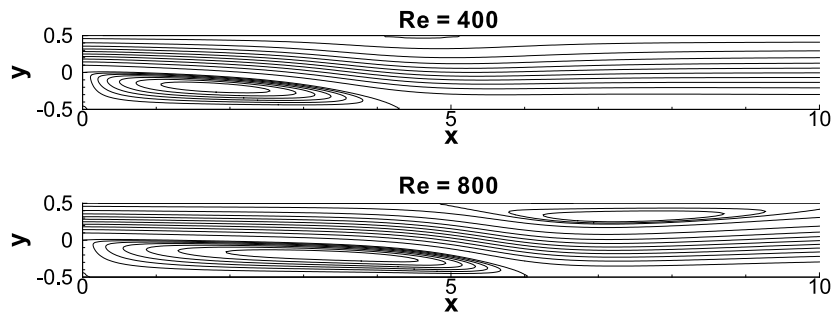


Fig. 9. Streamfunction contours for backward-facing step at  $Re = 400$  and  $800$ .

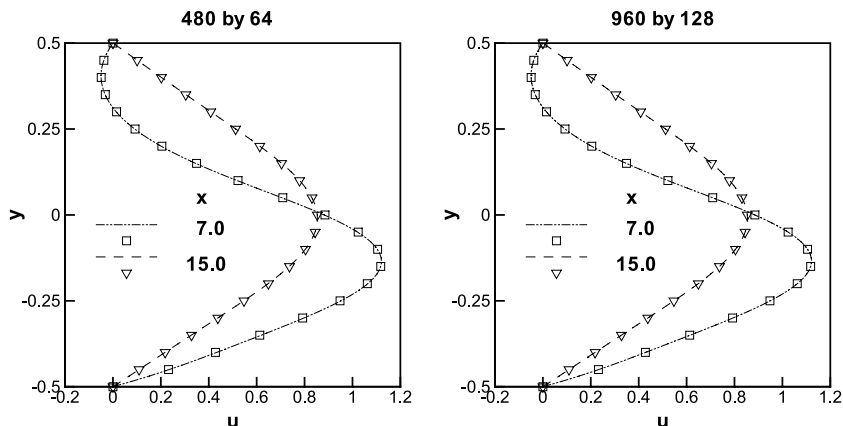


Fig. 10. Profiles of  $u$  at  $x = 7$  and  $15$  for backward-facing step on two uniform grids at  $Re = 800$ . Curves are present results; symbols are from Gartling [22].

The first set of performance results is for a uniform  $480 \times 64$  grid with  $Re$  varying from 200 to 800. These calculations use full coarsening with fine-grid defect correction in the  $x$ -direction. Defect corrections are performed every 2–6 fine-grid sweeps. Five grids are used in the multigrid process with computation starting on grid two. Computations are performed with  $\beta_0^2 = 0.8$ ,  $RVC = 0$  (no viscous correction to  $\beta^2$ ) and  $CFL = 80.0$  for each method on all grids. Convergence plots of fine grid  $L_2R^5$  vs. work units are shown in Fig. 11 for all three smoothers. Note the zig-zag behavior of the plots is a result of an increase in  $L_2R^5$  each time a new fine-grid defect correction is performed. Here one work unit is 0.17 s. All three methods show similar behavior for this case. It is also obvious, however, that the performance of all three methods suffers serious deterioration as Reynolds number is increased to 600 and 800. For  $Re = 600$  or 800, the deterioration with increasing grid stretching is even worse.

In order to obtain a more robust method we introduce semi-coarsening and retain fine-grid defect correction, both in  $x$ . Again we use a uniform  $480 \times 64$  grid with five grid levels and start the computation on grid two. For these computations, defect corrections were performed every two fine grid sweeps. Computations are performed with  $\beta_0^2 = 0.8$  for LBGS and BILU; however, LUSGS performs better in semi-coarsening for  $\beta_0^2 = 1.8$ . Again we set  $RVC = 0$  and  $CFL = 80.0$  for each method on all grids. Convergence plots of fine grid  $L_2R^5$  vs. work units for these methods are shown in Fig. 12. With semi-coarsening, LBGS and BILU show similar performance, but LUSGS take about twice as long to converge. Note that on a finer grid,  $960 \times 128$ , the relative performance of LUSGS with semi-coarsening is even worse. The deterioration of performance with Reynolds number, however, is now quite modest. Of course, as expected, the total number of work units is larger. Note the zig-zag behavior, when it occurs, is much less pronounced as the fine-grid defect correction is now much smaller.

The next set of results is for a stretched  $480 \times 64$  grid at  $Re = 800$ . Stretching is only in the  $y$ -direction with the grid spacing at the bottom and top walls varying from  $ds_0 = 0.004$  to 0.00004 while the spacing at the center (top of the step) is held fixed at  $ds_c = 0.008$ . Spacing in the  $x$ -direction is kept uniform. As above we use semi-coarsening with defect corrections, both in  $x$ , every two fine-grid sweeps. Values of  $\beta_0^2$ ,  $RVC$  and  $CFL$  are the same as in the previous set. Convergence plots of fine grid  $L_2R^5$  vs. work units for these methods are shown in Fig. 13. Again LBGS and BILU are similar, and LUSGS is much worse. What is most interesting is the insensitivity of these performance results to grid stretching. This is certainly desirable

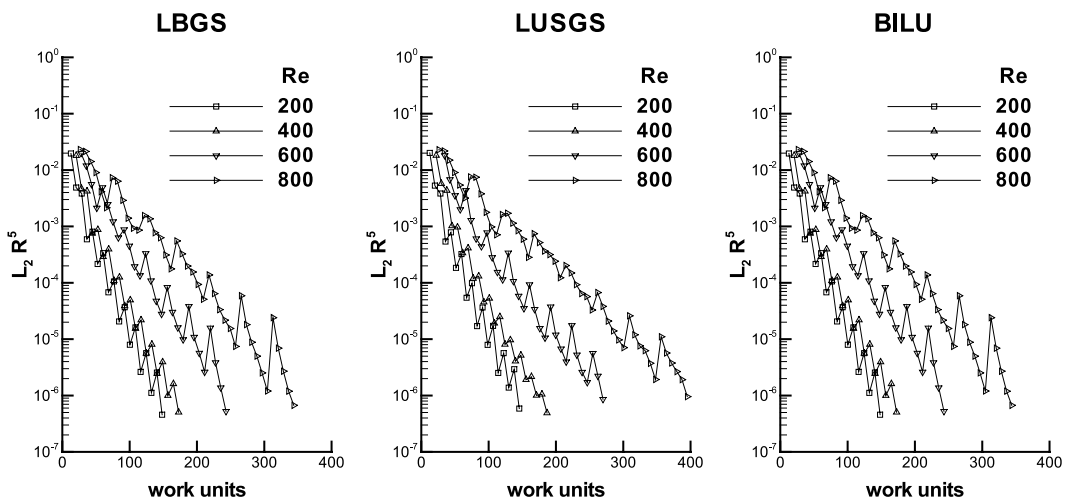


Fig. 11. Backward-facing step: convergence plots for each method on a uniform  $480 \times 64$  grid with full coarsening and defect correction in  $x$ .





temperature and an adiabatic wall condition is imposed on the upper channel wall and the step face. Again viscosity  $\mu$  and thermal conductivity  $\kappa$  are held constant. For this flow as before the residual  $R^k$  is evaluated with third-order upwind differences in the  $y$ -direction on all grids while in the  $x$ -direction we use first-order upwind with defect correction to third-order on the fine grid only. Streamfunction contours, computed on a uniform  $480 \times 64$  grid, for  $Re = 400$  and  $800$ , are shown in Fig. 14. Note that at  $Re = 800$  the lower recirculating zone is larger and the upper recirculating zone is no longer present.

The first set of performance results is for a uniform  $480 \times 64$  grid with  $Re$  varying from 200 to 800. These calculations use full coarsening with fine-grid defect correction in the  $x$ -direction every 2 fine-grid sweeps. Five grids are used in the multigrid process with computation starting on grid two. Computations are performed with  $\beta_0^2 = 0.6$ , RCV = 0 and CFL = 80.0 for each method on all grids. Convergence plots of fine grid  $L_2 R^5$  vs. work units are shown in Fig. 15 for all three smoothers. Here one work unit is 0.17 s. In this case all methods show comparable performance with negligible deterioration for increasing Reynolds number.

Next we apply the same methods on a stretched  $480 \times 64$  grid at  $Re = 800$ . Stretching is only in the  $y$ -direction with grid spacing at the bottom and top walls varying from  $ds_0 = 0.004$  to 0.00004 while grid

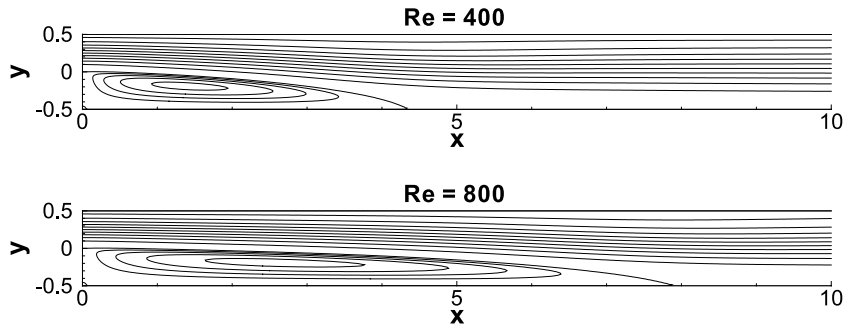


Fig. 14. Streamfunction contours for backward-facing step with heat transfer at  $Re = 400$  and  $800$ .

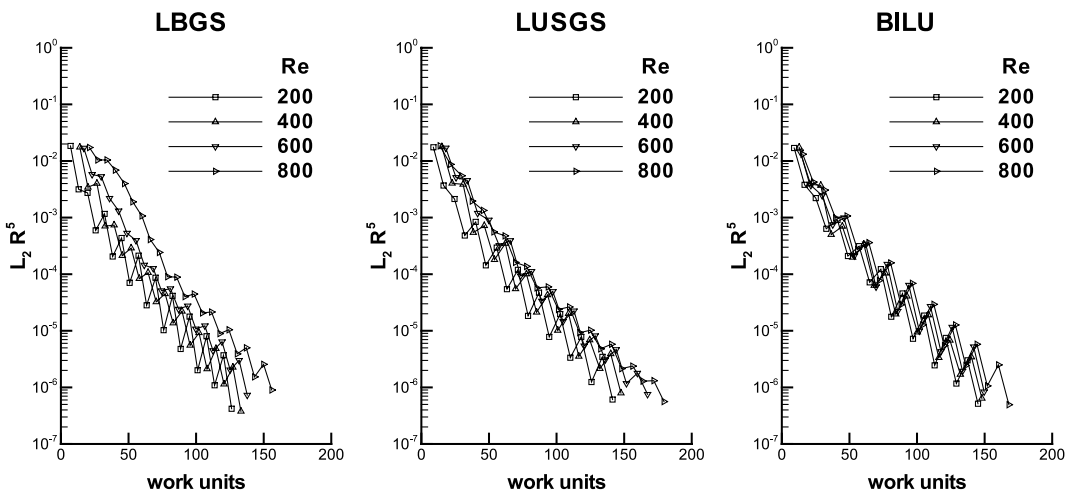


Fig. 15. Backward-facing step with heat transfer: convergence plots for each method on a uniform  $480 \times 64$  grid with full coarsening and defect correction in  $x$ .

spacing at the center is held fixed at  $ds_c = 0$ .

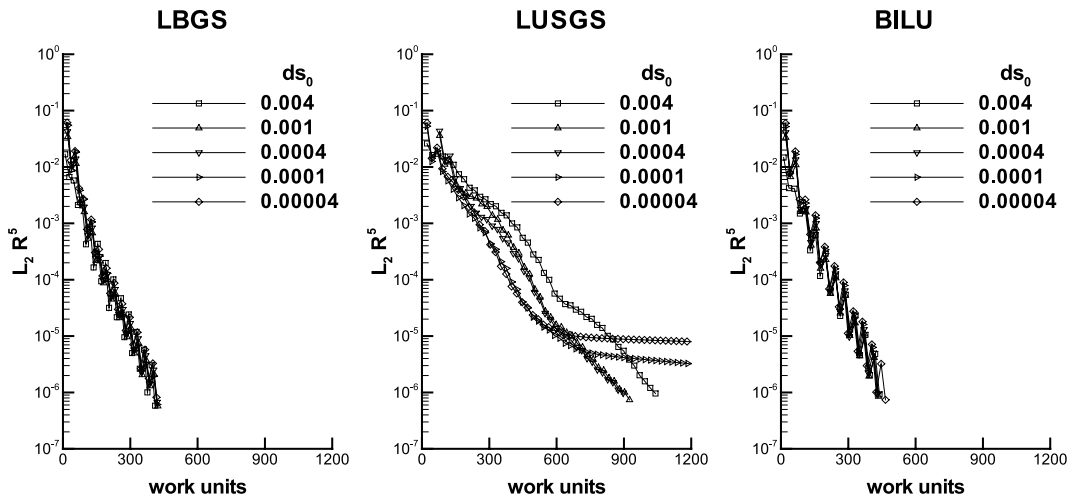


Fig. 18. Backward-facing step with heat transfer: convergence plots for each method on a stretched  $480 \times 64$  grid with semi-coarsening and defect correction in  $x$  at  $Re = 800$ .

in Fig. 17 for all three smoothers. Here LBGS and BILU give comparable performance while LUSGS is much worse.

Finally, we consider stretched grid performance with semi-coarsening and defect correction in  $x$ . Values of  $\beta_0^2$ , RCV and CFL are the same as in the previous set. Convergence plots of fine grid  $L_2 R^5$  vs. work units are shown in Fig. 18 for all three smoothers. Again LBGS and BILU give comparable performance while LUSGS is much worse with convergence almost ceasing at the two finest wall grid spacings.

### 9. Conclusions

The performance of multigrid solvers for the compressible Navier–Stokes equations at low speeds and large temperature variations has been investigated. Each solver employed time-derivative preconditioning together with preconditioned flux-differencing splitting of the inviscid terms. Three implicit smoothers were incorporated in a common multigrid procedure and the resulting methods were tested on four 2D laminar problems over a range of Reynolds numbers with both uniform and highly stretched grids.

For predominantly recirculating flows on uniform grids, the performance of all three methods is comparable and really quite good although a moderate slowing of convergence is seen with increasing Reynolds number. On highly stretched grids, however, the convergence rate of LUSGS deteriorates quite rapidly while the performance of LBGS and BILU remains insensitive to the stretching. It should be noted that in the case of the buoyancy-driven cavity flow none of the methods has any difficulty with the four to one ratio of wall temperatures.

For cases with a predominant flow direction, it appears that a robust method can be constructed on the basis of semi-coarsening and fine grid defect correction, both in the main flow direction. Such methods cost more per multigrid cycle but tend to converge more reliably and in fewer cycles. With semi-coarsening LBGS and BILU show comparable good performance on both uniform and highly stretched grids again with moderate slowing of convergence at the higher Reynolds numbers. The performance of LUSGS with semi-coarsening, on the other hand, degrades substantially on uniform grids and may get even worse on highly stretched grids. In the case of the backward-facing step with heat transfer, we again note that none of the methods has any difficulty with the two to one ratio of bottom wall to inlet temperatures.

From the present studies it is evident that an efficient robust multigrid solver can be based on either LBGS or BILU with either full or semi-coarsening, depending on the problem. It should be noted that BILU tends to be the more robust of the two showing a net decrease in cpu time at the more difficult higher Reynolds number cases albeit at a cost of a significant increase in global storage. Finally, we also note the benefit to be obtained from the use of directional defect correction in the case of a predominant flow direction.

## Appendix A

The block matrix coefficients in Eq. (24) are obtained by expansion of Eq. (19). For an interior cell these are given as follows:

$$\begin{aligned}
 \bar{D}_{i,j} &= - \left[ \hat{A}_{i-1/2,j}^+ + R_{i-1/2,j}^{xx} dx_{i-1/2}^{-1} \right] dy_j, \\
 \bar{A}_{i,j} &= - \left[ \hat{B}_{i,j-1/2}^+ + R_{i,j-1/2}^{yy} dy_{j-1/2}^{-1} \right] dx_i, \\
 \bar{C}_{i,j} &= + \left[ \hat{B}_{i,j+1/2}^- - R_{i,j+1/2}^{yy} dy_{j+1/2}^{-1} \right] dx_i, \\
 \bar{E}_{i,j} &= + \left[ \hat{A}_{i+1/2,j}^- - R_{i+1/2,j}^{xx} dx_{i+1/2}^{-1} \right] dy_j, \\
 \bar{B}_{i,j} &= + \left[ \hat{A}_{i+1/2,j}^+ - \hat{A}_{i-1/2,j}^- + R_{i+1/2,j}^{xx} dx_{i+1/2}^{-1} + R_{i-1/2,j}^{xx} dx_{i-1/2}^{-1} \right] dy_j \\
 &\quad + \left[ \hat{B}_{i,j+1/2}^+ - \hat{B}_{i,j-1/2}^- + R_{i,j+1/2}^{yy} dy_{j+1/2}^{-1} + R_{i,j-1/2}^{yy} dy_{j-1/2}^{-1} \right] dx_i \\
 &\quad + \left( P_Q^{-1} \right)_{i,j} dx_i dy_j / dt - D_{i,j} dx_i dy_j.
 \end{aligned} \tag{A.1}$$

When the cell  $(i, j)$  is located adjacent to a boundary, as noted in Section 4, the inviscid surface flux is obtained directly from Eq. (3) and the viscous flux is approximated by a one-sided second-order difference. As a result the above matrix coefficients must be modified. As an example, let the cell face  $(i - 1/2, j)$  lie on the left boundary. Then in Eq. (A.1) the coefficients  $\bar{A}_{i,j}$ ,  $\bar{C}_{i,j}$  and  $\bar{E}_{i,j}$  are unchanged,  $\bar{D}_{i,j}$  is set to zero, and  $\bar{B}_{i,j}$  is changed as follows:

$$\begin{aligned}
 \bar{B}_{i,j} &= + \left[ \hat{A}_{i+1/2,j}^+ + R_{i+1/2,j}^{xx} dx_{i+1/2}^{-1} + 3R_{i-1/2,j}^{xx} dx_{i-1/2}^{-1} \right] dy_j - \left[ A_{i-1/2,j} + 3R_{i-1/2,j}^{xx} dx_{i-1/2}^{-1} \right] dy_j L_b \\
 &\quad + \left[ \hat{B}_{i,j+1/2}^+ - \hat{B}_{i,j-1/2}^- + R_{i,j+1/2}^{yy} dy_{j+1/2}^{-1} + R_{i,j-1/2}^{yy} dy_{j-1/2}^{-1} \right] dx_i \\
 &\quad + \left( P_Q^{-1} \right)_{i,j} dx_i dy_j / dt - D_{i,j} dx_i dy_j.
 \end{aligned} \tag{A.2}$$

## Appendix B

We have employed a mesh clustering procedure that permits specification of the grid spacing at both ends of the interval. Let  $s_j$ ,  $j = 0$  to  $n$ , denote the grid points along a line with the locations  $s_0, s_1, s_{n-1}, s_n$  specified. Also define  $ds_j \equiv (s_{j+1} - s_{j-1})/2$ . Then we take

$$\begin{aligned}
 -d_j^1 &\equiv -s_{j-1} + 2s_j - s_{j+1} - p_j ds_j = 0 \quad \text{for } j = 1, \dots, n, \\
 -d_j^2 &\equiv \begin{cases} s_j - s_1 = 0 & \text{for } j = 1, \\ -p_{j-1} + 2p_j - p_{j+1} = 0 & \text{for } j = 2, \dots, n-1, \\ s_j - s_{n-1} = 0 & \text{for } j = n-1. \end{cases}
 \end{aligned} \tag{B.1}$$

Eq. (B.1) are linearized about the current state  $Q_j^k$ , where

$$Q_j^k = (s_j, p_j)^T \quad \text{with } \Delta Q_j = Q_j^{k+1} - Q_j^k \text{ and } \Delta Q_0 = \Delta Q_n.$$

Then

$$-A_j \Delta Q_{j-1} + B_j \Delta Q_j - C_j \Delta Q_{j+1} = D_j \quad (\text{B.2})$$

with

$$\begin{aligned} A_j &= \begin{bmatrix} 1 - p_j/2 & 0 \\ 0 & 0 \end{bmatrix}, & B_j &= \begin{bmatrix} 2 & -ds_j \\ 1 & 0 \end{bmatrix}, & C_j &= \begin{bmatrix} 1 + p_j/2 & 0 \\ 0 & 0 \end{bmatrix} & \text{for } j = 1, \\ A_j &= \begin{bmatrix} 1 - p_j/2 & 0 \\ 0 & 1 \end{bmatrix}, & B_j &= \begin{bmatrix} 2 & -ds_j \\ 0 & 2 \end{bmatrix}, & C_j &= \begin{bmatrix} 1 + p_j/2 & 0 \\ 0 & 1 \end{bmatrix} & \text{for } j = 2, \dots, n-2, \\ A_j &= \begin{bmatrix} 1 - p_j/2 & 0 \\ 0 & 0 \end{bmatrix}, & B_j &= \begin{bmatrix} 2 & -ds_j \\ 1 & 0 \end{bmatrix}, & C_j &= \begin{bmatrix} 1 + p_j/2 & 0 \\ 0 & 0 \end{bmatrix} & \text{for } j = n-1, \\ D_j &= \begin{bmatrix} d_j^1 \\ d_j^2 \end{bmatrix}. \end{aligned} \quad (\text{B.3})$$

$Q_j^0$  is initialized by distributing  $s_j$  uniformly between  $s_0$  and  $s_n$  and setting  $p_j = 0$  for all  $j$ . Then Eq. (B.2) is solved for  $\Delta Q_j$  by block-tridiagonal inversion and  $Q_j^k$  is updated. The process continues until the  $L_2$  norm of  $\Delta p_j$  is reduced to less than  $10^{-6}$ , typically 5–15 iterations. For a highly stretched case, the resulting grid spacing increases rapidly but smoothly from the beginning of the interval and then levels off to become nearly constant over the center of the interval.

## References

- [1] A. Brandt, Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics, The Weizmann Institute of Science, Rehovot, Israel, 1984.
- [2] A. Brandt, I. Yavneh, On multigrid solution of high-Reynolds incompressible entering flows, *J. Comput. Phys.* 101 (1992) 151.
- [3] J.L. Thomas, B. Diskin, A. Brandt, Textbook multigrid efficiency for the incompressible Navier–Stokes equations: high Reynolds number wakes and boundary layers, *Comput. Fluids* 30 (2001) 853.
- [4] T.W. Roberts, R.C. Swanson, D. Sidlikover, An algorithm for ideal multigrid convergence for the steady Euler equations, *Comput. Fluids* 28 (1999) 427.
- [5] T.W. Roberts, D. Sidlikover, S.V. Tsynkov, On the combined performance of nonlocal artificial boundary conditions with the new generation of advanced multigrid flow solvers, *Comput. Fluids* 31 (2002) 269.
- [6] N.A. Pierce, M.B. Giles, Preconditioned multigrid methods for compressible flow calculations on stretched meshes, *J. Comput. Phys.* 136 (1997) 425.
- [7] J. Steelant, E. Dick, S. Pattijn, Analysis of robust multigrid methods for steady viscous low Mach number flows, *J. Comput. Phys.* 136 (1997) 603.
- [8] J.R. Amaladas, H. Kamath, Implicit and multigrid procedures for steady-state computations with upwind algorithms, *Comput. Fluids* 28 (1999) 187.
- [9] D. Drikakis, O.P. Iliev, D.P. Vassileva, A nonlinear multigrid method for the three-dimensional incompressible Navier–Stokes equations, *J. Comput. Phys.* 146 (1998) 302.
- [10] L. Yuan, Comparison of implicit multigrid schemes for three-dimensional incompressible flows, *J. Comput. Phys.* 177 (2002) 134.
- [11] R.S. Montero, I.M. Llorente, M.D. Salas, Robust multigrid algorithms for the Navier–Stokes equations, *J. Comput. Phys.* 173 (2001) 412.
- [12] B. Epstein, A. Averbuch, I. Yavneh, An accurate ENO driven multigrid method applied to 3D turbulent transonic flows, *J. Comput. Phys.* 168 (2001) 316.
- [13] S.G. Sheffer, L. Martinelli, A. Jameson, An efficient multigrid algorithm for compressible reactive flows, *J. Comput. Phys.* 144 (1998) 484.

- [14] P. Gerlinger, P. Stoll, D. Bruggemann, An implicit multigrid method for the simulation of chemically reacting flows, *J. Comput. Phys.* 146 (1998) 322.
- [15] D.A. Caughey, Implicit multigrid computation of unsteady flows past cylinders of square cross-section, *Comput. Fluids* 30 (2001) 939.
- [16] E. Turkel, Preconditioned methods for solving the incompressible and low speed compressible equations, *J. Comput. Phys.* 72 (1987) 277.
- [17] Y.H. Choi, C.L. Merkle, The application of preconditioning in viscous flows, *J. Comput. Phys.* 105 (1993) 207.
- [18] B. van Leer, T.W. Lee, P. Roe, Characteristic time-stepping or local preconditioning of the Euler equations, AIAA paper 91-1552-CP, 1991.
- [19] B. van Leer, Toward the ultimate conservative difference scheme. V. A second order sequel to Godunov's method, *J. Comput. Phys.* 32 (1979) 101.
- [20] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, *J. Comput. Phys.* 48 (1982) 387.
- [21] S.T. Yu, B.N. Jiang, J. Wu, N.S. Liu, A div–curl–grad formulation for compressible buoyant flows solved by the least-squares finite element method, *Comput. Methods Appl. Mech. Eng.* 137 (1996) 59.
- [22] D.K. Gartling, A test problem for outflow boundary conditions – flow over a backward-facing step, *Int. J. Numer. Methods Fluids* 11 (1990) 953.