S©MR ISSN 1813-3304

СИБИРСКИЕ ЭЛЕКТРОННЫЕ
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ

Siberian Electronic Mathematical Reports
http://semr.math.nsc.ru

Том 17, стр. 2190–2203 (2020) УДК 519.6
DOI 10.33048/semi.2019.17.145 MSC 65F10,65F50

MULTIGRID METHODS WITH PSTS- AND HSS-SMOOTHERS FOR SOLVING THE UNSTEADY NAVIER-STOKES EQUATIONS

G.V. MURATOVA, T.S. MARTYNOVA, E.M. ANDREEVA, V.V. BAVIN, Z.-Q. WANG

Abstract. Multigrid methods are considered for staggered grid discretizations of the incompressible unsteady Navier-Stokes equations. After discretization and linearization of the problem, systems of linear algebraic equations with a strongly nonsymmetric matrix appear. Product-type skew-Hermitian triangular splitting and Hermitian/skew-Hermitian splitting methods are used as smoothers in the multigrid methods for solving the linear equation systems. Numerical experiments on a 2-D model problem were carried out using algebraic multigrid methods and indicated that these smoothers are robust with respect to the different Reynolds numbers.

Keywords: multigrid methods, product-type skew-Hermitian triangular splitting methods, Hermitian/skew-Hermitian splitting methods, incompressible Navier-Stokes equations.

1. Introduction

Multigrid methods (MGMs) are successful tools for the solution of the systems of linear algebraic equations (SLAEs) associated with discretization of boundary-value problems. The MGM efficiency depends on the adjustment of its components to the problem to be solved. Important parts of the MGM are a smoothing procedure or basic iterative method and the coarse-grid correction. We suggest to use special iterative methods as smoothers for the MGM and non-standard course-grid correction to a good approximation of the smooth error components. For nonelliptic and

---

Muratova, G.V., Martynova, T.S., Andreeva, E.M., Bavin, V.V., Wang, Z.-Q., Multigrid methods with PSTS- and HSS-smoothers for solving the unsteady Navier-Stokes equations.

Supported by RFBR (grant No. 19-51-53013 GFENa) and the Government of the Russian Federation (Project No. 075-15-2019-1928).

Received November, 11, 2020, published December, 25, 2020.

non-Hermitian problems a strong mathematical theory is not available, and the correct choice of MGM components is far from standard procedure [1, 2].

There exist two approaches in the MGM: geometric multigrid and algebraic multigrid methods (AMG) [3]. In this study, we consider the AMG. Having chosen a relaxation scheme, the crux of the problem is to determine what is meant by smooth error. The considered product-type skew-Hermitian triangular splitting (PSTS) and Hermitian/skew-Hermitian splitting (HSS) relaxations (similar to Gauss-Seidel iterative method) have the property that after making great progress toward convergence, it stalls, and little improvement is made with successive iterations. At this point, we define the error to be "algebraically"smooth and AMG is based on the next concept: $Ae \approx 0$ [4], i.e., smooth error has relatively small residuals (hereinafter, A is the matrix of the SLAE).

There are two coarsening approaches in the AMG: RS (Ruge-Stuben) and PMIS (parallel changes independent set) algorithms. The RS algorithm [3, 4] is a traditional coarsening approach, it is based on two heuristic criteria that achieve optimal convergence and minimal computational cost. The PMIS is based on the same principles as the RS algorithm except that a heuristic criterion is not strictly observed. Unlike the RS coarsening, the PMIS is not sequential [5]. An obvious advantage of the PMIS algorithm is the possibility of natural parallelization, which makes it applicable for 3-D problems, as well as for problems with a huge number of unknowns.

Let us briefly describe the organization of calculations using the MGM in accordance with [1] - [4]. To do this, consider a system of grid equations:

$$A_h u^h = b^h.$$

The interpolation operator P from a coarse grid H to a fine grid h allows the operator A_H to be represented on a coarse grid in the form

$$A_H = RA_hP,$$

where R is the restriction operator. Then solution correction step has the form

$$u^h_{new} = u^h_{old} + Pe^H.$$

The error e^H is the exact solution of the equation

$$A_H e^H = r^H,$$

where $r^H = Rr^h$; $r^h = b^h - A_h u^h_{old}$. In this case, before and after correction of the solution, μ_1 pre-smoothing and μ_2 post-smoothing steps are made using some iteration methods.

We consider two methods based on the Hermitian and skew-Hermitian splitting of the matrix A of the SLAE: PSTS iteration method [6, 7] and HSS iteration method [8]. These methods are used as smoothers for the MGM.

There are several different ways to prove the convergence of the MGM, depending on the assumptions we make. Note that these assumptions, as a rule, are rather difficult to verify in real problems. Usually, the smoothing and approximation properties are used to prove convergence [1, 2]. Based on these properties convergence of the MGM with PSTS smoothers has been proved. Numerical experiments were carried out using the AMG (PMIS algorithm) with PSTS-, HSS- and Gauss-Seidel-based smoothers.

2. Model problem

We are interested in numerical solution of the SLAEs that arise from finite difference approximation of the 2-D incompressible unsteady Navier-Stokes equations governing the flow of viscous Newtonian fluids. The primitive variables mathematical formulation of this problem is: find $\mathbf{V}$ and P (up to constant) so that

$$\frac{\partial \mathbf{V}}{\partial t} - \nu\Delta\mathbf{V} + (\mathbf{V}\cdot\nabla)\mathbf{V} + \nabla P = \mathbf{f} \quad on \quad \Omega\times(0,T],$$

$$div\mathbf{V} = 0 \quad on \quad \Omega\times[0,T],$$

$$\mathbf{V} = \mathbf{g} \quad on \quad \partial\Omega\times[0,T],$$

$$\mathbf{V}(\mathbf{x},0) = \mathbf{V}_0(\mathbf{x}) \quad on \quad \Omega$$

where ν is the kinematic viscosity coefficient; Δ is the Laplacian, ∇ is the gradient, div is the divergence; $\mathbf{V} = (w(x,y,t), v(x,y,t))$ is the velocity vector, P is the pressure, $\mathbf{g}$ is the velocity prescribed on the boundary $\partial\Omega$ and $\mathbf{V}_0$ is the divergence-free initial velocity field. We assume that the fluid motion occurs in the time interval $[0,T]$.

One of the main difficulties in using the AMG for computational fluid dynamics is the constructing smoothers that are robust over a wide range of the viscosity coefficient, especially for small values of ν. The difficulty of smoothing for the Navier-Stokes equations is pointed out in [9]. Classical MGMs were developed for elliptic partial differential equations. When applied to nonelliptic and singular perturbation problems such as high-Reynolds flows, performance seems to deteriorate significantly, because the solution becomes more complex.

As a model problem, we consider the standard lid-driven cavity problem [10] in the square domain $\Omega = (0,1)\times(0,1)$. Let us write this problem in scalar dimensionless form

$$\frac{\partial w}{\partial t} + w\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + \frac{\partial P}{\partial x} - \frac{1}{Re}\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2}\right) = 0, \tag{1}$$

$$\frac{\partial v}{\partial t} + w\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{\partial P}{\partial y} - \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) = 0, \tag{2}$$

$$\frac{\partial w}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{3}$$

$$\begin{gathered} w = 0, v = 0, \quad (x = 0, x = 1, y = 0), \\ w = 1, v = 0, \quad (y = 1), \\ w\,(x,y,0) = 0, \quad v\,(x,y,0) = 0. \end{gathered} \tag{4}$$

where Re is the Reynolds number ($Re = UL/\nu$, U is a characteristic velocity of the flow, L is a characteristic length scale). Homogeneous Dirichlet boundary conditions are prescribed for all velocity components with the exception of a positive unit horizontal velocity along the top face. The initial condition assumes the fluid to be at rest. We consider this problem as a test for study the effect of the suggested smoothers for the MGM.

The fully implicit scheme has been used for the time discretization. The use of implicit schemes removes restrictions on the time integration step, which is selected based on the required computational accuracy. We fix the time step δt and introduce a discrete time grid $t^n = n\delta t$, $n \geq 0$:

$$\frac{1}{\delta t}(\mathbf{V}^{(n+1)} - \mathbf{V}^{(n)}) + \nabla P^{(n+1)} = -(\mathbf{V}^{(n+1)} \cdot \nabla)\mathbf{V}^{(n+1)} + \frac{1}{Re}\Delta \mathbf{V}^{(n+1)}, \tag{5}$$

$$div\mathbf{V}^{(n+1)} = 0, \tag{6}$$

The boundary conditions are taken at $t = t^{n+1}$:

$$w^{(n+1)} = 0, \quad v^{(n+1)} = 0, \quad (x = 0, x = 1, y = 0),$$
$$w^{(n+1)} = 1, \quad v^{(n+1)} = 0, \quad (y = 1).$$

The uniform grid Ω is introduced in the domain Ω with steps h_1 and h_2; $h_1 = 1/N_1$, $h_2 = 1/N_2$, where N_1, N_2 are the number of cells in each direction. The grid cells are positioned such that the cell faces coincide with the boundary $\partial\Omega$ of Ω. The spatial discretization of the Navier-Stokes equations is performed on MAC (Marker-and-Cell) [11] (staggered) grids when pressure and velocities in two-dimensional problems are determined on three grids shifted relative to each other. So, the pressure P is located in the center of each cell, the x-component velocity w is on the middle points of vertical faces, the y-component velocity v is on the middle points of horizontal faces. The continuity equation (6) is discretized at cell centers using central difference schemes.

Thus we introduce the grid sets and the corresponding spaces:

$$\overline{D}_1 = \{x_{ij} = ((i + 1/2)h_1, jh_2) : i = 0, ..., N_1 - 1, j = 0, ..., N_2\},$$
$$\overline{D}_2 = \{x_{ij} = (ih_1, (j + 1/2)h_2) : i = 0, ..., N_1, j = 0, ..., N_2 - 1\},$$
$$D_3 = \{x_{ij} = (ih_1, jh_2) : i = 1, ..., N_1 - 1, j = 1, ..., N_2 - 1\}.$$

Let $\mathbf{V}_h = V_{1,h} \times V_{2,h}$ be the linear space of vector functions defined on $\overline{D}_1 \times \overline{D}_2$ and vanishing at the corresponding grid boundaries, and P_h is the space of functions defined on D_3 and orthogonal to unity. Thus,

$$V_{1,h} = \{w_{ij} = w(x_{ij}) : x_{ij} \in \overline{D}_1, w_{0,j} = w_{N_1-1,j} = w_{i,0} = w_{i,N_2} = 0\},$$
$$V_{2,h} = \{v_{ij} = v(x_{ij}) : x_{ij} \in \overline{D}_2, v_{0,j} = v_{N_1,j} = v_{i,0} = v_{i,N_2-1} = 0\},$$
$$P_h = \{P_{ij} = P(x_{ij}) : x_{ij} \in D_3, \sum_{ij} h_1h_2P_{ij} = 0\}.$$

Variables are denoted by a single set of indices, despite the fact that different variables are calculated at different grid nodes. As a result, the indices i, j refer to a set of three mismatched points.

The equations contain the discrete nonlinear terms. For treating this non-linearity Newton linearization around the old time level is used. If we want to linearize a nonlinear term $w^{(n+1)}\phi_x^{(n+1)}$, then

$$w^{(n+1)}\phi_x^{(n+1)} = w^{(n)}\phi_x^{(n+1)} + w^{(n+1)}\phi_x^{(n)} - w^{(n)}\phi_x^{(n)} + O(\delta t^2). \tag{7}$$

The expression in the right-hand side of (7) is linear in the variables at the new time level and possesses a discretization error $O(\delta t^2)$. A detailed description of the used computational scheme can be found in [12].

After discretization and linearization of the problem (1)-(4) we need to solve large sparse strongly nonsymmetric SLAEs at each time step. The choice of an appropriate iterative method and its implementation largely determine the overall efficiency of the computational algorithm. In practice, classical iterative methods are used, such as the methods of bi-conjugate gradients, the method of generalized minimal residuals, and others. We use the AMG with the special PSTS- and HSS-smoothers.

3. Smoothers based on the HSS and the PSTS iteration methods

Let us consider iterative solution of the large sparse SLAE

$$Au = b, \quad u, b \in \mathbb{C}^n, \tag{8}$$

where $A \in \mathbb{C}^{n\times n}$ is a non-Hermitian and positive definite matrix.

Naturally, the matrix A can be split into its Hermitian and skew-Hermitian parts as

$$A = A_0 + A_1, \tag{9}$$

where

$$A_0 = \frac{1}{2}(A + A^*), \quad A_1 = \frac{1}{2}(A - A^*), \tag{10}$$

and A^* denotes the conjugate transpose of the matrix A. Positive definiteness of the matrix A means that for all $x \in \mathbb{C}^n \setminus \{0\}$, $x^* A_0 x > 0$. Here x^* denotes the conjugate transpose of the complex vector x. When A is positive definite then its Hermitian part A_0 is Hermitian positive definite, and the diagonal of its skew-Hermitian part A_1 is purely imaginary [8].

Let in some matrix norm $|||\cdot|||$, $|||A_0||| << |||A_1|||$, then the matrix A is called strongly non-Hermitian one. This situation occurs in many real applications, such as the discretization of the Navier-Stokes equations.

The Hermitian and skew-Hermitian splitting (HSS) iteration methods, based on the splitting (9)-(10), for solving large sparse non-Hermitian positive definite SLAE have been proposed in [8]. These methods have been developed in [13, 14] and others.

Then, we can split the skew-Hermitian part A_1 of the matrix $A \in \mathbb{C}^{n\times n}$ into

$$A_1 = K_L + K_U, \tag{11}$$

where K_L and K_U are the strictly lower and the strictly upper triangular parts of A_1, respectively. Obviously, that $K_L = -K_U^*$.

Based on the splittings (9)-(11) the PSTS method has been presented in [6, 7]. It is a generalization of the iteration methods studied previously in [15, 16].

Convergence of the MGM with the skew-Hermitian triangular smoothers for solving convection-diffusion problems has been studied in [17]. In [18] it was shown that the MGMs with the HSS-based smoothers converge uniformly for second-order nonselfadjoint elliptic boundary value problems.

It should be noted that the behavior of the HSS and the PSTS iteration methods is similar to the behavior of the Gauss-Seidel method, which quickly damps the high-frequency harmonics of the error, slowing down in the future. We give the formulas of these iteration methods.

The HSS iteration method [8]: Given an initial guess $u^{(0)}$, for $k = 0, 1, 2, ...$ until $\{u^{(k)}\}$ convergence, compute

$$\begin{cases} (\alpha I + A_0)u^{(k+\frac{1}{2})} = (\alpha I - A_1)u^{(k)} + b, \\ (\alpha I + A_1)u^{(k+1)} = (\alpha I - A_0)u^{(k+\frac{1}{2})} + b, \end{cases}$$

where α is a given positive constant, I is an identity matrix.

We can rewrite the HSS iteration method in the following form:

$$u^{(k+1)} = G(\alpha)u^{(k)} + B(\alpha)^{-1}b,$$

where

$$G(\alpha) = B(\alpha)^{-1}(B(\alpha) - A),$$

and

$$B(\alpha) = \frac{1}{2\alpha}(\alpha I + A_0)(\alpha I + A_1). \tag{12}$$

The PSTS iteration method [6, 7]: Given an initial guess $u^{(0)}$, and two positive parameters ω and τ. For $k = 0, 1, 2, ...$ until $\{u^{(k)}\}$ convergence, compute

$$u^{(k+1)} = G(\omega, \tau)u^{(k)} + \tau B(\omega)^{-1}b, \tag{13}$$

where

$$G(\omega, \tau) = B(\omega)^{-1}(B(\omega) - \tau A),$$

ω and τ are two acceleration parameters, and $B(\omega)$ is defined by

$$B(\omega) = (B_c + \frac{\omega}{2}\widehat{K}_L)B_c^{-1}(B_c + \frac{\omega}{2}\widehat{K}_U), \tag{14}$$

where $\widehat{K}_L = K_L + F_0$, $\widehat{K}_U = K_U - F_0$, $F_0 \in \mathbb{C}^{n\times n}$ is a Hermitian matrix, $B_c \in \mathbb{C}^{n\times n}$ is a prescribed Hermitian positive definite matrix. Obviously that $\widehat{K}_L = -\widehat{K}_U^*$, $A_1 = (K_L + F_0) + (K_U - F_0) = \widehat{K}_L + \widehat{K}_U$.

When $F_0 = 0$ the PSTS method reduces to the iteration method studied in [16]; a special choice of the matrix F_0 allows to improve the convergence of the method. For the PSTS iteration method a convergence analysis, a technique for choosing the optimal parameter and an accelerating procedure have presented in [6, 7].

4. MGM convergence with PSTS smoothers

Let $B_0(\omega)$ and $B_1(\omega)$ be, respectively, the Hermitian and the skew-Hermitian parts of the matrix $B(\omega)$ defined by (14). That is to say,

$$B(\omega) = B_0(\omega) + B_1(\omega), \tag{15}$$

where

$$\begin{cases} B_0(\omega) = \frac{1}{2}(B(\omega) + B(\omega)^T) = B_c + (\frac{\omega}{2})^2\widehat{K}_L B_c^{-1}\widehat{K}_U, \\ B_1(\omega) = \frac{1}{2}(B(\omega) - B(\omega)^T) = \frac{\omega}{2}(\widehat{K}_L + \widehat{K}_U) = \frac{\omega}{2}A_1. \end{cases}$$

Suppose that

$$0 < \alpha_h I \leq A_0 \leq \beta_h I, \quad 0 < \alpha_c I \leq B_c \leq \beta_c I, \tag{16}$$

$$\alpha_l I \leq \widehat{K}_L B_c^{-1}\widehat{K}_U \leq \beta_l I, \quad \alpha_s I \leq B_0(\omega) \leq \beta_s I, \tag{17}$$

where I is an identity matrix. Since A_0 and B_c are Hermitian positive definite matrices, $B_0(\omega)$ and $\widehat{K}_L B_c^{-1}\widehat{K}_U$ are Hermitian matrices, and A_1 is a skew-Hermitian matrix, the bounds α_ψ and β_ψ, $\psi = h, c, l, s$, can be easily expressed with respect to the smallest and the largest eigenvalues or singular values of the corresponding matrix, respectively [6].

We need to require positive definiteness of the matrix $B(\omega)$ from (15). Since the matrix $\widehat{K}_L B_c^{-1}\widehat{K}_U$ is negative definite, then $B_0(\omega) > 0$, if the parameter $\omega \in (0, \omega_{max})$, where $\omega_{max} = 2\sqrt{\left(-\frac{\alpha_c}{\alpha_l}\right)}$.

Notice, that α_l и β_l satisfy the following inequalities:

$$\alpha_s \geq \alpha_c + \left(\frac{\omega}{2}\right)^2 \alpha_l, \quad \beta_s \leq \beta_c + \left(\frac{\omega}{2}\right)^2 \beta_l. \tag{18}$$

Theorem 1. [6] *Let the matrices A and $B(\omega)$ be positive definite. Then the PSTS iteration method is convergent, i.e., the spectral radius $\rho(G(\omega,\tau))$ of its iteration matrix $G(\omega,\tau)$ is less than 1, provided that the acceleration parameters ω and τ satisfy*

$$0 < \tau < \omega, \quad 0 < \omega < \omega_{max},$$

and

$$0 < \tau < \frac{2}{\Theta}, \quad \Theta = \frac{\beta_h}{\alpha_c + (\frac{\omega}{2})^2 \alpha_l}. \tag{19}$$

Let us study the convergence of the MGMs with PSTS-based smoothers.

We use some denotations and theoretical results from [19, 20].

Let $H_1 \subset H_2 \subset ...$ be a family of nested finite-dimensional linear spaces. The dimension of H_m is n_m and the inner product is denoted by $(.,.)_m$ with $\| \; . \; \|_m$ corresponding norm in H_m, $m = 1, 2...$

We are interested in solution of the problem (8) in H_m

$$A_m u = b.$$

Let this problem have a unique solution for some $b \in H_m$ and

$$A_m = \widetilde{A}_m + \widehat{A}_m,$$

where $\widetilde{A}_m$ is a symmetric positive definite operator in H_m. Let Q_m is the other symmetric positive definite operator in H_m and the condition for the spectral radius of the operator $G_m = Q_m^{-1} \widetilde{A}_m$ is satisfied $\rho\left(G_m\right) = 1$.

Using the operators $\widetilde{A}_m$ and Q_m we define the energy norm

$$\|u\|_{\widetilde{A}_m} = (\widetilde{A}_m u, u)^{1/2}, \quad \forall u \in H_m,$$

and the inner product

$$(u, v)_{Q_m} = (Q_m u, v), \quad \forall u, v \in H_m.$$

Then we may define the following norms on H_m :

$$\| u \|_{s,m} = \| u \|_{G_m^s} = (G_m^s u, u)_{Q_m}^{1/2}, s = 0, 1, ...$$
$$\| u \|_{1,m} = (\widetilde{A}_m u, u)^{1/2} = \| u \|_{\widetilde{A}_m},$$
$$\| u \|_{0,m} = (Q_m u, u)^{1/2} = (u, u)_{Q_m}^{1/2}.$$

Define the subspace $F_m \subset H_m$, $F_m = \{u \in H_m : (A_m u, v) = 0, \forall v \in H_{m-1}\}$

Now we make three basic propositions [19, 20]:

Proposition 1. *There exists $v \in H_{m-1}$, γ, $0 < \gamma \leq 1$ and $\delta < \infty$ such that for $\forall \; u \in H_m$*

$$\| u - v \|_{1,m}^2 \leq \delta^\gamma \| u \|_{1+\gamma,m}^2$$

Proposition 2. *There exists $\eta_m, \eta_m \to 0 \quad (m \to \infty)$ such that*

$$| (\widehat{A}_m u, v)_m | < \eta_m \| u \|_{1,m} \| v \|_{1,m}$$

for $\forall \; u \in F_m$, $\forall v \in H_m$, m is a positive integer.

Proposition 3. *There exists* $\mu_m, \mu_m \to 0 \quad (m \to \infty)$ *such that*

$$| \left(\widehat{A}_m u, v \right)_m | \leq \mu_m \parallel u \parallel_{1,m} \parallel v \parallel_{0,m}$$

for $\forall u, v \in H_m$, m *is a positive integer.*

It is assumed that η_m, μ_m are sufficiently small.

In [20] it was shown that propositions 1-3 were satisfied for sufficiently wide class of boundary problems in two dimension-limited domains with different boundary conditions. Proposition 1 means a generalization of the usual approximation property known for the symmetric case. Propositions 2 and 3 restrict the non-symmetric part $\widehat{A}_m$ of the operator A_m.

Let us consider two-grid algorithm. We denote the exact solution of the problem (8) by u^*. Let u^0 is an initial guess, u^1 is a problem solution after smoothing and u^2 is a problem solution after MGM-iteration. We need to estimate the energy-norm of the contraction number defined by

$$\sigma = \sup \frac{\parallel u^2 - u^* \parallel_1}{\parallel u^0 - u^* \parallel_1}, \quad u^0 \neq u^*.$$

Denote the error by $e^i = u^i - u^*, i = 0, 1, 2$.

Theorem 2. [20] *Let the three basic assumptions for the two-grid method be satisfied. Furthermore, let the following smoothing proposition also be satisfied: there exists* Δ $(0 < \Delta < \infty)$ *and* $\vartheta > 0$ *such that*

$$\parallel e^1 \parallel_1^2 + \vartheta \parallel e^1 \parallel_2^2 \leq (1 + \mu\Delta) \parallel e^0 \parallel_1^2 \tag{20}$$

with $\mu{=}\mu_k$ *from Proposition 3. Then* $\sigma \leq \widetilde{\sigma}$ *for the two-grid contraction number where*

$$\widetilde{\sigma} \equiv \widetilde{\sigma}(\epsilon) \equiv \sup \left\{ \begin{array}{c} \left(\frac{\xi^2 + \epsilon^2 \varsigma^2 + 2\epsilon\eta\xi\varsigma}{1 + \delta^{-1}\vartheta (\frac{1-\eta}{1+\eta})^{2/\gamma} \xi^{2/\gamma}} (1 + \mu\Delta) \right)^{1/2} : \\ \xi^2 + \zeta^2 - 2\eta\xi\zeta \leq 1, \ \zeta, \xi \geq 0 \end{array} \right\}$$

$$\xi = \frac{\parallel e^1 + u^* \parallel_1}{\parallel e^1 \parallel_1}, \qquad \varsigma = \frac{\parallel u^* \parallel_1}{\parallel e^1 \parallel_1},$$

where constants ϑ *and* Δ *depend on properties of smoothing system and constants* $\gamma, \delta, \eta, \mu$ *taken from propositions 1, 2, 3 respectively, while* ϵ *is calculation accuracy.*

In [20] the proof of the same theorem for the multigrid method is given. For further considerations it is necessary to have one more theorem from [19].

Theorem 3. [19] *Let the problem (8) be solved by iterative method (13) - (14) written in the form of*

$$u^{(n+1)} = u^{(n)} - \widetilde{B}^{-1} \left(A u^{(n)} - b \right), \qquad \widetilde{A} = A_0,$$

and let the three basic assumptions be satisfied. If there exists the constant $\vartheta > 0$ *so that inequality is satisfied*

$$\widetilde{B} + \widetilde{B}^* - \widetilde{A} \geq \vartheta \left(\widetilde{A} - \widetilde{B} \right)^* Q^{-1} \left(\widetilde{A} - \widetilde{B} \right), \tag{21}$$

then the smoothing assumption (20) is satisfied with

$$\Delta = (1+\vartheta)\,\beta\,(\mu\beta+2)\,, \tag{22}$$

where

$$\beta = \left(\rho\left(\widetilde{B}^{-1}\widetilde{A}\left(\widetilde{B}^{-1}\right)^{*}\widetilde{A}\right)\right)^{1/2}.$$

Now we can prove the convergence of the method.

Theorem 4. *For the method (13) - (14) there exists the constant $\vartheta > 0$ so that inequality (21) is satisfied.*

Proof. In case of using method (13)-(14) $\widetilde{B} = 1/\tau B$, $Q = A_0$. Consider inequality (21) and transform left and right parts of it. Using (15) we obtain

$$\widetilde{B} + \widetilde{B}^{*} - \widetilde{A} = 2/\tau B_0 - A_0, \tag{23}$$

$$\begin{aligned} &\left(\widetilde{A}-\widetilde{B}\right)^{*}\widetilde{A}^{-1}\left(\widetilde{A}-\widetilde{B}\right) = \left(A_0 - \tfrac{1}{\tau}B^{*}\right)A_0^{-1}\left(A_0 - \tfrac{1}{\tau}B\right) = \\ &= A_0 - \tfrac{1}{\tau}B^{*} - \tfrac{1}{\tau}B + \tfrac{1}{\tau^2}B^{*}A_0^{-1}B = A_0 - \tfrac{2}{\tau}B_0 + \tfrac{1}{\tau^2}B^{*}A_0^{-1}B \end{aligned} \tag{24}$$

Denote

$$S = \frac{2}{\tau}B_0 - A_0, \tag{25}$$

then from (16) - (19) it's easy to obtain that

$$\frac{\beta_h}{\alpha_c + (\frac{\omega}{2})^2\alpha_l} < \frac{2}{\tau},$$

thus, $2/\tau B_0 > A_0$ for acceleration parameters ω, τ satisfying the conditions of the Theorem 1, and

$$S = S^{*} > 0.$$

From (23) - (25) we obtain that for inequality (21):

$$S \geq \vartheta\left(-S + \frac{1}{\tau^2}B^{*}A_0^{-1}B\right).$$

Multiplying the left and the right parts of this inequality on $S^{-1/2}$ we obtain

$$I \geq \vartheta\left(-I + \frac{1}{\tau^2}S^{-1/2}B^{*}A_0^{-1}BS^{-1/2}\right). \tag{26}$$

Denote $L = \frac{1}{\tau}A_0^{-1/2}BS^{-1/2}$. Then (26) is transformed to

$$I \geq \vartheta\left(L^{*}L - I\right).$$

If we take

$$\vartheta = \frac{1}{\parallel L^{*}L - I \parallel}, \tag{27}$$

then inequality (21) is satisfied. □

Thus, if the conditions of the Theorem 4 are satisfied then two-grid method converges with the PSTS-smoother (14), ϑ is calculated by (27) and Δ is calculated by (22) correspondingly. The results obtained for the two-grid method can be extended to the multigrid method.

5. Numerical experiments

The AMG (PMIS algorithm) with the PSTS- and the HSS-smoothers is used for solution of the SLAE (8) arising at each time step of solving the model problem (1)-(4). Both of the smoothers are compared with the Gauss-Seidel (GS) one which is the classic and recognized smoothing method [2]. The number of the pre-smoothing and post-smoothing steps are $\mu_1 = \mu_2 = 15$ for all methods. Calculations are carried out using the V-cycle as the least demanding from the computational point of view. In the numerical tests, the time step is equal to 0.01 and the Reynolds number is $Re = \nu^{-1}$. For the PSTS, $B_c = I$ and F_0 was chosen such that the matrix $(F_0 + K_L)$ was unitary [6, 7].

In Table 1 we give the number of the AMG-levels, denoted by m and used in our implementations, computational operator complexities and setup phase times (in parentheses, in seconds) depending on the mesh size and viscosity coefficient values. Operator complexity is defined as the ratio of the total number of the matrix nonzero elements at all grid levels to the number of the matrix nonzero elements on the initial (finest) grid and serves as an indicator of the required memory, pointing to the number of operations per one multigrid cycle at the solution step. Reducing the operational complexity leads to a reduction in the time required for the calculation.

Grid	m	Operator complexities (Setup phase times)				
		$\nu = 10^{-1}$	$\nu = 10^{-2}$	$\nu = 10^{-3}$	$\nu = 10^{-4}$	$\nu = 10^{-5}$
60×60	6	1.52 (4.21)	1.64 (4.52)	1.82 (6.51)	1.83 (6.81)	1.84 (7.51)
120×120	7	1.54 (9.58)	1.65 (9.26)	1.84 (21.49)	1.86 (22.67)	1.85 (21.64)
180×180	8	1.56 (24.35)	1.64 (24.65)	1.85 (36.82)	1.83 (32.21)	1.84 (38.20)
260×260	8	1.57 (36.32)	1.65 (37.20)	1.86 (51.22)	1.91 (52.37)	1.92 (55.29)
520×520	9	1.55 (52.26)	1.84 (55.82)	1.85 (55.84)	1.93 (58.26)	1.94 (65.54)

Table 1. Levels number (m), operator complexities (setup phase times) versus ν for different grids

Some explanations should be given of our choice of the PMIS algorithm (and not the RS) in the AMG. The PMIS has less operator complexities than the RS, which is leading to smaller setup times. The PMIS requires more iterations than the RS, for which the number of iterations is almost constant. The worst convergence of the PMIS compared to the RS is due to the fact that for the PMIS coarsening fine grid structure is not preserved on coarse grids. However, the execution time per iteration of the PMIS is lower due to the less operator complexity, resulting in time savings for the solution phase. In addition, the PMIS provides natural parallelization, in contrast to the RS algorithm, which determines our choice in favor of the first one [21].

In Tables 2, 3, 4 below we compare the AMG-iterations with the HSS-, PSTS- and GS-based smoothers on the different grids for small viscosity values. In our implementations, all iterations are started from the zero vector, and terminated either when

$$\frac{\|r^{(p)}\|_2}{\|r^{(0)}\|_2} \leq 10^{-6},$$

where $r^{(p)} = b - Au^{(p)}$ is the residual vector of the SLAE (8) at the current iterate $u^{(p)}$, and $r^{(0)}$ is the initial residual. Our comparisons are done for the number of iteration steps (denoted by "IT") and the elapsed CPU time (in seconds and denoted by "CPU"). The abbreviation "n.c." in the tables means "no convergence". The experiments are run in MATLAB (version R2018b) with a machine precision 10^{-16}.

ν	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}	
Grid	IT	CPU	IT	CPU	IT	CPU	IT	CPU	IT	CPU
60×60	29	36.81	30	45.22	26	42.84	24	37.65	21	32.51
120×120	38	57.24	37	58.70	35	57.82	30	49.32	22	35.22
180×180	49	83.32	45	81.29	42	86.21	37	81.38	29	57.82
260×260	54	154.87	52	152.57	49	157.22	39	124.61	30	94.37
520×520	65	242.57	64	281.84	57	252.83	45	197.21	37	157.38

TABLE 2. AMG+HSS iterations and CPU with different ν.

ν	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}	
Grid	IT	CPU	IT	CPU	IT	CPU	IT	CPU	IT	CPU
60×60	42	46.82	38	45.70	37	49.24	32	42.36	22	26.82
120×120	49	64.38	45	57.22	39	61.52	35	54.81	24	36.55
180×180	51	97.51	46	94.32	40	87.26	36	82.64	26	54.81
260×260	56	157.84	49	150.57	45	142.81	37	124.83	29	91.29
520×520	61	287.86	59	283.35	57	280.55	49	232.36	40	187.65

TABLE 3. AMG+PSTS iterations and CPU with different ν.

ν	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}	
Grid	IT	CPU	IT	CPU	IT	CPU	IT	CPU	IT	CPU
60×60	44	48.01	54	51.26	64	84.65	n.c.	-	n.c.	-
120×120	59	94.81	67	122.67	83	151.52	n.c.	-	n.c.	-
180×180	64	97.82	82	157.84	94	192.51	n.c.	-	n.c.	-
260×260	n.c.	-	n.c.	-	n.c.	-	n.c.	-	n.c.	-
520×520	n.c.	-	n.c.	-	n.c.	-	n.c.	-	n.c.	-

TABLE 4. AMG+GS iterations and CPU with different ν.

Based on the numerical experiments the following conclusions can be drawn:

1. From the data shown in Tables 2-4, an increase in the number of iterations follows with an increase in the mesh size for all tested methods. This is due to some features of the algebraic approach in the MGM (more precisely, the PMIS algorithm in the AMG). The traditional scalable approach in the AMG (RS algorithm) works well for many 2-D problems arising from PDEs discretization. In this case, the solver can be obtained with the number of iterations that does not depend on the size of the problem. But when using AMG interpolation in combination with the PMIS, the AMG convergence deteriorates depending on the problem size. This result in a loss of scalability [21]. However, the RS algorithm has obvious drawbacks, such as poor efficiency for 3-D problems. When it is applied to 3-D problems, scalability is lost in many cases. The computational operator complexity and size of the stencil can increase significantly, resulting in increased execution time and memory usage [21].

2. From Tables 2-4 it follows that the AMG method with the HSS- and the PSTS- smoothers has fast convergence speed for all tested values of the viscosity coefficient ($\nu = 10^{-1} \div 10^{-5}$) on all used grids, while the AMG with the Gauss–Seidel smoother does not converge for $\nu = 10^{-4}, 10^{-5}$ on all considered grids, and does not converge on the grids 260×260 and 520×520 nodes for all values of the viscosity coefficient. For all tests, the AMG+HSS and AMG+PSTS methods outperform the AMG+GS method with respect to both number of iteration steps and CPU time.

3. The AMG+HSS method slightly ahead of the AMG+PSTS method in most cases, apparently due to the fact that in numerical tests the iterative parameter for the latter one was chosen experimentally and $\omega = \tau = \tau_{exp}$. The decrease in the number of the AMG+HSS and AMG+PSTS iterations with a decrease in the viscosity coefficient values is explained that these methods converge the faster, the more pronounced the nonsymmetry of the SLAE is. The number of iterations and CPU time of the AMG+GS method increases with decreasing ν.

4. The lack of convergence of the AMG+HSS and the AMG+PSTS methods for the smallest values of ν is not observed, which could be expected for the unsteady problem. The obtained SLAE is simpler than for the stationary problem, since the spectrum of this system lies in the right half-plane (as for SLAEs obtained by approximating elliptic equations). This is due to a shift in the spectrum when the time derivative is approximated.

6. Conclusions

Numerical experiments carried out for the 2-D model incompressible unsteady Navier-Stokes problem have shown efficiency of the HSS- and the PSTS- smoothers for the AMG and their advantage over the classical Gauss-Seidel smoother. The use of these smoothers for the AMG allows solving this problem for small values of the viscosity coefficient. It is supposed to solve the corresponding 3-D problem in the future using these methods. In addition, it is proposed to investigate the corresponding stationary problem at small values of the viscosity coefficient.

References

[1] W. Hackbusch, *Multigrid method and application*, Springer-Verlag, Berlin, 1985. Zbl 0595.65106

[2] U. Trottenberg, C. Oosterlee, A. Schuller, *Multigrid*, Academic Press, Orlando, 2001. Zbl 0976.65106

[3] R.D. Falgout, *An introduction to algebraic multigrid*, Computing in Science and Engineering, **8** (2006), 24–33.

[4] K. Stuben, *Algebraic multigrid (AMG): Experiences and comparisons*, Appl. Math. Comput., **13** (1983), 419–451. Zbl 0533.65064

[5] U.M. Yang, *Parallel algebraic multigrid methods – high performance preconditioners*, Lecture Notes in Comp. Science and Eng., **51**, Springer, Berlin, 2006, 209–236. Zbl 1097.65125

[6] L.A. Krukier, T.S. Martynova, Z.-Z. Bai, *Product-type skew-Hermitian triangular splitting iteration methods for strongly non-Hermitian positive definite linear systems*, J. of Comp. and Appl. Math., **232**:1 (2009), 3–16. Zbl 1184.65038

[7] Z.-Z. Bai, L.A. Krukier, T.S. Martynova, *Two-step iterative methods for solving the stationary convection-diffusion equation with a small parameter for the highest derivative on a uniform grid*, Comput. Math. Math. Phys., **46**:2 (2006), 2825–293. Zbl 1210.39018

[8] Z.-Z. Bai, G.H. Golub, M.K. Ng, *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl., **24**:3 (2003), 603–626. Zbl 1036.65032

[9] P. Wesseling, C.W. Oosterlee, *Geometric multigrid with applications to computational fuid dynamics*, J. Comput. Appl. Math., **128**:1-2 (2001), 311–334. Zbl 0989.76069

[10] J.-L. Guermond, C. Migeon, G. Pineau, L. Quartapelle, *Start-up flows in a three-dimensional rectangular driven cavity of aspect ratio 1:1:2 at Re=1000*, J. Fluid Mech., **450** (2002), 169–199. Zbl 1049.76510

[11] S. McKee, M.F. Tome, V.G. Ferreira, J.A. Cuminato, A. Castelo, F.S. Sousa, N. Mangiavacchi, *The MAC method*, Comput. & Fluids, **37**:8 (2008), 907–930. Zbl 1237.76128

[12] G. Muratova, T. Martynova, E. Andreeva, V. Bavin, Z.-Q. Wang, *Numerical solution of the Navier-Stokes equations using multigrid methods with HSS-based and STS-based smoothers*, Symmetry, **12**:2 (2020), 233.

[13] Z.-Z. Bai, *Splitting iteration methods for non-Hermitian positive definite systems of linear equations*, Hokkaido Math. J., **36**:4 (2007), 801–814. Zbl 1138.65027

[14] Z.-Z. Bai, G.H. Golub, M.K. Ng, *On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, Linear Algebra Appl., **428**:2 (2008), 413–440. Zbl 1135.65016

[15] L. Wang, Z.-Z. Bai, *Skew-Hermitian triangular splitting iteration methods for non-Hermitian positive definite linear systems of strong skew-Hermitian parts*, BIT, **44** (2004), 363–386. Zbl 1069.65031

[16] M.A. Botchev, L.A. Krukier, *Iterative solution of strongly nonsymmetric systems of linear algebraic equations*, Comput. Math. Math. Phys., **37**:11 (1997), 1241–1251. Zbl 0946.65018

[17] G.V. Muratova, L.A. Krukier, E.M. Andreeva, *Fourier analysis of multigrid method with triangular skew-symmetric smoothers*, Commun. Appl. Math. Computat., **27**:3 (2013), 355–362. Zbl 1299.65289

[18] Sh. Li, Zh. Huang, *Convergence analysis of HSS-multigrid methods for second-order nonselfadjoint elliptic problems*, BIT, **53**:4 (2013), 987–1012. Zbl 1284.65181

[19] Z. Cao, *Convergence of multigrid methods for nonsymmetric, indefinite problems*, Appl. Math. Comput., **28**;4 (1988), 269–288. Zbl 0661.65037

[20] J. Mandel, *Multigrid convergence for nonsymmetric, indefinite variational problems and one smoothing step*, Appl. Math. Comput., **19** (1986), 201–216. Zbl 0614.65031

[21] H. Sterck, U.M. Yang, J. Heys, *Reducing complexity in parallel algebraic multigrid preconditioners*, SIAM J. Matrix Anal. Appl., **27**:4 (2006), 1019–1039. Zbl 1102.65034

Galina Viktorovna Muratova
Southern Federal University,
Vorovich Institute of Mathematics, Mechanics and Computer Science,
200/1, Bld. 2, Stachki ave.,
Rostov-on-Don, 344090, Russia
Email address: muratova@sfedu.ru

Tatiana Sergeevna Martynova
Southern Federal University,
Vorovich Institute of Mathematics, Mechanics and Computer Science,
200/1, Bld. 2, Stachki ave.,
Rostov-on-Don, 344090, Russia
Email address: martynova@sfedu.ru

Evgeniya Mihailovna Andreeva
Southern Federal University,
Vorovich Institute of Mathematics, Mechanics and Computer Science,
200/1, Bld. 2, Stachki ave.,
Rostov-on-Don, 344090, Russia
Email address: andreeva@sfedu.ru

Vadim Vadimovich Bavin
Southern Federal University,
Vorovich Institute of Mathematics, Mechanics and Computer Science,
200/1, Bld. 2, Stachki ave.,
Rostov-on-Don, 344090, Russia
Email address: tefp9999@mail.ru

Zeng-Qi Wang
Shanghai Jiao Tong University,
School of Mathematical Sciences,
Shanghai, 200000, P.R. China
Email address: wangzengqi@sjtu.edu.cn