Preconditioned iterations for the linearized Navier–Stokes system in rotation form

Maxim A. Olshanskii

Department of Mechanics and Mathematics, Moscow M.V. Lomonosov University, Moscow 119899, Russia

Abstract

The paper studies a performance of a preconditioned iterative method applied to solve the linearized Navier–Stokes system. We try BiCGstab algorithm and discretized the problem by a stabilized finite element method. Essential for building a good preconditioner is the rotation form of the original PDE system. Numerical results demonstrate the robustness of the method.

Keywords: Incompressible Navier–Stokes; Preconditioning; BiCGstab; Saddle point; Rotation form

1. Introduction

A performance of numerical tools for solving large sparse linear algebraic systems of equations strongly depends on properties of a corresponding matrix or operator. If approximate solution of a partial differential equation is a goal, then these properties are determined by the differential equation and a method of discretization. In the paper we consider the incompressible Navier–Stokes equations as the underlying PDE problem. Further we consequentially re-formulate the equations and apply a discretization method, ensuring that physically meaningful discrete solutions are reproduced. Special care is taken to obtain linear algebraic systems, such that reliable solvers can be constructed by a proper preconditioning in conjunction with Krylov-subspace method, like BiCGstab.

Our starting point is the Navier–Stokes equations, which describe a motion of viscous incompressible fluid in some bounded domain $\Omega \in \mathbb{R}^n$, $n = 2,3$:

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega, \\ \operatorname{div}\mathbf{u} &= 0 \quad \text{in } \Omega, \end{aligned} \tag{1}$$

some boundary and initial conditions are supplied. With the objective to build a robust solver based on efficient linear algebra tools, we re-formulate the problem using the formal vector relation

$$(\mathbf{u}\cdot\nabla)\mathbf{u} = (\operatorname{curl}\mathbf{u})\times\mathbf{u} + \tfrac{1}{2}\nabla(\mathbf{u}^2),$$

and obtain the new pressure function $P = p + \frac{1}{2}\nabla(\mathbf{u}^2)$ (Bernoulli pressure) and nonlinear terms in the other form $(\operatorname{curl}\mathbf{u})\times\mathbf{u}$ instead of $(\mathbf{u}\cdot\nabla)\mathbf{u}$. If we solve a steady problem with a fixed-point method or an unsteady one with some time-stepping procedure, we have to solve a linearized problem on every iterative or time step. At this point the principal difference between two formulations of the continuous problem appears from the algorithmic, including numerical linear algebra, point of view: Contrary to original formulation, the new nonlinear terms $(\operatorname{curl}\mathbf{u})\times\mathbf{u}$ give zero order terms in linearized equation, if handled properly. By proper handling we mean that linearized equation gives rise to non-singular linear algebraic system (LAS) and important properties of the original system such as conservation of the energy are preserved — in terms of linear algebra the latter means that the matrix, resulting after linearization and discretization of $(\operatorname{curl}\mathbf{u})\times\mathbf{u}$, is *skew-symmetric*. The corresponding linearized system is

$$\begin{aligned} \alpha\mathbf{u} - \nu\Delta\mathbf{u} + \mathbf{w}\times\mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega, \\ -\operatorname{div}\mathbf{u} &= g \quad \text{in } \Omega, \end{aligned}$$

where $\alpha \geq 0$ and $\mathbf{w} = \operatorname{curl}\mathbf{u}^o$, $\mathbf{u}^o$ is some known approximation to $\mathbf{u}$.

2. Linear solver

Assume the Dirichlet boundary conditions for $\mathbf{u}$ and let A and B be matrices stemming from a finite element method applied to Eq. (1). For locally supported trial

* Tel.: +7 (095) 939-3834; E-mail: maxim.olshanskii@mtu-net.ru

Computational Fluid and Solid Mechanics 2003
K.J. Bathe (Editor)

functions $\psi_i, \psi_j \in U_h$ and $\phi_k \in Q_h$:

$$\begin{aligned} A_{i,j} &= \nu(\nabla\psi_j, \nabla\psi_i) + \gamma(\operatorname{div}\psi_j, \operatorname{div}\psi_i) + (\alpha\psi_j \\ &\quad + \mathbf{w}\times\psi_j, \psi_i) \\ B_{k,i} &= -(\operatorname{div}\psi_i, \phi_k). \end{aligned} \tag{2}$$

The role of the consistent term $\gamma(\operatorname{div}\psi_j, \operatorname{div}\psi_i)$ will be seen later. With these notations one needs to solve the system of linear algebraic equations:

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{3}$$

This linear system is of saddle point type. A spectrum of the (non-symmetric) matrix from (3) contains eigenvalues with both positive and negative real parts and condition number of the matrix strongly depends on a mesh-size and ν. It is poor conditioned for $h \to 0$ and/or $\nu \to 0$. Thus to solve Eq. (3) iteratively a preconditioning is highly desirable. Let $\hat{A}$ be a preconditioner for A and $\hat{S}$ a preconditioner for S, the Schur complement of the system: $S := BA^{-1}B^T$.

Using patterns from Klawonn and Starke [2] and Silvester et al. [6], we consider the block triangular preconditioner:

$$\mathcal{P}^{-1} = \begin{pmatrix} \hat{A}^{-1} & \hat{A}^{-1}B^T\hat{S}^{-1} \\ 0 & -\hat{S}^{-1} \end{pmatrix} \tag{4}$$

If $\hat{A} = A$ and $\hat{S} = S$, then a preconditioned Krylov subspace method for (3) will converge in at most two iterations. Generally, we are looking for $\hat{A}$ and $\hat{S}$ to be close to A and S, but such that $\hat{A}^{-1}x$ and $\hat{S}^{-1}y$ can be "easily" computed for given vectors x and y. Another approach for building a preconditioner $\mathcal{P}^{-1}$ can be the method of sparse approximate inverses (SPAI, see, e.g., Grote and Huckle [1]). For the linearized equations in convection form this method may be not so efficient in the case of small ν, since the strong upstream dependencies in solution may yield too many nonzero entries in the approximate inverse. The situation changes for rotation form. Now the linearized term has zero order, imposing only "local" dependence. Thus the SPAI method can be efficient. We plan to consider it elsewhere.

Further we are specific about particular choice of $\hat{A}$ and $\hat{S}$. The choice of the BiCGstab method may be more attractive for fluid dynamic applications compared to methods with residual minimization property, like GMRES, since the storage requirements and complexity of the BiCGstab are generally less and independent of an iteration number.

Two essential ingredients: $\hat{A}$ and $\hat{S}$ are remaining to be defined. $\hat{A}$ is a preconditioner for the "**u**"-part of problem. The continuous counterpart of it reads:

$$\alpha\mathbf{v} - \nu\Delta\mathbf{v} - \gamma\nabla\operatorname{div}\mathbf{v} + \mathbf{w}\times\mathbf{v} = r.h.s \quad \text{in } \Omega. \tag{5}$$

Compared to the convection-diffusion vector problem that appears, if one linearizes equations in the original form (1), Eq. (5) has 2nd-order and 0-order terms only. This allows *standard* multigrid components to work well and to solve LAS with the matrix A. Multi-grid components include block Jacobi or block Gauss–Seidel iterations as smoothers. The robustness of the multigrid as a solver was proved in Olshanskii and Reusken [5] for the case $\gamma = 0$. At the same time, if $\gamma = O(1)$ and $\nu, \alpha \to 0$, the problem (5) has anisotropic diffusion terms and may have a large kernel in the limit case; therefore the performance of standard multigrid tools deteriorates. There is a trade between keeping the relation γ/ν reasonably bounded and advantages of taking $\gamma > 0$ for the stability of FE method and the Schur complement preconditioner. We summarize all these, saying that for $\hat{A}^{-1}$ we take few V-cycles of the multigrid method.

Matrix S is a dense matrix: Most entries are nonzero and can not be computed in a straightforward (economical) manner. Therefore, nonstandard considerations have to be made to build $\hat{S}$. Constructing $\hat{S}$ we benefit from the rotation form of equations and the presence of the γ-term in (2). Let M_p be a mass matrix for the space Q_h and $\hat{M}_p$ its diagonal lumping, then

$$\hat{S}^{-1} = (\nu + \gamma)\hat{M}_p^{-1} + S_0(w)^{-1}, \tag{6}$$

where $S_0(w) := B\hat{M}_u(w)^{-1}B^T$ is a Schur compliment of the matrix

$$\begin{pmatrix} \hat{M}_u(w) & B^T \\ B & 0 \end{pmatrix}, \tag{7}$$

where in 2D case $\hat{M}_u(w) = \begin{pmatrix} \alpha\hat{M}_u & -\hat{M}_u^w \\ \hat{M}_u^w & \alpha\hat{M}_u \end{pmatrix}$,

here $\hat{M}_u$ is a lumped mass matrix for U_h and $\hat{M}_u^w$ is a lumped mass-type matrix, corresponding to the w-weighted scalar product $\int_\Omega w(\mathbf{x})u(\mathbf{x})v(\mathbf{x})\,d\mathbf{x}$. For further clarity we note that (7) can be viewed as the matrix of a discrete counterpart of the following reduced problem (1):

$$\begin{aligned} \alpha\mathbf{v} + \mathbf{w}\times\mathbf{v} + \nabla q &= r.h.s. \quad \text{in } \Omega, \\ -\operatorname{div}\mathbf{v} &= r.h.s. \quad \text{in } \Omega. \end{aligned} \tag{8}$$

In the 3D case $\hat{M}_u(w)$ is a 3×3 block matrix. The Schur complement $S_0(w)$ of the reduced problem (in 2D or 3D) is now a *sparse* matrix that mimics a mixed discretization for the pressure-Poisson problem. Therefore a multigrid method is a good candidate to evaluate $S_0(w)^{-1}$. If $\alpha > 0$, then $\hat{S}$ is a robust preconditioner to S w.r.t. a viscosity parameter ν (see Olshanskii [3]). The problem (8) (resp. (7)) can be ill-posed for $\alpha = 0$. Two improvements can be made for small α (see analysis in Olshanskii [4]): one can drop $S_0(w)^{-1}$, than it is important to keep $\gamma = O(1)$ to ensure $cond(\hat{S}^{-1}S)$ is reasonably small. Either one can

consider auxiliary $\bar{\alpha} = O(\|w\|)$ in $S_0(w)^{-1}$ (the only case in our numerical experiments when the latter modification was vital for good convergence is marked by an asterisk in Table 3).

3. Numerical results

We consider problem (1) with analytical solution $\mathbf{u} = (u_1, u_2)$ and p as

$$u_1(x,y) = \frac{r_2}{2\pi}\frac{e^{r_2 y}}{(e^{r_2}-1)}\sin\left(\frac{2\pi(e^{r_2 y}-1)}{e^{r_2}-1}\right) \times\left(1-\cos\left(\frac{2\pi(e^{r_1 x}-1)}{e^{r_1}-1}\right)\right),$$

$$u_2(x,y) = \frac{r_1}{2\pi}\frac{e^{r_1 x}}{(e^{r_1}-1)}\sin\left(\frac{2\pi(e^{r_1 x}-1)}{e^{r_1}-1}\right) \times\left(1-\cos\left(\frac{2\pi(e^{r_2 y}-1)}{e^{r_2}-1}\right)\right),$$

$$p(x,y) = r_1 r_2\frac{e^{r_1 x}e^{r_2 y}}{(e^{r_1}-1)(e^{r_2}-1)}\sin\left(\frac{2\pi(e^{r_1 x}-1)}{e^{r_1}-1}\right) \times\sin\left(\frac{2\pi(e^{r_2 y}-1)}{e^{r_2}-1}\right),$$

with $r_1 = 3$, $r_2 = 0.1$ and $\Omega = (0,1)\times(0,1)$. This type of flow simulates a rotating vortex. The vortex center has coordinates (x_0, y_0), $x_0 \sim 0.785$, $y_0 \sim 0.512$. The Bernoulli pressure can be computed and we reformulate the problem in the rotation form. Calculations were made using LBB-stable P2isoP1-P0 finite element spaces (piecewise-constant pressure and piecewise-linear continuous velocity on 2-times finer triangulation) on regular uniform triangulation with mesh size h. Results for the linearized problem are given in Tables 1–4. Quantities in the tables should be interpreted as follows. N_{outer} is a number of preconditioned BiCGstab iterations needed to solve Eq. (3) up to desired tolerance (10^{-6}); N_{inner} stands for a number of multigrid iterations needed to compute "exact" inverse of A on each step of BiCGstab method, when we take $\hat{A} = A$ (see results in Table 1); ψ_A is a convergence factor of a multigrid method to solve a system with matrix A, we use 4 iterations of the multigrid for the preconditioner $\hat{A}$ in all calculations except those reflected in Table 1; ψ_S is a convergence factor of a multigrid method to solve a system with matrix $S_0(w)$. Table 3 shows also the L_2-norm of the error. In Table 4 we present results for the nonlinear problem in rotation form. Here N_{stp} is a number of nonlinear iterations, N_{iter} is *total* number of BiCGstab iterations executed to solve (with rather low relative accuracy) auxiliary linear problems on every nonlinear step, ψ_{outer} is an averaged convergence factor in these linear iterations.

Table 1
Convergence data with an exact solver for A and $\alpha = 0$, $\gamma = 0.2$

	Viscosity			
$h = 1/32$	2e−2	5e−3	1e−3	1e−4
N_{outer}	5	5	6	6
N_{inner}	13	27	65	230
	Mesh size			
$\nu = 1\text{e}{-3}$	1/16	1/32	1/64	1/128
N_{outer}	5	6	6	6
N_{inner}	55	65	85	93

Table 2
Convergence data with an inexact solver for A and $\alpha = 1$, $\gamma = 0$

	Viscosity			
$h = 1/32$	2e−2	5e−3	1e−3	1e−4
N_{outer}	13	12	14	10
ψ_A	0.05	0.05	0.05	0.02
ψ_S	0.39	0.31	0.30	0.21
	Mesh size				
$\nu = 1\text{e}{-3}$	1/16	1/32	1/64	1/128	1/256
N_{outer}	10	14	16	18	18
ψ_A	0.05	0.05	0.04	0.04	0.05
ψ_S	0.23	0.30	0.41	0.51	0.70

Table 3
The effect of γ on an error and iterations

Viscosity		Value of γ				
		0	0.05	0.2	0.5	1.0
1e−1	$\|u-u_h\|$	1.9e−2	1.3e−2	8.6e−3	9.0e−3	1.3e−2
	$\|p-p_h\|$	4.3e−2	4.4e−2	4.9e−2	6.7e−2	1.3e−1
	N_{outer}	11	9	6	6	4
	ψ_A	0.06	0.07	0.11	0.16	0.21
1e−3	$\|u-u_h\|$	1.8e−0	1.1e−1	8.0e−2	1.2e−1	1.4e−1
	$\|p-p_h\|$	3.3e−1	5.0e−2	4.6e−2	5.3e−2	6.0e−2
	N_{outer}	29	11	10	19	30
	ψ_A	0.08	0.37	0.50	0.59	0.61

Several conclusions follow from numerical results. The rotation form of the Navier–Stokes equations gives rise to linearized equations which can be solved in a robust way by the BiCGstab method. The corresponding linear algebraic system admits efficient preconditioner (see robustness results in Tables 1 and 2). To achieve robustness for the case $\alpha = 0$, the additional stabilizing term ($\gamma > 0$) is, however, needed. This additional term also improves the accuracy of solution (see Table 3) for small ν. At the same time, if $\gamma = O(1)$ and $\nu \to 0$, then standard multigrid solvers for the discretized velocity problem (5) are not so efficient

Table 4
Convergence data for nonlinear problem

Viscosity	Quantity	Mesh size			
		1/16	1/32	1/64	1/128
$\nu = 2e{-}2$	N_{stp}	7	7	7	7
	N_{iter}	45	50	50	52
	ψ_{outer}	0.13	0.15	0.16	0.16
$\nu = 5e{-}3$	N_{stp}	73	78	81	83
	N_{iter}	290	383	467	490
	ψ_{outer}	0.09	0.13	0.15	0.15

(N_{inner} increases as $\nu \to 0$ in Table 1). We still need more efficient tools for this case to build good preconditioner $\hat{A}$. One noted disadvantage of solving the Navier–Stokes equations in rotation form is that nonlinear iterations of fixed point type appears to be not so robust w.r.t. ν (see Table 4), while the inner linear solvers perform excellently.

Acknowledgements

The work was partially supported by the Russian Foundation for Basic Research grant 02-01-00592.

References

[1] Grote MG, Huckle T. Parallel preconditioning with sparse approximate inverses. SIAM J Sci Comput 1997;18:838–853.
[2] Klawonn A, Starke G. Block triangular preconditioners for nonsymmetric saddle point problem. Numer Math 1999;81:577–594.
[3] Olshanskii MA. An iterative solver for the Oseen problem and the numerical solution of incompressible Navier–Stokes equations. Numer Linear Alg Appl 1999;6:353–378.
[4] Olshanskii MA. A low order Galerkin finite element method for the Navier–Stokes equations of steady incompressible flow: A stabilization issue and iterative methods. Comput Meth Appl Mech Eng 2002;191:5515–5536.
[5] Olshanskii MA, Reusken A. Navier–Stokes equations in rotation form: A robust multigrid solver for the velocity problem. SIAM J Sci Comput 2002;23:1683–1706.
[6] Silvester D, Elman HC, Kay D, Wathen A. Efficient preconditioning of the linearized Navier–Stokes equations. J Comput Appl Math 2001;128:261–279.