

Robust Multigrid Method for Solving the Navier-Stokes Equations on Structured Grids

Sergey I. Martynenko¹ & Potroh S. Martynenko¹

¹ Central Institute of Aviation Motors n.a. P.I. Baranov, Moscow, Russia

Correspondence: Sergey I. Martynenko, Central Institute of Aviation Motors n.a. P.I. Baranov, 2, Aviamotornaya St., Moscow 111116, Russia. Tel: 8-495-362-0024. E-mail: Martynenko@ciam.ru

Received: April 6, 2012

Accepted: May 15, 2012

Online Published: May 28, 2012

doi:10.5539/mas.v6n6p73

URL: <http://dx.doi.org/10.5539/mas.v6n6p73>

Abstract

The paper represents an efficient multigrid algorithm without the problem-dependent components for solving the Navier–Stokes equations in primitive variables formulation on structured grids. The algorithm consists of Vanka smoother, pressure decomposition and robust multigrid technique. Detailed description of the proposed approach and results of numerical experiment are given.

Keywords: CFD, Navier-Stokes equations, Vanka smoother, pressure decomposition, multigrid methods, structured grids

1. Introduction

Let $\Omega \in \mathbb{R}^N$ be a bounded, connected domain with a piecewise smooth boundary $\partial\Omega$. Navier-Stokes equations governing flow of a Newtonian, incompressible viscous fluid can be written in the following operator form

$$\begin{cases} \mathcal{N}(\vec{V}) + \nabla P = F \\ \nabla \vec{V} = G \end{cases}, \quad (1)$$

where $\mathcal{N}$ is a nonlinear convection-diffusion operator, ∇P is a pressure gradient, F and G are source terms. Given a boundary data, the problem is to find the velocity field $\vec{V}$ and pressure P . In follows we assume that the boundary conditions are included in the operators and source terms.

Discretization of (1) using the finite differences or the finite elements and Picard or Newton linearization result in a generalized saddle point system

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}, \quad (2)$$

in which α and β represent the discrete velocities and discrete pressure, respectively. Here nonsymmetric A is a block diagonal matrix, where each block corresponds to a discrete convection-diffusion operator $\mathcal{N}$ with appropriate boundary conditions. The rectangular matrix B^T represents the discrete gradient operator while B represents its adjoint, the divergence operator.

The saddle point problem (2) can be solved by the preconditioned Uzawa algorithm defined as

$$A\alpha^{(m+1)} = f - B^T \beta^{(m)}, \quad (3a)$$

$$Q\beta^{(m+1)} = Q\beta^{(m)} + B\alpha^{(m+1)} - g, \quad (3b)$$

where matrix Q is a preconditioner. Substitution of $\alpha^{(m+1)} = A^{-1}(f - B^T \beta^{(m)})$ from (3a) into (3b) gives

$$\beta^{(m+1)} = (I - Q^{-1}BA^{-1}B^T)\beta^{(m)} + Q^{-1}(BA^{-1}f - g).$$

It is result to the following estimation

$$\|\beta - \beta^{(m+1)}\| \leq \|I - Q^{-1}BA^{-1}B^T\| \cdot \|\beta - \beta^{(m)}\|,$$

where β is an exact solution. Choice of the preconditioner Q so

$$\|I - Q^{-1}BA^{-1}B^T\| \leq q < 1$$

leads to the Uzawa iteration convergence

$$\|\beta - \beta^{(m)}\| \leq q^m \|\beta - \beta^{(0)}\|.$$

Preconditioning has been and remains a most active area of research, some the most widely used and promising methods are described in Benzi (2005). Total efforts needed for solving the Navier-Stokes equations by Uzawa method are sum of efforts for the algorithm adaptation to the given problem (choice of the preconditioner Q , problem-dependent components for the matrices A and Q inversion, etc.) and efforts for solution of the saddle point problem (2). Optimal adaptation of the algorithm to the given problem is difficult question, but it guarantees high computational efficiency of the Uzawa solver.

On the other hand, the algorithms with the least number of the problem-dependent components seem to be more preferable for practical applications because of reduction of efforts for their adaptation. However in this case it is difficult to obtain high computational efficiency.

The goal of this paper is to present and test an iterative robust solver for the Navier-Stokes equations on structured grids. The method is based on the problem-independent components: Vanka smoother, pressure decomposition and robust multigrid technique. Special attention is paid to robustness and efficiency of the offered algorithm.

2. Vanka Smoother

Two-dimensional steady Navier-Stokes equations (1) can be written as:

a) continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (4a)$$

b) X-momentum

$$\frac{\partial(u^2)}{\partial x} + \frac{\partial(vu)}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (4b)$$

c) Y-momentum

$$\frac{\partial(uv)}{\partial x} + \frac{\partial(v^2)}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right), \quad (4c)$$

where Re is a Reynolds number.

Discrete linearized continuity and momentum equations can be written as:

a) discrete continuity equation

$$\frac{u_{i+1,j} - u_{ij}}{h_x} + \frac{v_{ij+1} - v_{ij}}{h_y} = 0, \quad (5a)$$

b) discrete linearized X-momentum

$$\begin{aligned} & \bar{A}_{ij}^w u_{i-1,j} + \bar{A}_{ij}^e u_{i+1,j} + \bar{A}_{ij}^s u_{ij-1} + \bar{A}_{ij}^n u_{ij+1} + \bar{A}_{ij} u_{ij} + \\ & + \bar{A}_{ij}^{NE} v_{ij+1} + \bar{A}_{ij}^{NW} v_{i-1,j+1} + \bar{A}_{ij}^{SE} v_{ij} + \bar{A}_{ij}^{SW} v_{i-1,j} - \frac{p_{ij} - p_{i-1,j}}{h_x} = s_{ij}^u, \end{aligned} \quad (5b)$$

c) discrete linearized Y-momentum

$$\begin{aligned} & \tilde{A}_{ij}^w v_{i-1,j} + \tilde{A}_{ij}^e v_{i+1,j} + \tilde{A}_{ij}^s v_{ij-1} + \tilde{A}_{ij}^n v_{ij+1} + \tilde{A}_{ij} v_{ij} + \\ & + \tilde{A}_{ij}^{NE} u_{i+1,j} + \tilde{A}_{ij}^{NW} u_{ij} + \tilde{A}_{ij}^{SE} u_{i+1,j-1} + \tilde{A}_{ij}^{SW} u_{ij-1} - \frac{p_{ij} - p_{ij-1}}{h_y} = s_{ij}^v. \end{aligned} \quad (5c)$$

Coefficients in the momentum equations (excepting discrete pressure gradients) are functions of the velocity components. Stencils for the approximation of the continuity (4a) and momentum (4b)-(4c) equations on a staggered grid are shown on Figure 1.

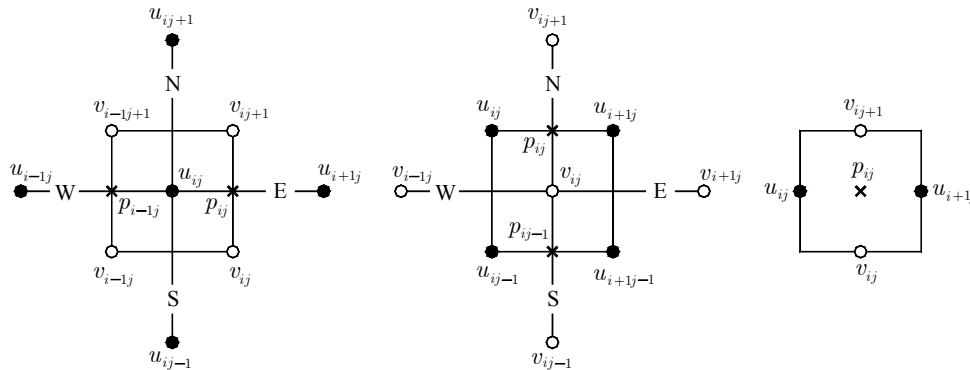


Figure 1. Stencils and the control volumes for discretization of the Navier-Stokes equations on the staggered grids: X-momentum (left), Y-momentum (centre), continuity equation (right)

Vanka-like procedures solve for four velocity components $u_{ij}, u_{i+1,j}, v_{ij}$ and v_{ij+1} (at the four cell faces) and the pressure p_{ij} (at the center of the cell) simultaneously for each cell (Figure 1, right) Vanka (1986). The discrete X and Y-momentum equations (5b)-(5c) for the four cell faces together with the continuity equation (5a) can be written as

$$\begin{pmatrix} \bar{A}_{ij} & \bar{A}_{ij}^e & \bar{A}_{ij}^{SE} & \bar{A}_{ij}^{NE} & -(h_x)^{-1} \\ \bar{A}_{i+1,j}^w & \bar{A}_{i+1,j}^e & \bar{A}_{i+1,j}^{SW} & \bar{A}_{i+1,j}^{NW} & (h_x)^{-1} \\ \tilde{A}_{ij}^{NW} & \tilde{A}_{ij}^{NE} & \tilde{A}_{ij}^n & \tilde{A}_{ij}^s & -(h_y)^{-1} \\ \tilde{A}_{ij+1}^{SW} & \tilde{A}_{ij+1}^{SE} & \tilde{A}_{ij+1}^s & \tilde{A}_{ij+1}^n & (h_y)^{-1} \\ -(h_x)^{-1} & (h_x)^{-1} & -(h_y)^{-1} & (h_y)^{-1} & 0 \end{pmatrix} \begin{pmatrix} u_{ij} \\ u_{i+1,j} \\ v_{ij} \\ v_{ij+1} \\ p_{ij} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ 0 \end{pmatrix}, \quad (6)$$

where the components of the right-side vector are given by

$$\begin{aligned} b_1 &= s_{ij}^u - \bar{A}_{ij}^w u_{i-1,j} - \bar{A}_{ij}^e u_{i+1,j} - \bar{A}_{ij}^n u_{ij-1} - \bar{A}_{ij}^s u_{ij+1} - \bar{A}_{ij}^{NW} v_{i-1,j+1} - \bar{A}_{ij}^{SW} v_{i-1,j} - \frac{p_{i-1,j}}{h_x}, \\ b_2 &= s_{i+1,j}^u - \bar{A}_{i+1,j}^w u_{i,j} - \bar{A}_{i+1,j}^e u_{i+2,j} - \bar{A}_{i+1,j}^n u_{i+1,j-1} - \bar{A}_{i+1,j}^s u_{i+1,j+1} - \bar{A}_{i+1,j}^{NE} v_{i+1,j+1} - \bar{A}_{i+1,j}^{SE} v_{i+1,j} + \frac{p_{i+1,j}}{h_x}, \\ b_3 &= s_{ij}^v - \tilde{A}_{ij}^w v_{i-1,j} - \tilde{A}_{ij}^e v_{i+1,j} - \tilde{A}_{ij}^n v_{ij-1} - \tilde{A}_{ij}^s v_{ij+1} - \tilde{A}_{ij}^{SE} u_{i+1,j-1} - \tilde{A}_{ij}^{SW} u_{ij-1} - \frac{p_{ij-1}}{h_y}, \\ b_4 &= s_{ij+1}^v - \tilde{A}_{ij+1}^w v_{i,j+1} - \tilde{A}_{ij+1}^e v_{i+1,j+1} - \tilde{A}_{ij+1}^n v_{ij+2} - \tilde{A}_{ij+1}^s u_{i+1,j+1} - \tilde{A}_{ij+1}^{NW} u_{ij+1} + \frac{p_{ij+1}}{h_y}. \end{aligned}$$

The system (6) can be solved by Gaussian elimination. Accounting nonlinear nature of the momentum equations, the velocity components and pressure can be updated using underrelaxation as

$$\begin{pmatrix} u_{ij} \\ u_{i+1,j} \\ v_{ij} \\ v_{ij+1} \\ p_{ij} \end{pmatrix}^{\text{new}} = \begin{pmatrix} u_{ij} \\ u_{i+1,j} \\ v_{ij} \\ v_{ij+1} \\ p_{ij} \end{pmatrix}^{\text{old}} + \begin{pmatrix} \omega_u & & & & \\ & \omega_u & & & \\ & & \omega_v & & \\ & & & \omega_v & \\ & & & & \omega_p \end{pmatrix} \left[\begin{pmatrix} u_{ij} \\ u_{i+1,j} \\ v_{ij} \\ v_{ij+1} \\ p_{ij} \end{pmatrix}^{\text{new}} - \begin{pmatrix} u_{ij} \\ u_{i+1,j} \\ v_{ij} \\ v_{ij+1} \\ p_{ij} \end{pmatrix}^{\text{old}} \right].$$

In practice the underrelaxation parameters are taken as $\omega_u = \omega_v$ and $\omega_p = 1$ (Thompson & Ferziger, 1989). If a starting guess is not close to the solution, it is recommended values $0.2 < \omega_u = \omega_v < 0.8$ in the first iterations.

Vanka method has no the problem-dependent components (preconditioners, artificial boundary conditions for pressure, relaxation parameters (if a starting guess is close to the solution) etc.), but slow convergence rate is observed and the rate depends strongly on the unknown ordering. As a result, the method requires different acceleration techniques. Since Vanka-like methods have very good smoothing properties, often the approach is

used in multigrid algorithms as smoother.

3. Pressure Decomposition

In partial cases the fluid flows can be described by the simplified Navier-Stokes equations. Basic assumption is that the pressure is not changed across the flow. Numerical solution of the simplified Navier-Stokes equations can be reduced to solution of series of the saddle point problem (2) with zero block of the least size in the coefficient matrix. This fact allows develop very efficient algorithms for solving the simplified Navier-Stokes equations. Unfortunately these algorithms cannot be applied for solving the full Navier-Stokes equations because the pressure is changed in all spatial directions. However algorithms developed for the simplified Navier-Stokes equations can be used for convergence acceleration of the iterative methods for the full Navier-Stokes equations.

For the given purpose, it is necessary to artificially extract «one-dimensional parts of pressure» from the pressure field by adding and subtracting items $p^x(t, x)$, $p^y(t, y)$ and $p^z(t, z)$ which depends only on one spatial variable, i.e.

$$p(t, x, y, z) = p^x(t, x) + p^y(t, y) + p^z(t, z) + (-p^x(t, x) - p^y(t, y) - p^z(t, z) + p(t, x, y, z)),$$

where superscripts x , y , z and xyz indicate dependence of the components on the spatial coordinates. Let us introduce a new variable

$$p^{xyz}(t, x, y, z) = -p^x(t, x) - p^y(t, y) - p^z(t, z) + p(t, x, y, z).$$

Finally the pressure can be represented as

$$p(t, x, y, z) = p^x(t, x) + p^y(t, y) + p^z(t, z) + p^{xyz}(t, x, y, z). \quad (7)$$

Basic idea of the method consists in application of the efficient numerical methods developed for the simplified Navier-Stokes equations for computation of «part of pressure» (i.e. for $p^x(t, x) + p^y(t, y) + p^z(t, z)$). Fast computation of the «part of pressure» results in reduction of total computational efforts needed for the full Navier-Stokes equations.

In spite of simplicity of the representation (7), it is necessary to comment the principle of formal decomposition of pressure:

Remark 1. All items $p^x(t, x)$, $p^y(t, y)$, $p^z(t, z)$ and $p^{xyz}(t, x, y, z)$ in (7) have no physical meaning, but physical meaning has their sum. In follows, the items $p^x(t, x)$, $p^y(t, y)$ and $p^z(t, z)$ will be called as «one-dimensional components of the pressure», and $p^{xyz}(t, x, y, z)$ as «multidimensional component». The quotes «» will indicate absence of the physical meaning of the «pressure components».

Remark 2. In N -dimensional case ($N=2,3$) pressure is represented as sum of $N+1$ «components», therefore the method requires N extra conditions for computation of «the one-dimensional components». The convergence acceleration technique uses N mass conservation equations as a priori information of physical nature.

Remark 3. In spite of representation of the pressure as sum of $N+1$ «components», all momentum equations have only two «pressure gradients». For example, for X-momentum we obtain

$$\frac{\partial p}{\partial x} = \frac{\partial}{\partial x} (p^x(t, x) + p^y(t, y) + p^z(t, z) + p^{xyz}(t, x, y, z)) = \frac{\partial p^x}{\partial x} + \frac{\partial p^{xyz}}{\partial x}.$$

Remark 4. Efficiency of the acceleration technique depends strongly on the flow nature. For directed fluid flows (for example, flows in nozzles, pipes etc.) gradient of one of «one-dimensional components of pressure» $p^x(t, x)$, $p^y(t, y)$ or $p^z(t, z)$ is dominant. In this case impressive reduction of computational work is expected as compared with traditional algorithms ($p^x = p^y = p^z = 0$), i.e.

$$p^x(t, x) + p^y(t, y) + p^z(t, z) \rightarrow p(t, x, y, z) \Rightarrow p^{xyz}(t, x, y, z) \rightarrow 0 \Rightarrow \text{efforts} \downarrow.$$

However for strongly rotated flows (for example, flow in a driven cavity) the approach shows the least efficiency.

Remark 5. In 3D case the method will be more efficient than in 2D case.

Remark 6. Velocity components and corresponding «one-dimensional components of pressure» in (7) are computed only in coupled manner. Velocity components and «multidimensional component» $p^{xyz}(t, x, y, z)$ in (7) can be computed in decoupled or coupled manner.

Remark 7. Gradients of the «one-dimensional components» can be obtained in analytical form for the explicit schemes. Implicit schemes require an auxiliary problem for determination of the «one-dimensional components» in (7).

The pressure decomposition can be used for fast computation of a starting guess close to the solution of the full Navier-Stokes equations or built-in in some basic solver (SIMPLE, Uzawa or Vanka method, etc.).

Similarly to Vanka method, the convergence acceleration technique does not require the problem-dependent components. Detailed description of the approach and its applications for solving benchmark and applied problems are given in Martynenko (2009; 2011).

4. Model Problem

For clearness of the multigrid algorithm representation, we consider problem about 2D steady fluid flow in unit cavity. Geometry of the problem is shown on Figure 2. Boundary conditions for the velocity components in (4) are given by

$$u(0, y) = \begin{cases} 100y(0.2 - y), & y \leq 0.2 \\ 0, & y > 0.2 \end{cases}, \quad u(1, y) = u(x, 0) = u(x, 1) = 0,$$

$$v(x, 1) = \begin{cases} 100(x - 0.8)(1 - x), & x \geq 0.8 \\ 0, & x < 0.8 \end{cases}, \quad v(0, y) = v(1, y) = v(x, 0) = 0.$$

Convergence acceleration based on the pressure decomposition (7) requires two mass conservation equations. Integration of the continuity equation (4a) over the control volumes V_1 and V_2 shown on Figure 2 gives

$$\int_0^1 u(x, y) dy = \int_0^1 u(0, y) dy - \int_0^x v(\xi, 1) d\xi, \quad (8a)$$

$$\int_0^1 v(x, y) dx = \int_0^y u(0, \xi) d\xi. \quad (8b)$$

The mass conservation equations (8a) and (8b) will be used for fast computation of the «one-dimensional components of pressure» in (7).

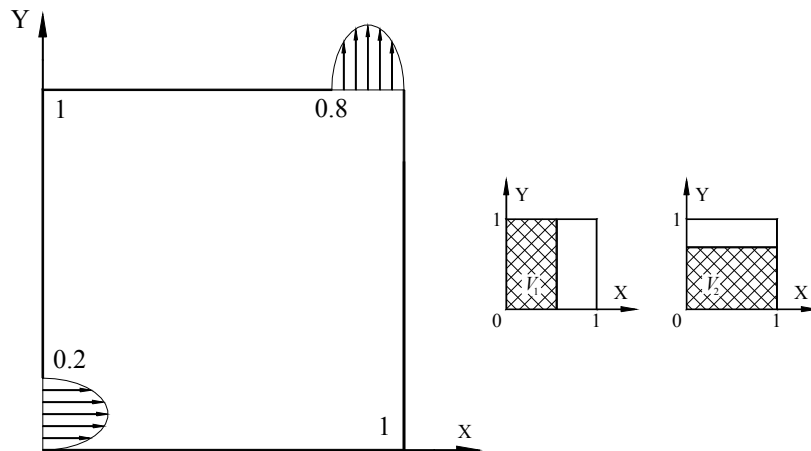


Figure 2. Geometry of the model problems and the control volumes V_1 and V_2

5. Robust Multigrid Technique

Robust multigrid technique (RMT) has been developed as a variant of the geometric multigrid methods with the problem-independent transfer operators Martynenko (2006). To overcome problem of robustness, RMT consists of two parts: analytical (adaptation of the boundary value problems to RMT) and computational (solution of the adapted problems by original multigrid algorithm). RMT is intended for solving a large class of nonlinear applied problems on the structured grids with almost optimal convergence rate. Really the smoothing procedure is single problem-dependent component of RMT.

5.1 Analytical Part: Adaptation of the Navier-Stokes Equations to RMT

Adaptation of the Navier-Stokes equations (1) to RMT consists of representation of the solution as sum of two functions

$$\vec{V} = C_{\vec{V}} + \hat{\vec{V}} \quad \text{and} \quad P = C_p + \hat{P}, \quad (9)$$

where the functions $C_{\vec{V}}$ and C_p will be coarse grid corrections to velocity and pressure, and the functions $\hat{\vec{V}}$ and $\hat{P}$ will be approximations to the solution in the following multigrid iterations.

Representation (9) is called Σ -modification of the solution. Substitution of (9) into (1) yields the following Σ -modified form of the Navier-Stokes equations (1)

$$\begin{cases} \mathcal{N}(C_{\vec{V}} + \hat{\vec{V}}) + \nabla(C_p + \hat{P}) = F \\ \nabla(C_{\vec{V}} + \hat{\vec{V}}) = G \end{cases}.$$

Since the nonlinear convection-diffusion operator can be represented as

$$\mathcal{N}(C_{\vec{V}} + \hat{\vec{V}}) = \mathcal{N}^*(C_{\vec{V}}) + \mathcal{N}(\hat{\vec{V}}),$$

we obtain the Σ -modified form of the Navier-Stokes equations

$$\begin{cases} \mathcal{N}^*(C_{\vec{V}}) + \nabla(C_p) = F^* \\ \nabla(C_{\vec{V}}) = G^* \end{cases},$$

where the source terms take the form

$$F^* = F - \mathcal{N}(\hat{\vec{V}}) - \nabla(\hat{P}) \quad \text{and} \quad G^* = G - \nabla(\hat{\vec{V}}).$$

Convergence of RMT means reduction of the corrections $C_{\vec{V}}$ and C_p , therefore the approximations to the solution $\hat{\vec{V}}$ and $\hat{P}$ will satisfy to the Navier-Stokes equations (1) i.e.

$$\begin{cases} C_{\vec{V}} \rightarrow 0 \\ C_p \rightarrow 0 \end{cases} \Rightarrow \begin{cases} F^* \rightarrow 0 \\ G^* \rightarrow 0 \end{cases} \Rightarrow \begin{cases} \mathcal{N}(\hat{\vec{V}}) + \nabla(\hat{P}) \rightarrow F \\ \nabla(\hat{\vec{V}}) \rightarrow G \end{cases}.$$

Two-dimensional steady Navier-Stokes equations (4) can be rewritten in the Σ -modified form as:

a) Σ -modified continuity equation

$$\frac{\partial c_u}{\partial x} + \frac{\partial c_v}{\partial y} + R^{uv}(x, y) = 0, \quad (10a)$$

b) Σ -modified X-momentum

$$\frac{\partial(c_u^2)}{\partial x} + 2 \frac{\partial(\hat{u}c_u)}{\partial x} + \frac{\partial(\hat{u}c_v)}{\partial y} + \frac{\partial(\hat{v}c_u)}{\partial y} + \frac{\partial(c_u c_v)}{\partial y} = -\frac{\partial c_p}{\partial x} + \frac{1}{\text{Re}} \left(\frac{\partial^2 c_u}{\partial x^2} + \frac{\partial^2 c_u}{\partial y^2} \right) + R^u(x, y), \quad (10b)$$

c) Σ -modified Y-momentum

$$\frac{\partial(\hat{u}c_v)}{\partial x} + \frac{\partial(\hat{v}c_u)}{\partial x} + \frac{\partial(c_u c_v)}{\partial x} + \frac{\partial(c_v^2)}{\partial y} + 2 \frac{\partial(\hat{v}c_v)}{\partial y} = -\frac{\partial c_p}{\partial y} + \frac{1}{\text{Re}} \left(\frac{\partial^2 c_v}{\partial x^2} + \frac{\partial^2 c_v}{\partial y^2} \right) + R^v(x, y), \quad (10c)$$

where discrete analogues of the functions c_u , c_v and c_p will be coarse grid corrections for velocity components u , v and pressure p and discrete analogues of the functions $\hat{u}$, $\hat{v}$ and $\hat{p}$ will be approximations to the solutions in the following multigrid iterations.

The source terms in the Σ -modified Navier-Stokes equations coincide with non-modified ones:

$$R^{uv}(x, y) = \frac{\partial \hat{u}}{\partial x} + \frac{\partial \hat{v}}{\partial y}, \quad (11a)$$

$$R^u(x, y) = -\frac{\partial(\hat{u}^2)}{\partial x} - \frac{\partial(\hat{u}\hat{v})}{\partial y} - \frac{\partial\hat{p}}{\partial x} + \frac{1}{\text{Re}} \left(\frac{\partial^2 \hat{u}}{\partial x^2} + \frac{\partial^2 \hat{u}}{\partial y^2} \right), \quad (11b)$$

$$R^v(x, y) = -\frac{\partial(\hat{u}\hat{v})}{\partial x} - \frac{\partial(\hat{v}^2)}{\partial y} - \frac{\partial\hat{p}}{\partial y} + \frac{1}{\text{Re}} \left(\frac{\partial^2 \hat{v}}{\partial x^2} + \frac{\partial^2 \hat{v}}{\partial y^2} \right). \quad (11c)$$

Additional convection terms in Σ -modified momentum equations (10b) and (10c) are result of nonlinear nature of the convection-diffusion operator $\mathcal{N}$ (i.e. $\mathcal{N} \neq \mathcal{N}^*$). Approximation of the source terms (11) in Eq. (10) defines the accuracy, monotonicity and conservatism of the numerical solutions. Approximation of other terms in (10) affects only on the multigrid convergence rate.

Boundary conditions for the velocity components are Σ -modified in the same manner. Since the Dirichlet boundary conditions are given for the model problem, we obtain zero Dirichlet boundary conditions for the velocity corrections c_u and c_v .

Also the mass conservation equations (8) should be rewritten in the Σ -modified form:

$$\int_0^1 c_u(x, y) dy = -\int_0^1 \hat{u}(x, y) dy + \int_0^1 u(0, y) dy - \int_0^x \nu(\xi, 1) d\xi, \quad (12a)$$

$$\int_0^1 c_v(x, y) dx = -\int_0^1 \hat{v}(x, y) dx + \int_0^y u(0, \xi) d\xi. \quad (12b)$$

For clearness the pressure decomposition (7) will be used for fast computation of a starting guess. First of all it is necessary to formulate an auxiliary problem, we hope that the solution of the auxiliary problem will be close to the solution of the Navier-Stokes equations (4) as compared with zero starting guess. Pressure decomposition for the 2D steady Navier-Stokes equations in «corrections-residuals» variables formulation (10)-(11) is written as

$$c_p(x, y) = c_p^x(x) + c_p^y(y). \quad (13)$$

Formulation of the auxiliary problem is based on replacement of the Σ -modified continuity equation (10a) by the Σ -modified mass conservation equations (12). Accounting (13), we obtain:

a) Σ -modified X-momentum (10b) and Σ -modified mass conservation equations (12a)

$$\left\{ \begin{array}{l} \frac{\partial(c_u^2)}{\partial x} + 2\frac{\partial(\hat{u}c_u)}{\partial x} + \frac{\partial(\hat{u}c_v)}{\partial y} + \frac{\partial(\hat{v}c_u)}{\partial y} + \frac{\partial(c_u c_v)}{\partial y} = -\frac{dc_p^x}{dx} + \frac{1}{\text{Re}} \left(\frac{\partial^2 c_u}{\partial x^2} + \frac{\partial^2 c_u}{\partial y^2} \right) + R^u(x, y) \\ \int_0^1 c_u(x, y) dy = -\int_0^1 \hat{u}(x, y) dy + \int_0^1 u(0, y) dy - \int_0^x \nu(\xi, 1) d\xi \end{array} \right., \quad (14a)$$

b) Σ -modified Y-momentum (10c) and Σ -modified mass conservation equations (12b)

$$\left\{ \begin{array}{l} \frac{\partial(\hat{u}c_v)}{\partial x} + \frac{\partial(\hat{v}c_u)}{\partial x} + \frac{\partial(c_u c_v)}{\partial x} + \frac{\partial(c_v^2)}{\partial y} + 2\frac{\partial(\hat{v}c_v)}{\partial y} = -\frac{dc_p^y}{dy} + \frac{1}{\text{Re}} \left(\frac{\partial^2 c_v}{\partial x^2} + \frac{\partial^2 c_v}{\partial y^2} \right) + R^v(x, y) \\ \int_0^1 c_v(x, y) dx = -\int_0^1 \hat{v}(x, y) dx + \int_0^y u(0, \xi) d\xi \end{array} \right., \quad (14b)$$

where the braces mean that the momentum and mass conservation equations are solved only in coupled manner.

Let us summarize the main features of the auxiliary problem (14):

1) due to (13), the momentum equations in the auxiliary problem (14) are not pressure-linked. Systems (14a) and (14b) can be solved by line (2D) or plane (3D) Seidel method coupled with secant iterations. Previously similar algorithm had been proposed by Briley (1967) for solving the simplified Navier-Stokes equations.

2) solution of the auxiliary problem satisfies to the mass conservation equations (12), but not satisfies to the continuity equation (10a). It gives us a reason to hope that solution of (14) will be an accurate starting guess for the Navier-Stokes equations (4).

To illustrate abovementioned deduction, Figures 3 and 4 represent comparison of the solutions of the auxiliary problem (14) and the Navier-Stokes equations for the model problem ($Re=100$, grid 101×101). Good agreement is observed in the stream function behavior, but agreement in pressure is not accurate because of only «part of pressure» ($p^x + p^y$) is computed in the auxiliary problem (14).

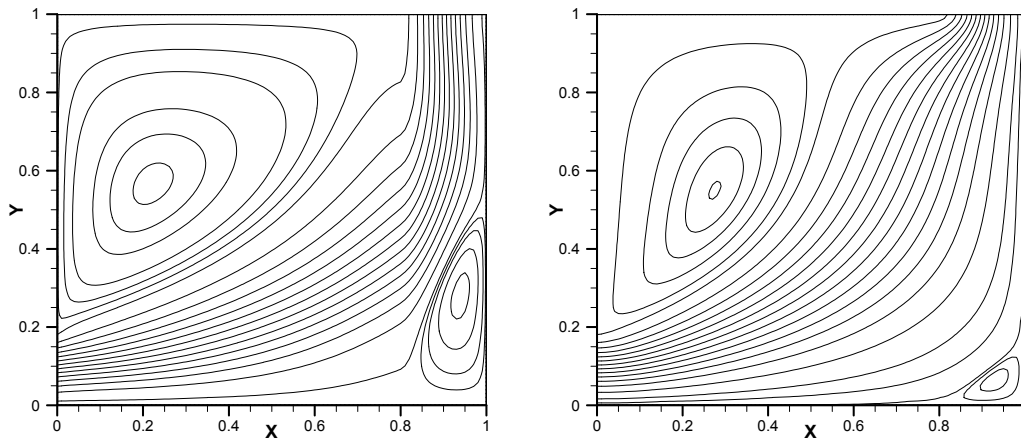


Figure 3. Isolines of the stream function: solution of the auxiliary problem (14) (left) vs. solution of the Navier-Stokes equations (4) (right)

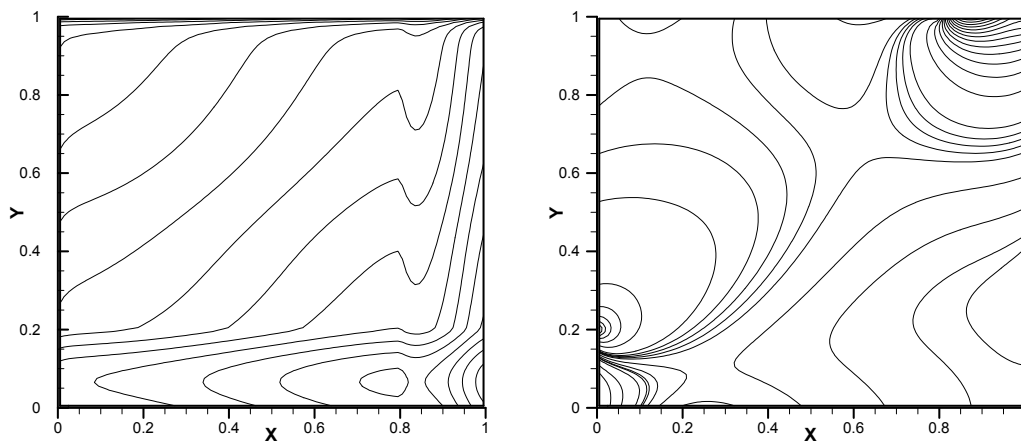


Figure 4. Isobars: solution of the auxiliary problem (14) ($p^x + p^y$, left) vs. solution of the Navier-Stokes equations (4) (right)

5.2 Computational Part

First stage of the computational part of RMT consists of the finest grid generation in domain for the following control volume approximation of the modified equations. The finest grid G_1^0 consists of two sets $G^v(0;1)$ and $G^f(0;1)$ of the grid points defined as

$$G^v(0;1) = \{x_i^v \mid x_i^v = (i-1)h, \quad i = 1, 2, \dots, H_x^0 + 1, \quad h = 1/H_x^0\},$$

$$G^f(0;1) = \{x_i^f \mid x_i^f = 0.5(x_i^v + x_{i+1}^v), \quad i = 1, 2, \dots, H_x^0\}.$$

The finest grid G_1^0 with $H_x^0 = 8$ is shown on Figure 5.

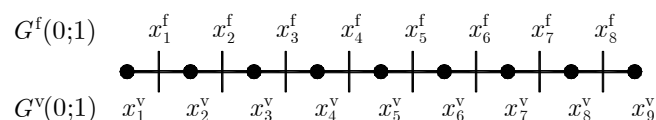


Figure 5. Staggered grid in X direction

Coarsening in RMT is based on representation of the finest grid G_1^0 as union of three coarse grids G_1^1 , G_2^1 and G_3^1 as shown on Figure 6. It is easy to see the following properties of the coarse grids:

1) the coarse grids G_1^1 , G_2^1 and G_3^1 have no common points, i.e.

$$G_n^1 \cap G_m^1 = \emptyset, \quad n \neq m.$$

2) the finest grid G_1^0 is the union of the coarse grids G_1^1 , G_2^1 and G_3^1 , i.e.

$$G_1^0 = \bigcup_{k=1}^3 G_k^1.$$

3) all grids are similar to each other, but a mesh size on the coarse grids is three times as large as than the mesh size on the finest grid.

4) the discrete functions can be assigned to the grid points x^v or to the grid points x^f , but in both cases the control volume on the coarse grids is union of three control volumes on the finest grid (Figure 7).

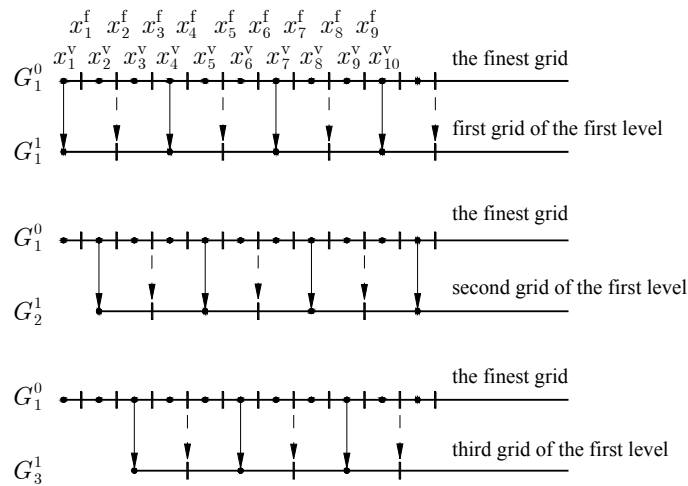


Figure 6. Coarsening in RMT

Properties 1, 2, and 4 give efficient parallelism, problem-independent prolongation operator and problem-independent restriction operator, respectively.

The finest grid forms the zero level and three coarse grids form the first level. The coarse grid generation is further recurrently repeated: each grid $G_i^l, i = 1, 2, \dots, 3^l$ of a current level l is considered to be the finest grid for the coarse grids $G_j^{l+1}, j = 1, 2, \dots, 3^{l+1}$ of the next level $l+1$. Nine coarse grids derived from the three grids of the first level form the second level, etc. The coarse grid generation is finished when no further coarsening can be performed. The grid hierarchy will be called a multigrid structure (Figure 8).

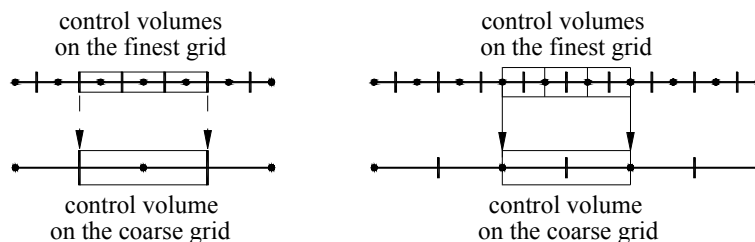


Figure 7. Control volumes on the finest and coarse grids

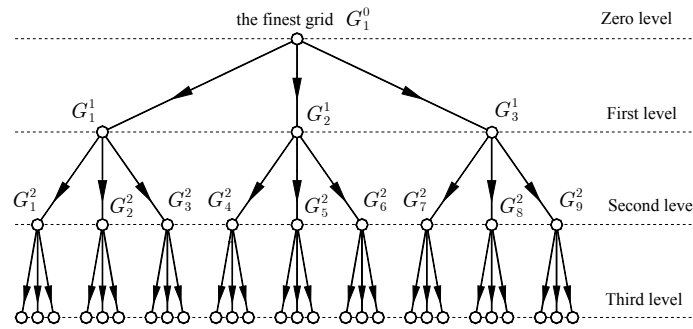


Figure 8. Multigrid structure

Note that the classical multigrid methods use the reverse numbering of levels.

Since N -dimensional grid ($N = 2, 3$) can be represented as product of N one-dimensional grids, similar triple coarsening is performed independently in each spatial direction. Therefore l^{th} level consists of 3^{Nl} grids in multidimensional case.

The number of levels can be computed in advance. Assume that majority of the coarsest grids has three grid points. Then the number of the finest grid points is $H_x^0 + 1$ or $\approx 3^{L^+ + 1}$, where L^+ is number of the coarsest level. Therefore

$$H_x^0 + 1 \approx 3^{L^+ + 1} \Rightarrow L^+ = \left\lfloor \frac{\lg(H_x^0 + 1)}{\lg 3} - 1 \right\rfloor,$$

where square brackets mean integer part.

In 3D case the finest grid $(H_x^0 + 1) \times (H_y^0 + 1) \times (H_z^0 + 1)$ with the mesh sizes $h_x = 1/H_x^0$, $h_y = 1/H_y^0$ and $h_z = 1/H_z^0$ has different coarsest levels

$$L_x^+ = \left\lfloor \frac{\lg(H_x^0 + 1)}{\lg 3} - 1 \right\rfloor, \quad L_y^+ = \left\lfloor \frac{\lg(H_y^0 + 1)}{\lg 3} - 1 \right\rfloor, \quad L_z^+ = \left\lfloor \frac{\lg(H_z^0 + 1)}{\lg 3} - 1 \right\rfloor.$$

Each level $l = \{l_x, l_y, l_z\}$ consists of the computational grids $G_{k_x k_y k_z}^{l_x l_y l_z}$, $k_x = 1, 2, \dots, 3^{l_x}$, $k_y = 1, 2, \dots, 3^{l_y}$,

$k_z = 1, 2, \dots, 3^{l_z}$ with mesh size $h_x 3^{l_x}, h_y 3^{l_y}, h_z 3^{l_z}$, where h_x, h_y, h_z are the mesh sizes of the finest grid G_{111}^{000} and

$l_x = \min\{l; L_x^+\}$, $l_y = \min\{l; L_y^+\}$, $l_z = \min\{l; L_z^+\}$. Control volume on the grid $G_{k_x k_y k_z}^{l_x l_y l_z}$ is an union of

$3^{l_x + l_y + l_z}$ control volumes on the finest grid.

An example of the coarse grid generation (two level structure) for the finest grid with $H_x^0 = 30$ is shown on Figure 9. Let us define the mapping of indices to simplify operations with the multigrid structure. The notion «a grid of the l^{th} level» means the one-to-one mapping of indices of the coarse grid points onto the indices of the finest grid points. In the rest of the paper, the braces $\{\}$ will denote the mapping. The mapping of indices of the grid points from the sets G^v and G^f is written as $x_{\{i\}}^v$ and $x_{\{i\}}^f$ respectively, where i and $\{i\}$ are the coarse and finest grid indices. For example, for grid G_1^1 shown on Figure 9 we obtain

$$x_{\{1\}}^v = x_3^v, x_{\{2\}}^v = x_6^v, x_{\{3\}}^v = x_9^v, x_{\{4\}}^v = x_{12}^v, \dots,$$

$$x_{\{1\}}^f = x_4^f, x_{\{2\}}^f = x_7^f, x_{\{3\}}^f = x_{10}^f, x_{\{4\}}^f = x_{13}^f, \dots$$

The mapping of indices gives a close-to-the-finest-grid notation. For example, the second order derivative of the discrete function assigned to the grid points x^v on the multigrid structure is approximated as

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{x_{ij}^v} = \frac{u_{\{i-1j\}} - 2u_{\{ij\}} + u_{\{i+1j\}}}{h_x^2 3^{2l_x}} + O(h_x^2 3^{2l_x}).$$

Figure 9 shows that the approximation in the grid points $x_{\{2\}}^v \in G_1^1$ (marked by circle) is expressed as

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{x_{\{2\}}^v} \approx \frac{u_{\{1\}} - 2u_{\{2\}} + u_{\{3\}}}{h_x^2 3^{2l_x}} = \frac{u_3 - 2u_6 + u_9}{9h_x^2}.$$

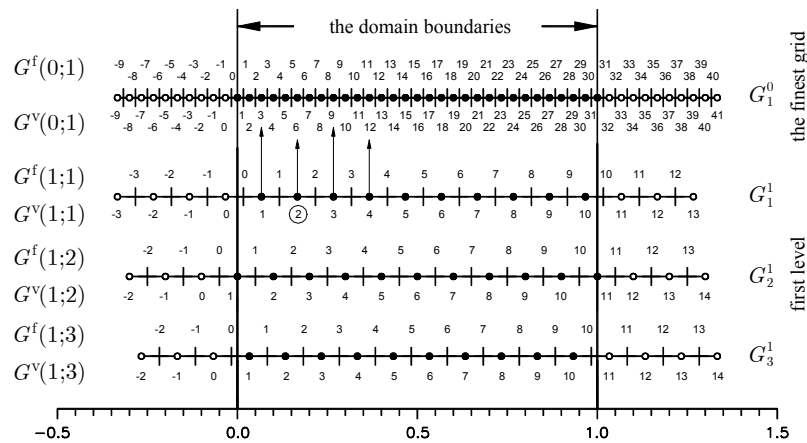


Figure 9. Coarse grids of the first level and the index mapping

5.2.1 Multigrid Iterations

Multigrid cycle of RMT for solving linear problems is shown on Figure 10. Computations start on the coarsest level. When the coarsest level solution has been obtained, the transfer to the next finer level is performed. It should be emphasized that the transfer does not add any interpolation errors to the correction c as shown on Figure 11. It means that RMT has problem-independent prolongation operator. Smooth parts of the error are deleted on all grids of the next finer levels in the same manner (computation of the coefficient matrix and right-hand side vector and the smoothing iterations). The coarse grid correction to be added to $\hat{u}$ on the finest grid is c ($\hat{u} = \hat{u} + c$). The multigrid iterations repeatedly improve the approximation to the solution $\hat{u}$ until the current approximation becomes accurate enough. In particular applications the finest grid should be reconstructed after each multigrid iteration for their adaptation to the solution singularities.

In RMT more computational work must be spent on coarse grids in order to allow for the best approximation to the solution on the finest grid. Smoothing before coarse grid correction (pre-smoothing) is deleted in RMT to simplify solution of nonlinear problems (like a sawtooth cycle in classical multigrid). It is clear that RMT takes the intermediate place between classical and cascadic multigrid algorithms.

For nonlinear problems when no a priori information of the solution is available to assist in the choice of the initial guess on the finest grid, it is obviously wasteful to start the computation of the finest grid as shown on Figure 12. It is similarly to nested iterations in classical multigrid methods. Main difference consists of formulation of discrete problems of the coarse grids.

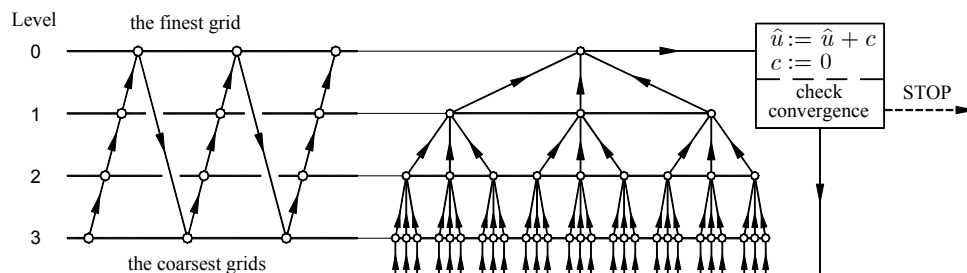


Figure 10. Multigrid cycle for the linear problems

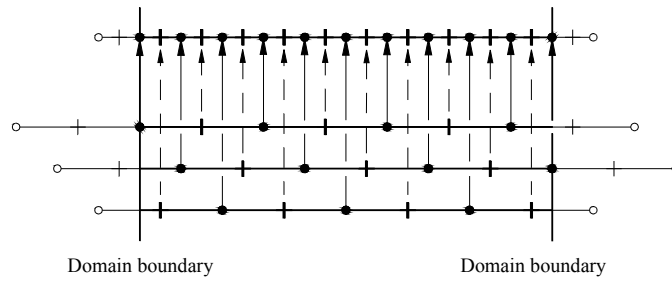


Figure 11. Prolongation operator of RMT

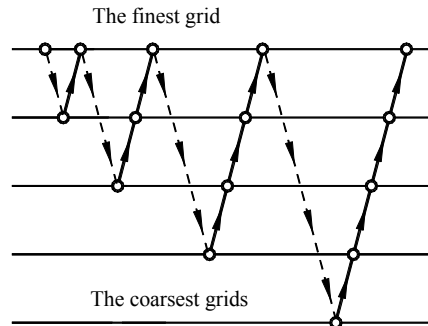


Figure 12. Multigrid cycle for the nonlinear problems

5.2.2 Control Volume Approximation of the Σ -modified Equations on the Multigrid Structure

Integration of the Σ -modified X-momentum (10b) over the control volume

$$V_{\{ij\}} = \{(x,y) \mid x_{\{i-1\}}^f \leq x \leq x_{\{i\}}^f; y_{\{j-1\}}^v \leq y \leq y_{\{j+1\}}^v\}$$

on some coarse grid gives

$$\begin{aligned} \frac{J_e^u - J_w^u}{h_x 3^{l_x}} + \frac{J_n^u - J_s^u}{h_y 3^{l_y}} = & -\frac{(c_p)_{\{ij\}} - (c_p)_{\{i-1,j\}}}{h_x 3^{l_x}} + \\ & + \frac{1}{\text{Re}} \frac{(c_u)_{\{i-1,j\}} - 2(c_u)_{\{ij\}} + (c_u)_{\{i+1,j\}}}{h_x^2 3^{2l_x}} + \frac{1}{\text{Re}} \frac{(c_u)_{\{ij-1\}} - 2(c_u)_{\{ij\}} + (c_u)_{\{ij+1\}}}{h_y^2 3^{2l_y}} + \langle R^u \rangle_{\{ij\}} \end{aligned}$$

where convective fluxes are given by

$$J_w^u = (c_u)_w \langle \hat{u} \rangle_{\{ij\}}^w, \quad J_e^u = (c_u)_e \langle \hat{u} \rangle_{\{ij\}}^e,$$

$$J_s^u = (c_u)_s \langle \hat{u} \rangle_{\{ij\}}^s + (c_u)_s \langle \hat{v} \rangle_{\{ij\}}^s + (c_u c_v)_s, \quad J_n^u = (c_u)_n \langle \hat{u} \rangle_{\{ij+1\}}^n + (c_u)_n \langle \hat{v} \rangle_{\{ij+1\}}^n + (c_u c_v)_n,$$

and the source term

$$\langle R^u \rangle_{\{ij\}} = \frac{1}{h_x 3^{l_x}} \frac{1}{h_y 3^{l_y}} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \int_{y_{\{j-1\}}^v}^{y_{\{j+1\}}^v} \left(-\frac{\partial(\hat{u}^2)}{\partial x} - \frac{\partial(\hat{u}\hat{v})}{\partial y} - \frac{\partial \hat{p}}{\partial x} + \frac{1}{\text{Re}} \left(\frac{\partial^2 \hat{u}}{\partial x^2} + \frac{\partial^2 \hat{u}}{\partial y^2} \right) \right) dy dx \quad (15)$$

is an averaged residual of the X-momentum equation.

Indices w,e,n and s denote values on western, eastern, northern and southern faces of the control volume. The convection terms with the exception of the coefficients $\langle \hat{u} \rangle_{\{ij\}}^w$, $\langle \hat{u} \rangle_{\{ij\}}^e$, $\langle \hat{u} \rangle_{\{ij\}}^s$, $\langle \hat{u} \rangle_{\{ij\}}^n$, $\langle \hat{v} \rangle_{\{ij\}}^s$ and $\langle \hat{v} \rangle_{\{ij\}}^n$ can be approximated by upwind differences.

The coefficients $\langle \hat{u} \rangle_{\{ij\}}^w$, $\langle \hat{u} \rangle_{\{ij\}}^e$, $\langle \hat{u} \rangle_{\{ij\}}^s$, $\langle \hat{u} \rangle_{\{ij\}}^n$, $\langle \hat{v} \rangle_{\{ij\}}^s$ and $\langle \hat{v} \rangle_{\{ij\}}^n$

$$\begin{aligned}\langle \hat{u} \rangle_{\{ij\}}^w &= \frac{1}{h_y 3^{l_y}} \int_{y_{\{j\}}^v}^{y_{\{j+1\}}^v} \hat{u}(x_{\{i-1\}}^f, y) dy, \quad \langle \hat{u} \rangle_{\{ij\}}^e = \frac{1}{h_y 3^{l_y}} \int_{y_{\{j\}}^v}^{y_{\{j+1\}}^v} \hat{u}(x_{\{i\}}^f, y) dy, \\ \langle \hat{u} \rangle_{\{ij\}}^s &= \frac{1}{h_x 3^{l_x}} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \hat{u}(x, y_{\{j\}}^v) dx, \quad \langle \hat{u} \rangle_{\{ij\}}^n = \frac{1}{h_x 3^{l_x}} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \hat{u}(x, y_{\{j+1\}}^v) dx, \\ \langle \hat{v} \rangle_{\{ij\}}^s &= \frac{1}{h_x 3^{l_x}} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \hat{v}(x, y_{\{j\}}^v) dx, \quad \langle \hat{v} \rangle_{\{ij\}}^n = \frac{1}{h_x 3^{l_x}} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \hat{v}(x, y_{\{j+1\}}^v) dx,\end{aligned}$$

are average values of approximations to the velocity components on the control volume faces. To overcome problem of robustness, all integrals should be evaluated on the finest grid. As an example, we consider evaluation of the integral

$$\langle q \rangle_{\{i\}} = \frac{1}{h 3^l} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} q(x) dx$$

in domain $\Omega = [0, 1]$. The function $q(x)$ can be redefined as

$$\tilde{q}(x) = \begin{cases} 0, & x \notin \Omega \\ q(x), & x \in \Omega \end{cases}.$$

Further we define a characteristic function $\Upsilon(x)$ as

$$\Upsilon(x) = \begin{cases} 0, & x \notin \Omega \\ 1, & x \in \Omega \end{cases}.$$

Now the coefficient $\langle q \rangle_{\{i\}}$ can be rewritten as

$$\langle q \rangle_{\{i\}} = \frac{\langle \tilde{q} \rangle_{\{i\}}^l}{\Lambda_{\{i\}}^l},$$

where $\langle \tilde{q} \rangle_{\{i\}}^l$ and $\Lambda_{\{i\}}^l$ are given by

$$\langle \tilde{q} \rangle_{\{i\}}^l = \frac{1}{h} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \tilde{q}(x) dx \quad \text{and} \quad \Lambda_{\{i\}}^l = \frac{1}{h} \int_{x_{\{i-1\}}^f}^{x_{\{i\}}^f} \Upsilon(x) dx.$$

$\Lambda_{\{i\}}^l$ is the number of control volumes on the finest grid forming the real part of the control volume on the given grid. Computation of $\langle \tilde{q} \rangle_{\{i\}}^l$ starts from the finest grid ($l = 0, \{i\} = i$) where the integral

$$\langle \tilde{q} \rangle_i^{l=0} = \frac{1}{h} \int_{x_{i-1}^f}^{x_i^f} \tilde{q}(x) dx$$

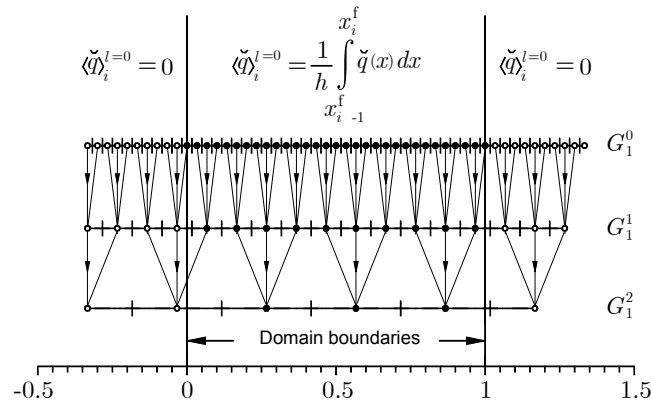
is computed by some method (trapezoids, rectangles etc.), and

$$\Lambda_i^{l=0} = \frac{1}{h} \int_{x_{i-1}^f}^{x_i^f} \Upsilon(x) dx = \begin{cases} 0, & x_i^v \notin \Omega \\ 1, & x_i^v \in \Omega \end{cases}.$$

After that $\langle \tilde{q} \rangle_{\{i\}}^l$ and $\Lambda_{\{i\}}^l$ are computed by the recursive equation

$$\begin{cases} \langle \tilde{q} \rangle_{\{i\}}^l = \langle \tilde{q} \rangle_{\{i\}-3^{l-1}}^{l-1} + \langle \tilde{q} \rangle_{\{i\}}^{l-1} + \langle \tilde{q} \rangle_{\{i\}+3^{l-1}}^{l-1}, \\ \Lambda_{\{i\}}^l = \Lambda_{\{i\}-3^{l-1}}^{l-1} + \Lambda_{\{i\}}^{l-1} + \Lambda_{\{i\}+3^{l-1}}^{l-1} \end{cases}, \quad l = 1, 2, \dots, L. \quad (16)$$

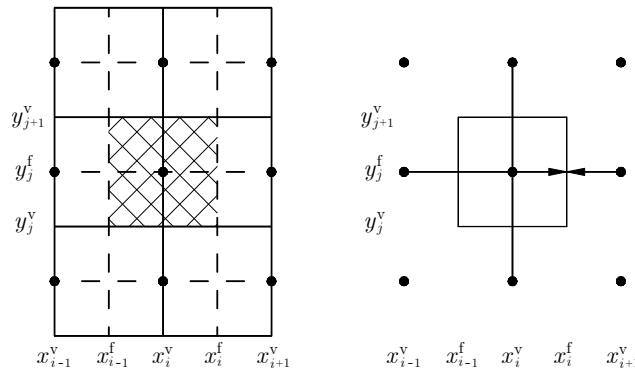
Scheme of the integral evaluation on coarse grids G_1^1 and G_1^2 is shown on Figure 13. Virtual nodes and faces on each grid are intended only for the computational procedure.

Figure 13. Scheme of the integral evaluation in domain $\Omega = [0, 1]$

As an example, we consider computation of the average value

$$\langle \hat{u} \rangle_{ij}^e = \frac{1}{h_y 3^{l_y}} \int_{y_{ij}^v}^{y_{ij+1}^v} \hat{u}(x_{ij}^f, y) dy$$

on the control volume faces, where $\hat{u}(x, y) = e^{x+y}$. Figure 14 represents the finest uniform staggered grid and location of the control volume, the function $\hat{u}$ is assigned to the grid points $\bullet$.

Figure 14. Finest staggered grid (left) and computation of the average value $\langle \hat{u} \rangle_{ij}^e$ (right)

The average value on the control volume face on the finest grid can be computed as

$$\langle \hat{u} \rangle_{ij}^e = \frac{1}{h_y} \int_{y_{ij}^v}^{y_{ij+1}^v} \hat{u}(x_{ij}^f, y) dy \approx \frac{1}{2} (\hat{u}_{ij} + \hat{u}_{i+1,j}), \quad (17)$$

as shown on Figure 14. Exact average value $\langle \hat{u} \rangle_{ij}^{\text{exact}}$ on the control volume faces

$V_{ij} = \{(x, y) \mid x_{i-1}^f \leq x \leq x_{i+1}^f, y_{ij}^v \leq y \leq y_{ij+1}^v\}$ is given by

$$\langle \hat{u} \rangle_{ij}^{\text{exact}} = \frac{1}{h_y 3^{l_y}} \int_{y_{ij}^v}^{y_{ij+1}^v} \exp(x_{ij}^f + y) dy = \exp(x_{ij}^f) \frac{\exp(y_{ij+1}^v) - \exp(y_{ij}^v)}{y_{ij+1}^v - y_{ij}^v},$$

where $y_{ij+1}^v - y_{ij}^v = h_y 3^{l_y}$. Since the coarse grid points can be located outside the domain $\Omega = [0, 1]$ (i.e.

$y_{ij}^v < 0$ or $y_{ij+1}^v > 1$), the integral should be redefined as

$$\langle \hat{u} \rangle_{ij}^{\text{exact}} = \exp(x_{ij}^f) \frac{\exp(b) - \exp(a)}{b - a}, \quad (18)$$

where $a = \max(0, y_{ij}^v)$ and $b = \min(1, y_{ij+1}^v)$.

Six-level ($L^+ = 5$) multigrid structure with 1001×1001 finest grid ($h_x = h_y = 1/1000$) is used for the numerical experiment. Computation of the average value $\langle \hat{u} \rangle_{ij}^e$ on the finest and coarse grids is performed as Eqs. (16) and (17), respectively. Error of the computation is given by

$$\text{ErrMAX} = \max_{ij} | \langle \hat{u} \rangle_{ij}^{\text{exact}} - \langle \hat{u} \rangle_{ij}^e |,$$

where $\langle \hat{u} \rangle_{ij}^{\text{exact}}$ is the exact value (18). Results of the numerical experiment are

LevelX = 5	LevelY = 5	ErrMAX = .261E-06
LevelX = 4	LevelY = 4	ErrMAX = .207E-06
LevelX = 3	LevelY = 3	ErrMAX = .391E-06
LevelX = 2	LevelY = 2	ErrMAX = .473E-06
LevelX = 1	LevelY = 1	ErrMAX = .574E-06
LevelX = 0	LevelY = 0	ErrMAX = .388E-06

where LevelX (l_x) and LevelY (l_y) are numbers of grid levels in X and Y directions, respectively. It easy to see that the error of the integral computation weakly depends on the mesh size on the coarse grid ($h_x 3^{l_x}, h_y 3^{l_y}$). Note that the computational procedure is based on the additivity property of the integrals and, as a result, the procedure is problem-independent component of RMT.

Source term (17) is computed in the same manner. At first, the source term is computed on the finest grid as

$$\langle R^u \rangle_{ij} = -\frac{(\hat{u}^2)_e - (\hat{u}^2)_w}{h_x} - \frac{(\hat{u}\hat{v})_n - (\hat{u}\hat{v})_s}{h_y} - \frac{\hat{p}_{ij} - \hat{p}_{i-1,j}}{h_x} + \frac{1}{\text{Re}} \frac{\hat{u}_{i-1,j} - 2\hat{u}_{ij} + \hat{u}_{i+1,j}}{h_x^2} + \frac{1}{\text{Re}} \frac{\hat{u}_{ij-1} - 2\hat{u}_{ij} + \hat{u}_{ij+1}}{h_y^2}.$$

Values of the approximations to the velocity components $\hat{u}$ and $\hat{v}$ on the control volume faces can be approximated by high order schemes. Such different representation of the convection terms in both sides of the modified momentum equations is similar to the defect correction procedure Hackbusch (1981). Finally the source term on the coarse grids is evaluated as a double integral. The abovementioned averaging is a restriction operator of RMT.

6. Numerical Experiments

In the application of the algorithm to the cavity problem shown on Figure 2, a number of flow Reynolds numbers and computational grid nodes have been considered. Computations have been made for Reynolds numbers of 100 and 500 with computational grids consisting of 101×101 and 1001×1001 nodes. The residuals are defined as

$$\begin{aligned} \|r_{uv}\| &= \max_{ij} |R^{uv}(x_i^f, y_j^f)|, \\ \|r_u\| &= \max_{ij} |R^u(x_i^v, y_j^f)|, \\ \|r_v\| &= \max_{ij} |R^v(x_i^f, y_j^v)|, \end{aligned}$$

where R^{uv} , R^u and R^v are given by Eq. (11). In addition, the residuals of the mass conservation equations (8) are defines as

$$\begin{aligned} \|r_u^*\| &= \max_i \left| -h_y \sum_{j=1}^{N_y} \hat{u}_{ij} + h_y \sum_{j=1}^{N_y} \hat{u}_{1j} - h_x \sum_{k=1}^i \hat{v}_{kN_y+1} \right|, \\ \|r_v^*\| &= \max_j \left| -h_x \sum_{i=1}^{N_x} \hat{v}_{ij} + h_y \sum_{k=1}^j \hat{u}_{1k} \right|. \end{aligned}$$

The auxiliary problem (14) has been used for fast formulation of the starting guess for the solution of the full Navier-Stokes equations. Figure 15 represents convergence history of RMT obtained with four smoothing

iterations on each grid.

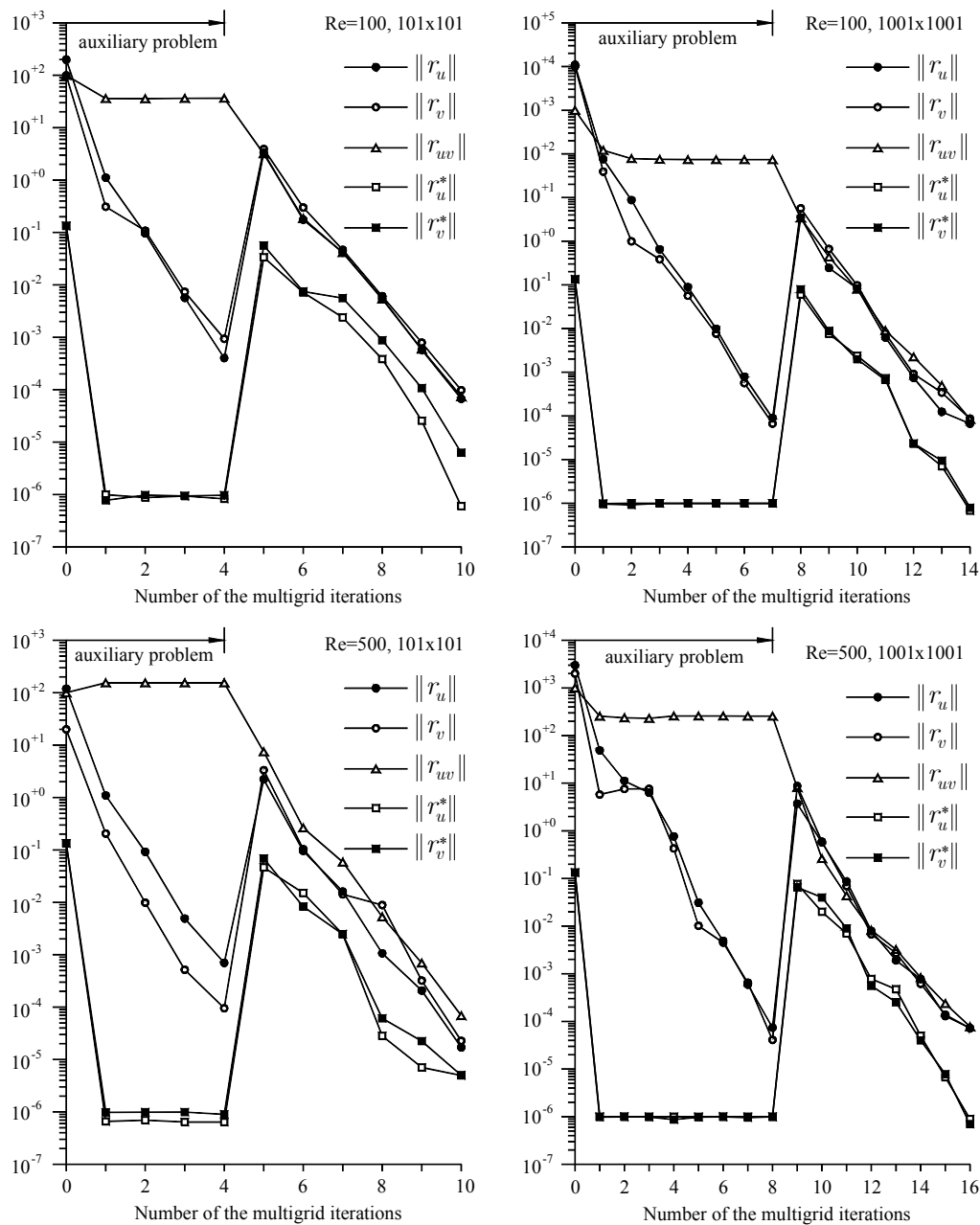


Figure 15. Convergence history of RMT

7. Convergence Analysis of RMT for Linear Problems

Convergence analysis of RMT is performed for linear boundary value problem

$$L(u) = f,$$

where the linear operator L and right-hand side function f include the boundary conditions. The problem can be rewritten in the Σ -modified form

$$L(c) = f - L(\hat{u}).$$

Discrete analogue of the Σ -modified boundary value problem on some grid G_k^l ($k = 1, 2, \dots, 3^{Nl}$, $l = 0, 1, \dots, L^+$, $N = 2, 3$) is written as

$$a_{G_k^l} c_{G_k^l} = R_{G_1^0 \rightarrow G_k^l} (b - Au^{(n)})_0,$$

where $R_{G_1^0 \rightarrow G_k^l}$ is a restriction operator, i.e. the operator transfers the residual $b - Au^{(n)}$ from the finest grid G_1^0 on the current grid G_k^l . Previously it was shown that the operator is problem-independent component of RMT.

Since all grid of the same level have no common nodes and control volume faces, discrete equations on all grids of the same level can be written as

$$\begin{pmatrix} a_{G_1^l} & & & \\ & a_{G_2^l} & & \\ & & \dots & \\ & & & a_{G_{3^{Nl}}^l} \end{pmatrix} \begin{pmatrix} c_{G_1^l} \\ c_{G_2^l} \\ \dots \\ c_{G_{3^{Nl}}^l} \end{pmatrix} = \begin{pmatrix} R_{G_1^0 \rightarrow G_1^l} (b - Au^{(n)})_0 \\ R_{G_1^0 \rightarrow G_2^l} (b - Au^{(n)})_0 \\ \dots \\ R_{G_1^0 \rightarrow G_{3^{Nl}}^l} (b - Au^{(n)})_0 \end{pmatrix},$$

or

$$A_l c_l = R_{0 \rightarrow l} (b - Au^{(n)})_0, \quad (19)$$

where A_l is a block diagonal matrix, the number of blocks is equal to the number of grids forming the level (i.e. 3^{Nl} , $l = 0, 1, 2, \dots, L^+$).

Transfer to the level consisting of the finer grids is abbreviated as

$$c_l = P_{l+1 \rightarrow l} c_{l+1}.$$

Accordingly to Figure 11, the permutation matrix $P_{l+1 \rightarrow l}$ corresponds to the problem-independent prolongation operator.

Smoothing iteration for (19) can be written as

$$W_l (c_l^{(v_l+1)} - c_l^{(v_l)}) = R_{0 \rightarrow l} (b - Au^{(n)})_0 - A_l c_l^{(v_l)}$$

or

$$c_l^{(v_l+1)} = (I - W_l^{-1} A_l) c_l^{(v_l)} + W_l^{-1} R_{0 \rightarrow l} (b - Au^{(n)})_0, \quad (20)$$

where matrix W_l defines the smoothing procedure and v_l is number of the smoothing iterations.

Convergence analysis of RMT is based on the following assumptions:

Assumption 1. Smoothing procedure is convergent iterative method, i.e.

$$\|S_l\| = \|I - W_l^{-1} A_l\| \leq q < 1,$$

where S_l is a matrix of the smoothing iterations.

Assumption 2. Eq. (20) is solved exactly on the coarsest grids ($l = L^+$):

$$c_{L^+} = A_{L^+}^{-1} R_{0 \rightarrow L^+} (b - Au^{(n)})_0.$$

Eq. (20) can be written as

$$c_l - c_l^{(v_l)} = S_l^{v_l} (c_l - c_l^{(0)}), \quad (21)$$

where c_l is an exact solution of Eq. (19)

$$c_l = A_l^{-1} R_{0 \rightarrow l} (b - Au^{(n)})_0.$$

Since the starting guess $c_l^{(0)}$ is obtained by “prolongation” of approximation to the solution from coarse grids

$$c_l^{(0)} = P_{l+1 \rightarrow l} c_{l+1}^{(v_l+1)},$$

we obtain

$$c_l - c_l^{(0)} = d_l (b - Au^{(n)})_0 + P_{l+1 \rightarrow l} (c_{l+1} - c_{l+1}^{(v_l+1)}),$$

where

$$d_l = A_l^{-1} R_{0 \rightarrow l} - P_{l+1 \rightarrow l} A_{l+1}^{-1} R_{0 \rightarrow l+1}.$$

It results in the recursive form of Eq. (21)

$$c_l - c_l^{(v_l)} = S_l^{v_l} d_l (b - Au^{(n)})_0 + S_l^{v_l} P_{l+1 \rightarrow l} (c_{l+1} - c_{l+1}^{(v_l+1)}). \quad (22)$$

Assume that the grid visiting order corresponds to order shown on Figure 10. The recursive Eq. (22) should be written for each level of the multigrid structure. Accounting the Assumption 2, Eq (22) takes the form

$$c_{L^+} = c_{L^+}^{(v_{L^+}+1)}$$

for the level consisting from the coarsest grids ($l = L^+$).

For the next level ($L^+ - 1$) consisting of the finer grids, we obtain the following form of Eq. (22)

$$c_{L^+-1} - c_{L^+-1}^{(v_{L^+-1})} = S_{L^+-1}^{v_{L^+-1}} d_{L^+-1} (b - Au^{(n)})_0.$$

Continuing this procedure for each level ($l = L^+ - 2, L^+ - 3, \dots, 0$) of the multigrid structure, we can conclude

$$c_l - c_l^{(v_l)} = S_l^{v_l} d_l^* (b - Au^{(n)})_0,$$

where

$$d_l^* = \begin{cases} d_l + P_{l+1 \rightarrow l} S_{l+1}^{v_{l+1}} d_{l+1}^*, & l = L^+ - 1, L^+ - 2, \dots, 0 \\ A_{L^+-1}^{-1} R_{0 \rightarrow L^+-1} - P_{L^+ \rightarrow L^+-1} A_{L^+}^{-1} R_{0 \rightarrow L^+}, & l = L^+ - 1 \end{cases}.$$

Finally the multigrid iterations of RMT is written as

$$u^{(n+1)} = Mu^{(n)} + (A_0^{-1} - S_0^{v_0} d_0^*) b,$$

where the matrix of the multigrid iterations takes the form

$$M = S_0^{v_0} \left(d_0 + \sum_{l=1}^{L^+-1} \prod_{k=1}^l P_{k \rightarrow k-1} S_k^{v_k} d_l \right) A_0.$$

To estimate the matrix norm, it is used the assumption:

Assumption 3 (approximation property)

$$\|d_l\| = \|A_l^{-1} R_{0 \rightarrow l} - P_{l+1 \rightarrow l} A_{l+1}^{-1} R_{0 \rightarrow l+1}\| \leq C_A \|A_0\|^{-1},$$

where the constant C_A is independent on l .

Accounting the approximation property and Assumption 1, the matrix norm can be estimated as

$$\|M\| \leq C_A q_0^{v_0} \left(1 + \sum_{l=1}^{L^+-1} \prod_{k=1}^l q_k^{v_k} \right) \leq C_A \bar{q} \frac{\bar{q}^{L^+} - 1}{\bar{q} - 1}, \quad \text{where } \bar{q} = \max_{1 \leq l \leq L^+} q_l^{v_l}.$$

The estimation shows that convergence of the multigrid iterations ($\|M\| < 1$) can be obtained by performing sufficient number of the smoothing iterations on the multigrid structure. In addition, smoothing on the finer grids has more influence on the convergence of RMT.

Note the estimation does not account the smoothing properties of used iterative methods. As a result, convergence rate of RMT cannot be studied theoretically using the estimation. Computational cost of the multigrid iterations is proportional to $N \lg N$ arithmetical operations, where N is the number of the finest grid nodes Martynenko (2006). The problem-independent prolongation operator of RMT leads to the multiplier $\lg N$. In sense of the computational work RMT loses to the problem-adapted classical multigrid methods required N arithmetical operations for solving the boundary value problems.

8. Conclusions

Proposed algorithm based on the problem-independent components (Vanka smoother, pressure decomposition and robust multigrid technique) makes it possible to solve many boundary value problems for the Navier-Stokes equations to within truncation error at a cost of $c \cdot N \lg N$ arithmetic operations, where N is the number of unknowns and c is a constant which depends on the problem.

Acknowledgements

Work supported by Russian Foundation for the Basic Research (project no. 12-01-00109). I wish to express a great appreciation to Professor M. P. Galanin (Keldysh Institute of Applied Mathematics of Russian Academy of Sciences), who has guided and supported the researches.

References

- Benzi, M., Golub, G. H., & Liesen, J. (2005). Numerical solution of saddle point problems. *Acta Num*, 1-137.
- Briley, W. R. (1967). Numerical method for predicting three-dimensional steady viscous flow in ducts. *J. Comp.*

- Phys.*, 14, 8-28. [http://dx.doi.org/10.1016/0021-9991\(74\)90002-3](http://dx.doi.org/10.1016/0021-9991(74)90002-3)
- Hackbusch, W. (1981). Bemerkungen zur iterativen Defektkorrektur und zu ihrer Kombination mit Mehrgitterverfahren. *Rev. Roumaine Math. Pures Appl.*, 26, 1319-1329.
- Martynenko, S. (2011). Convergence Acceleration of Iterative Algorithms for Solving Navier–Stokes Equations on Structured Grids. *Hydrodynamics - Optimizing Methods and Tools*, 175-200.
- Martynenko, S. I. (2006). Robust Multigrid Technique for black box software. *Comp. Meth. in Appl. Math.*, 6(4), 413-435.
- Martynenko, S. I. (2009). A physical approach to development of numerical methods for solving Navier-Stokes equations in primitive variables formulation. *Int. J. of Comp. Science and Math.*, 2(4), 291-307. <http://dx.doi.org/10.1504/IJCSM.2009.030910>
- Thompson, M. C., & Ferziger, J. H. (1989). An adaptive multigrid technique for the incompressible Navier–Stokes equations. *J. Comp. Phys.*, 82, 94-121. [http://dx.doi.org/10.1016/0021-9991\(89\)90037-5](http://dx.doi.org/10.1016/0021-9991(89)90037-5)
- Vanka, S. P. (1986). Block-implicit multigrid solution of Navier–Stokes equations in primitive variables. *J. Comp. Phys.*, 65, 138-158. [http://dx.doi.org/10.1016/0021-9991\(86\)90008-2](http://dx.doi.org/10.1016/0021-9991(86)90008-2)