ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

PSEUDOMULTIGRID GAUSS-SEIDEL METHOD FOR LARGE SCALE AND HIGH PERFORMANCE COMPUTING

S.I. Martynenko, V.M. Volokhov and P.D. Toktaliev

The Institute of Problems of Chemical Physics of the Russian Academy of Sciences
Academician Semenov avenue 1, Chernogolovka, Moscow region, 142432 Russian Federation
e-mail: Martynenko@icp.ac.ru; vvm@icp.ac.ru; Toktaliev@icp.ac.ru

Keywords: multigrid, robustness, structured grids, parallel multigrid

Abstract. *The paper represents a new approach for development of robust multigrid methods. The approach is based on minimization of the number of problem-dependent components in multigrid algorithm. Robust Multigrid Technique (RMT) is application of the essential multigrid principle in single-grid algorithm. Problem-dependent components of the technique are:*
1) the number of smoothing iterations;
2) unknown ordering (for anisotropic problems);
3) underrelaxation parameter (for nonlinear problems);
4) stopping criterion of the multigrid iterations.
The least number of the problem-dependent components makes it possible to use RMT in black box software. Disadvantage of RMT is extra computational efforts as compared with optimized multigrid.

Since coarse grid correction is computed on the single grid, it is possible to avoid the large communication overhead and the processor idleness on the very coarse grids in parallel multigrid.

The paper represents main components of RMT, results of numerical experiments, convergence and complexity analysis, estimation of speed-up and efficiency of parallel algorithm.

1 INTRODUCTION

Classical multigrid methods are among the fastest numerical algorithms for the solution of boundary value problems. Multigrid methods are characterized by their so-called components: the smoothing procedure, the coarsening strategy, the coarse grid operators, the transfer operators from fine grids to coarse and from coarse to fine and the cycle type. These components have to be specified for each concrete problem. In *optimized multigrid* algorithms, one tries to tailor the components to the problem at hand in order to obtain the highest possible efficiency for the solution process [8]. In contrast with direct and many iterative methods, optimized multigrid offers capability of solving the boundary value problems with complexity and storage proportional to the number of unknowns.

On the other hand, the idea of *robust multigrid* algorithms is to choose the components independently of the given problem, uniformly for as large a class of problems as possible [8].

Various variants of the optimized multigrid methods have been proposed and developed [8]. Less progress has been made in development of the robust multigrid algorithms. In framework of the work a multigrid algorithm is called *robust* if
- it is efficient for a large class of problems
- it has the least number of the problem-dependent components
- it is highly parallel.

One approach consists of minimizing the number of problem-dependent components in multigrid solver. Basic idea of Robust Multigrid Technique (RMT) is to apply the essential multigrid principle (about approximation of the smooth part of the error on coarser grids [8]) in monogrid (or single-grid) algorithm [2, 3].

RMT has the advantage that the intergrid operators are trivial and problem-independent. In addition, smoothing on the single grid gives the most powerful coarse correction strategy to make the task of the smoother less demanding. As a result, weak (and well parallelizable) smoothers can be used in the multigrid technique. Block Seidel smoother can handle many applied problems in unified manner starting from the Poisson equation up to the saddle point problems. Absence of coarse grids leads to almost perfect load balance and low communication overhead in parallel multigrid. Disadvantage of RMT is extra computational efforts as compared with optimized multigrid.

Really RMT is an opportunity to accelerate convergence of Gauss-Seidel iterations at solving (non)linear boundary value problems on structures grids.

2 BASIC COMPONENTS OF ROBUST MULTIGRID TECHNIQUE

Basic principles of RMT for solving boundary value problems (BVP) will be explained by studying a one-dimensional model problem and traditional multigrid notation [5].

To overcome the problem of robustness, RMT consists of two parts: analytical (adaptation of BVPs to the technique by representation of the solution as a sum (Σ-modification) or a product (Π-modification) of two functions (coarse grid correction and approximation to the solution)) and computational (generation of the finest and coarse grids, control volume approximation of the modified BVPs on the grid hierarchy (multigrid structure) and solution of the discretized equations by original multigrid solver).

Main difficulty of the robust multigrid algorithm development is to reach close-to-optimal convergence rate with the least number of the problem-dependent components.

First stage of the computational part of RMT consists of the finest grid generation for the following control volume approximation of the modified boundary value problems. The finest

grid G_1^0 consists of two sets $G^{\rm v}(0;1)$ and $G^{\rm f}(0;1)$ of the grid points. In domain $\Omega = [0,1]$ the sets are defined as

$$G^{\rm v}(0;1) = \{x_i^{\rm v} \mid x_i^{\rm v} = h(i-1), \quad i = 1,2,\ldots,N_x^0+1, \quad h = 1/N_x^0\}\,,$$
$$G^{\rm f}(0;1) = \{x_i^{\rm f} \mid x_i^{\rm f} = 0.5\,(x_i^{\rm v} + x_{i+1}^{\rm v}), \quad i = 1,2,\ldots,N_x^0\}\,.$$

The finest grid G_1^0 with $N_x^0 = h^{-1} = 8$ is shown on Fig. 1, where h is a mesh size.

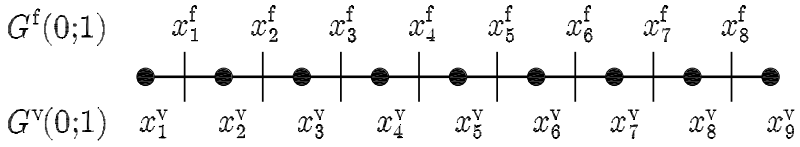

Figure 1: The finest grid G_1^0 with $N_x^0 = h^{-1} = 8$

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 Fig. 2. Hereinafter we will use standard multigrid notation for simplicity of RMT description.

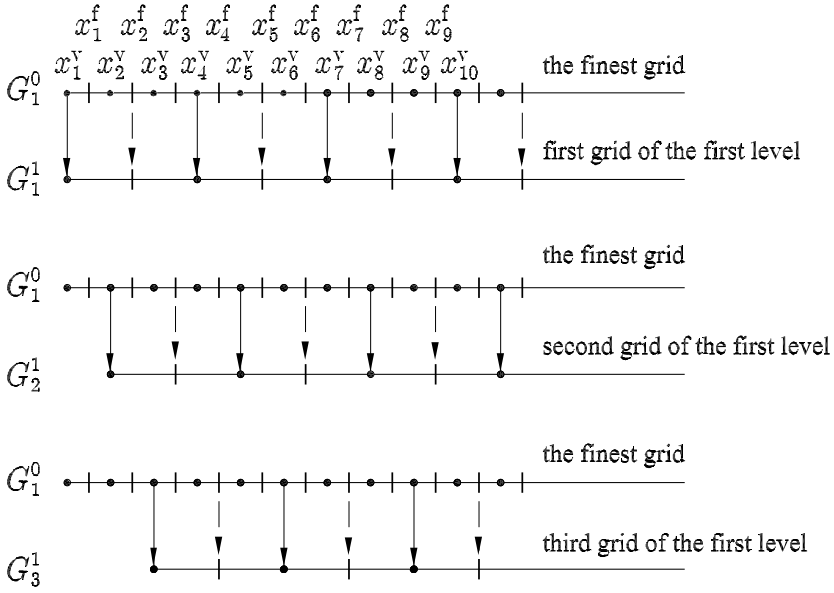

Figure 2: Triple coarsening in Robust Multigrid Technique

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 the mesh size on the finest grid.

4. the 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 (Fig. 3).

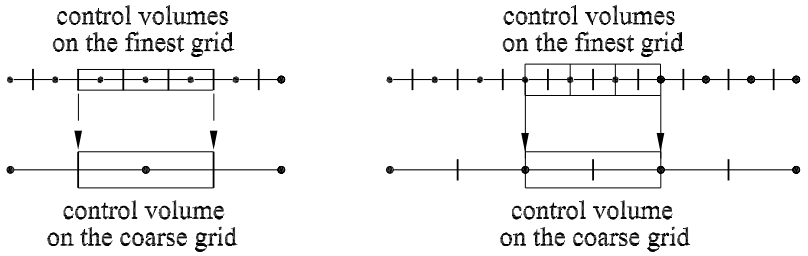

Figure 3: Control volumes on the finest and coarse grids

The finest grid forms the zero level and three coarse grids form the first level. The coarse grid generation is further recurrently repeated: each computational grid G_i^l, $i = 1, \ldots, 3^l$ of a current level l is considered to be the finest grid for the coarse grids G_j^{l+1}, $j = 1, \ldots, 3^{l+1}$ 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. Union of the finest grid and all coarse grids will be called *the multigrid structure* (Fig. 4).

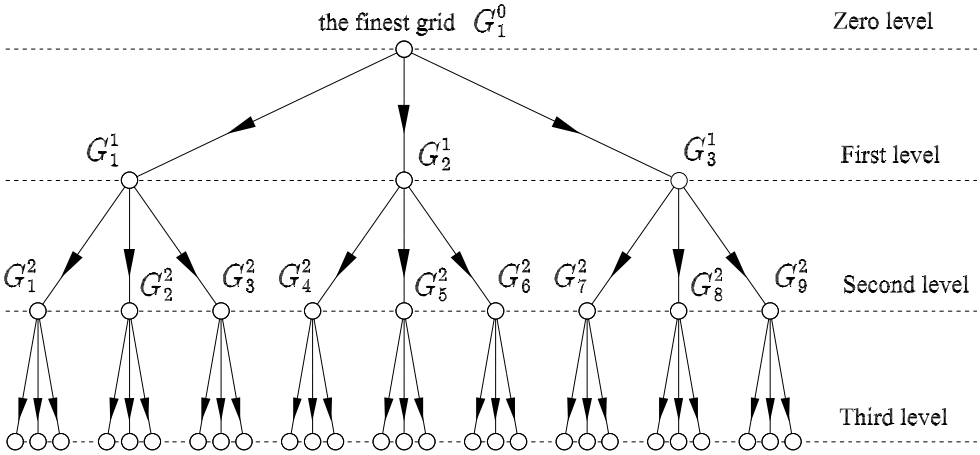

Figure 4: Multigrid structure

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

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

$$N_x^0 + 1 \approx 3^{L^+ + 1} \;\Rightarrow\; L^+ = \left[\frac{\lg(N_x^0 + 1)}{\lg 3} - 1 \right],$$

where square brackets mean integer part.

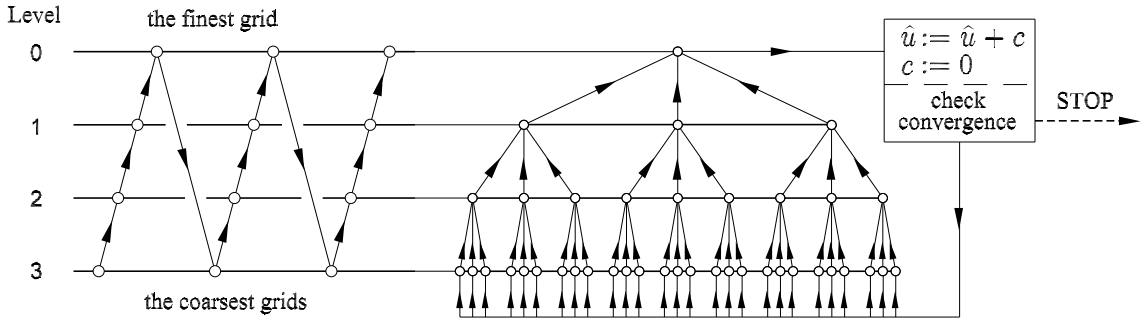

Figure 5: Multigrid cycle

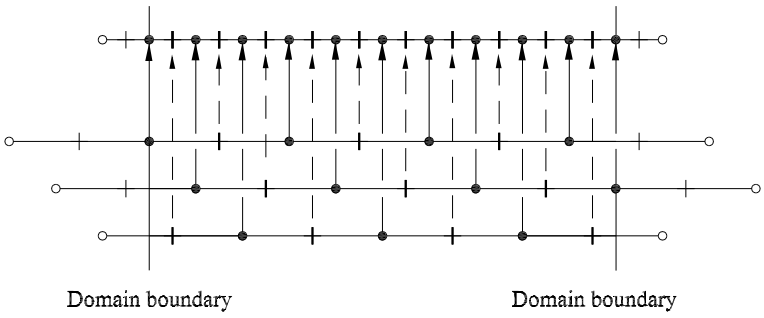

Figure 6: Transfer of correction from coarse grids to fine grid

Multigrid cycle of RMT is shown on Fig. 5. The multigrid schedule of RMT is sawtooth cycle, i.e. a special case of the V-cycle, in which smoothing before coarse grid correction (pre-smoothing) is deleted. Multigrid iterations 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 $\boldsymbol{c}$ as shown on Fig. 6. 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, after that the smoothing iterations). The coarse grid correction to be added to approximation of the solution $\hat{\boldsymbol{u}}$ on the finest grid is $\boldsymbol{c}$ ($\hat{\boldsymbol{u}} := \hat{\boldsymbol{u}} + \boldsymbol{c}$). The multigrid iterations repeatedly improve the approximation to the solution $\hat{\boldsymbol{u}}$ until the current approximation becomes accurate enough.

3 INTERGRID OPERATORS

Details of the control volume approximation on the multigrid structures are given in [5]. Here we focus attention on the intergrid operators of RMT in order to use their properties for the convergence analysis.

For reason of clearness consider one-dimensional Poisson equation

$$\frac{d^2u}{dx^2} = 10e^x, \qquad u(0) = u(1) = 0\,, \tag{1}$$

with exact solution

$$u(x) = 10\big(e^x + (1-e)x - 1\big)\,. \tag{2}$$

The solution $u(x)$ can be represented as

$$u(x) = c(x) + \hat{u}(x)\,, \tag{3}$$

where discrete analogues of the functions $c(x)$ and $\hat{u}(x)$ will be coarse grid correction and approximation to the solution, respectively. Substitution of (3) into (1) gives a Σ-modified form of the initial problem

$$\frac{d^2c}{dx^2} = 10e^x - \frac{d^2\hat{u}}{dx^2}, \qquad c(0) = c(1) = 0 \qquad \hat{u}(0) = \hat{u}(1) = 0\,. \tag{4}$$

The finest and coarse grids for the two-level algorithm are shown on Fig. 7. Let us define the index mapping to simplify operations with the multigrid structure. The notion "a grid of the level l" means one-to-one index mapping of the coarse grid points onto the indices of the finest grid points. In the rest of the paper, the mapping will be denoted by the braces $\{\}$. The index mapping of the grid points from the sets $G^{\rm v}$ and $G^{\rm f}$ is written as $x^{\rm v}_{\{i\}}$ and $x^{\rm f}_{\{i\}}$ respectively, where i and $\{i\}$ are the coarse and finest grid indices. The index mapping gives a close-to-the-finest-grid notation. Really the mapping shows that all derivatives and integrals are approximated on the finest grid, i.e. RMT uses only a single grid for solving BVP.

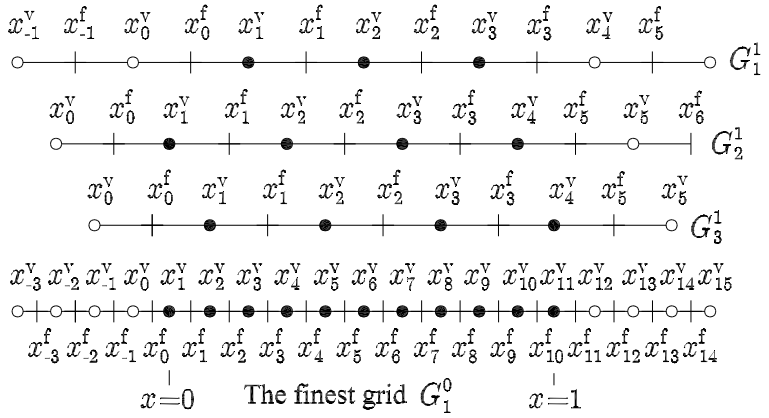

Figure 7: The finest (G_1^0) and three coarse grids (G_1^1, G_2^1 and G_3^1)

Let points $x^{\rm v}$ be vertices and points $x^{\rm f}$ be the control volume faces of the computational grids. Integration of (4) over the control volume $[x^{\rm f}_{\{i-1\}}, x^{\rm f}_{\{i\}}]$ gives

$$\frac{c_{\{i-1\}} - 2c_{\{i\}} + c_{\{i+1\}}}{h^2 3^{2l}} = \frac{1}{h3^l} \int\limits_{x^{\rm f}_{\{i-1\}}}^{x^{\rm f}_{\{i\}}} \left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right) dx\,, \tag{5}$$

where $h3^l$ is mesh size of the grids ($l = 0, 1$).

Fig. 7 shows that the index mapping for the coarse grids G_1^1 is given by

$$x^{\rm v}_{\{0\}} = x^{\rm v}_0, \qquad x^{\rm v}_{\{1\}} = x^{\rm v}_3, \qquad x^{\rm v}_{\{2\}} = x^{\rm v}_6, \qquad x^{\rm v}_{\{3\}} = x^{\rm v}_9, \qquad x^{\rm v}_{\{4\}} = x^{\rm v}_{12},$$
$$x^{\rm f}_{\{0\}} = x^{\rm f}_1, \qquad x^{\rm f}_{\{1\}} = x^{\rm f}_4, \qquad x^{\rm f}_{\{2\}} = x^{\rm f}_7, \qquad x^{\rm f}_{\{3\}} = x^{\rm f}_{10},$$

i.e. the first vertex of the coarse grid G_1^1 coincides with third vertex of the finest grid G_1^0 etc.

Right-hand side of (5) on the finest grid ($l = 0$) is approximated as

$$\frac{1}{h}\int\limits_{x_{i-1}^{\mathrm{f}}}^{x_i^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx \approx 10\,e^{x_i^{\mathrm{v}}} - \frac{\hat{u}_{i-1} - 2\hat{u}_i + \hat{u}_{i+1}}{h^2}\,.$$

Vector of resudials on the finest grid is defined by the components

$$r_1 = 0\,, \quad r_i = 10\,e^{x_i^{\mathrm{v}}} - \frac{\hat{u}_{i-1} - 2\hat{u}_i + \hat{u}_{i+1}}{h^2}\,, \quad r_{N_x^0+1} = 0\,. \tag{6}$$

Eq. (5) in the first vertex $x_{\{1\}}^{\mathrm{v}}$ of the coarse G_1^1 of the first level ($l = 1$) takes the form

$$\frac{c_{\{0\}} - 2c_{\{1\}} + c_{\{2\}}}{h^2 3^2} = \frac{1}{h3}\int\limits_{x_{\{0\}}^{\mathrm{f}}}^{x_{\{1\}}^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx\,.$$

Approximation of the boundary conditions on the coarse grids is discussed in [3, 5]. The equation

$$c_{\{0\}} = \frac{2}{\xi(\xi+1)}\;c\big|_{x=0} + 2\,\frac{\xi-1}{\xi}\,c_{\{1\}} - \frac{\xi-1}{\xi+1}\,c_{\{2\}} + O(h^3 3^{3l})\,,$$

is used for elimination of the coarse grid correction $c_{\{0\}}$ in the virtual vertex $x_{\{0\}}^{\mathrm{v}}$, where

$$\xi = \begin{cases} \dfrac{x_{\{1\}}^{\mathrm{v}}}{h\,3^l}\,, & x = 0 \\[2ex] \dfrac{1 - x_{\{N_k^l+1\}}^{\mathrm{v}}}{h\,3^l}\,, & x = 1 \end{cases}\,.$$

Here the value $c_{\{0\}}$ in the vertex $x_{\{0\}}^{\mathrm{v}}$ located outside the domain $\Omega = [0, 1]$ can be eliminated as follows

$$c_{\{0\}} = -c_{\{1\}} + \frac{1}{5}\,c_{\{2\}}\,.$$

Since the control volume $[x_{\{0\}}^{\mathrm{f}}, x_{\{1\}}^{\mathrm{f}}]$ is union of three control volumes of the finest grid (Fig. 3)

$$[x_{\{0\}}^{\mathrm{f}}, x_{\{1\}}^{\mathrm{f}}] = [x_1^{\mathrm{f}}, x_4^{\mathrm{f}}] = [x_1^{\mathrm{f}}, x_2^{\mathrm{f}}] \cup [x_2^{\mathrm{f}}, x_3^{\mathrm{f}}] \cup [x_3^{\mathrm{f}}, x_4^{\mathrm{f}}]\,,$$

the right-hand side can be rewritten as

$$\frac{1}{h3}\int\limits_{x_{\{0\}}^{\mathrm{f}}}^{x_{\{1\}}^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx = \frac{1}{3}\left(\frac{1}{h}\int\limits_{x_1^{\mathrm{f}}}^{x_2^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx + \right.$$

$$\left. + \frac{1}{h}\int\limits_{x_2^{\mathrm{f}}}^{x_3^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx + \frac{1}{h}\int\limits_{x_3^{\mathrm{f}}}^{x_4^{\mathrm{f}}}\left(10\,e^x - \frac{d^2\hat{u}}{dx^2}\right)dx\right) = \frac{1}{3}\big(r_2 + r_3 + r_4\big)\,.$$

Therefore (5) in the first vertex $x^{\mathrm{v}}_{\{1\}}$ of the coarse grid G^1_1 takes the final form

$$\frac{-3\,c_{\{1\}} + 1.2\,c_{\{2\}}}{9h^2} = \frac{1}{3}\big(r_2 + r_3 + r_4\big)\,.$$

Continuing the same considerations for the vertices $x^{\mathrm{v}}_{\{2\}}$ and $x^{\mathrm{v}}_{\{3\}}$ of the coarse grid G^1_1, we obtain the following system of linear equations

$$\frac{1}{9h^2}\begin{pmatrix} -3 & 1.2 & 0 \\ 1 & -2 & 1 \\ 0 & 1.2 & -3 \end{pmatrix}\begin{pmatrix} c_{\{1\}} \\ c_{\{2\}} \\ c_{\{3\}} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} r_2 + r_3 + r_4 \\ r_5 + r_6 + r_7 \\ r_8 + r_9 + r_{10} \end{pmatrix}. \tag{7}$$

Analogously we have for the coarse grid G^1_2

$$\frac{1}{9h^2}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1.5 & -6 \end{pmatrix}\begin{pmatrix} c_{\{1\}} \\ c_{\{2\}} \\ c_{\{3\}} \\ c_{\{4\}} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} 0 \\ r_3 + r_4 + r_5 \\ r_6 + r_7 + r_8 \\ r_9 + r_{10} + r_{11} \end{pmatrix}, \tag{8}$$

and for the coarse grid G^1_3

$$\frac{1}{9h^2}\begin{pmatrix} -6 & 1.5 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} c_{\{1\}} \\ c_{\{2\}} \\ c_{\{3\}} \\ c_{\{4\}} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} r_1 + r_2 + r_3 \\ r_4 + r_5 + r_6 \\ r_7 + r_8 + r_9 \\ 0 \end{pmatrix}. \tag{9}$$

Solutions of the systems (7), (8) and (9) with a zero starting guess ($\hat{u} = 0$) are shown on Fig. 8. In RMT more computational work must be spent on coarse grids in order to allow for the best approximation to the solution on the fine grids. Multiple coarse grid correction strategy used in RMT makes it possible to eliminate the problem-dependent interpolation from the multigrid algorithm. An efficient procedure for the fast integral evaluation on the multigrid structure is given in [5, 6]. Virtual vertices on all grids are needed only for this procedure.

Systems (7), (8) and (9) can be incorporated into common system for convergence analysis of RMT. Coefficient matrix of the common system has a block structure, where the number of blocks is equal to the number of coarse grids. For this example, we have

$$A_1 = \frac{1}{9h^2}\begin{pmatrix} -3 & 1.2 & 0 & & & & & & & & & \\ 1 & -2 & 1 & & & & & & & & & \\ 0 & 1.2 & -3 & & & & & & & & & \\ & & & 1 & 0 & 0 & 0 & & & & & \\ & & & 1 & -2 & 1 & 0 & & & & & \\ & & & 0 & 1 & -2 & 1 & & & & & \\ & & & 0 & 0 & 1.5 & -6 & & & & & \\ & & & & & & & -6 & 1.5 & 0 & 0 \\ & & & & & & & 1 & -2 & 1 & 0 \\ & & & & & & & 0 & 1 & -2 & 1 \\ & & & & & & & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{10}$$

Total number of the vertices on all coarse grids of level l ($l = 1, 2, \ldots, L^+$) coincides with the number of the vertices on the finest grid ($l = 0$).

Right-hand side vector of the common system is

$$\mathcal{R}_{0\to 1}\boldsymbol{r}_0 = \frac{1}{3}\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7 \\ r_8 \\ r_9 \\ r_{10} \\ r_{11} \end{pmatrix}, \tag{11}$$

where components of the residual vector $\boldsymbol{r}$ are computed on the finest grid as (6), $\mathcal{R}_{0\to 1}$ is an restriction operator transferring the residual from the finest grid (zero level) to the coarse grids of the first level. It is clear that the restriction operator is the problem-independent component of RMT.

Vector of unknowns of the common system is union of vectors of systems (7), (8) and (9). Taking into account the index mapping, we have

$$\boldsymbol{c}_1 = \begin{pmatrix} c_{\{1\}}(G_1^1) \\ c_{\{2\}}(G_1^1) \\ c_{\{3\}}(G_1^1) \\ c_{\{1\}}(G_2^1) \\ c_{\{2\}}(G_2^1) \\ c_{\{3\}}(G_2^1) \\ c_{\{4\}}(G_2^1) \\ c_{\{1\}}(G_3^1) \\ c_{\{2\}}(G_3^1) \\ c_{\{3\}}(G_3^1) \\ c_{\{4\}}(G_3^1) \end{pmatrix} = \begin{pmatrix} c_3 \\ c_6 \\ c_9 \\ c_1 \\ c_4 \\ c_7 \\ c_{10} \\ c_2 \\ c_5 \\ c_8 \\ c_{11} \end{pmatrix} \tag{12}$$

Here $c_{\{1\}}(G_1^1)$ is value of the correction in the first vertex of the coarse grid G_1^1 etc. as shown on Fig. 9.

Prolongation in the example can be written as

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_8 \\ c_9 \\ c_{10} \\ c_{11} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} c_3 \\ c_6 \\ c_9 \\ c_1 \\ c_4 \\ c_7 \\ c_{10} \\ c_2 \\ c_5 \\ c_8 \\ c_{11} \end{pmatrix} \tag{13}$$

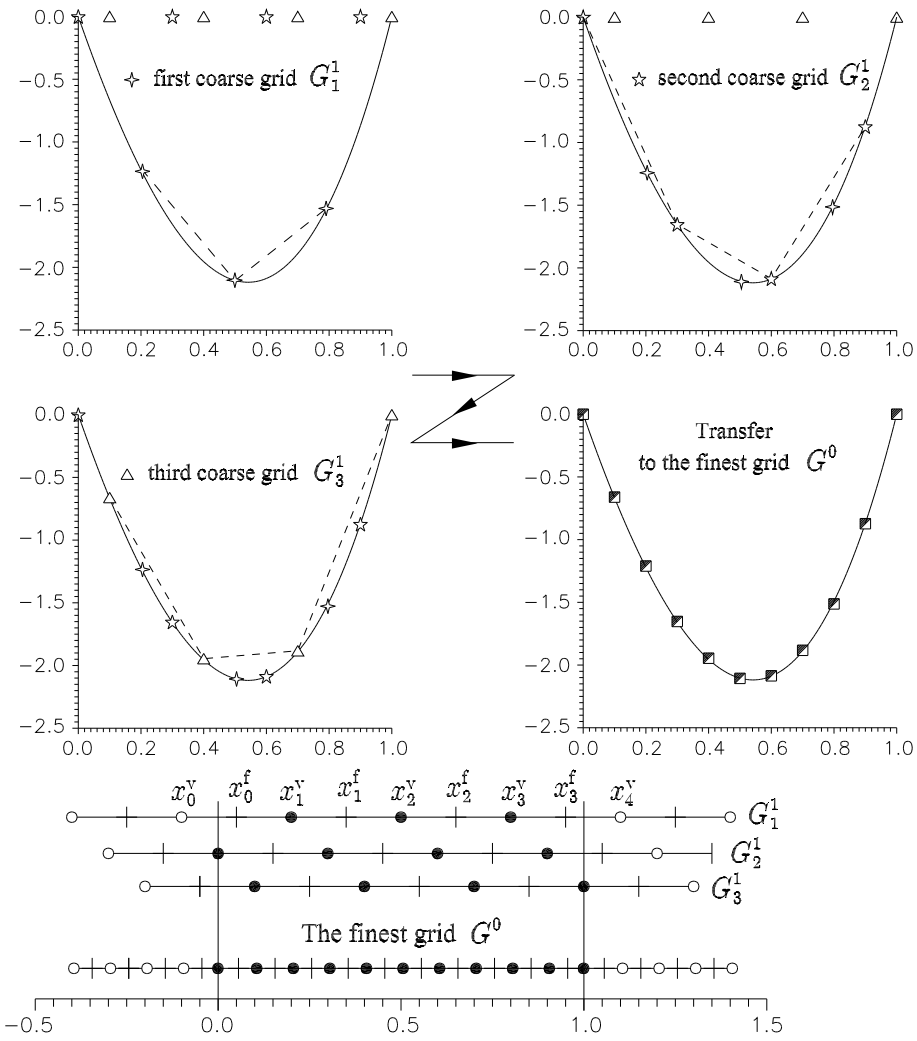

Figure 8: Solutions of the systems (7), (8) and (9) (dotted lines) vs. exact solution (2) (solid line)

or in matrix form

$$\boldsymbol{c}_0 = \mathcal{P}_{1\to 0}\boldsymbol{c}_1, \tag{14}$$

where $\mathcal{P}_{1\to 0}$ is the problem-independent prolongation operator (permutable matrix). Note that $\mathcal{P}_{1\to 0}^{-1} = \mathcal{P}_{1\to 0}^{\mathrm{T}}$.

We summarize some abovementioned results. Approximation of the Σ-modified BVP (4) on three coarse grids of the first level results in the common system of linear equation

$$A_1\boldsymbol{c}_1 = \mathcal{R}_{0\to 1}\boldsymbol{r}_0, \tag{15}$$

where the block-structured coefficient matrix A_1, vector of unknowns $\boldsymbol{c}_1$ and right-hand side vector $\mathcal{R}_{0\to 1}\boldsymbol{r}_0$ are given by (10), (12) and (11), respectively. Assume that the system (15) is solved by direct method, i.e.

$$\boldsymbol{c}_1 = A_1^{-1}\mathcal{R}_{0\to 1}\boldsymbol{r}_0,$$

where the vector $\boldsymbol{c}_1$ is defined by (12). After that the vector $\boldsymbol{c}_1$ is prolongated from the grids of the first level up to the finest grid as (13) (or in the matrix form (14)).

All matrices and all vectors in (15) have no "coarse grid variables" because RMT is the single-grid algorithm. Size of all matrices and all vectors is l-independent. It should be em-

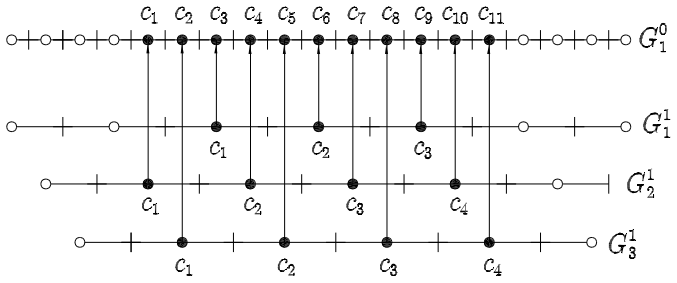

Figure 9: Prolongation of the correction in the one-dimensional example

phasized that multigrid notation is used only for clearness of the description and to follow traditional terminology.

Let us summarize main properties of the problem-independent transfer operators of RMT:

$$\mathcal{R}_{0\to l} = \begin{cases} I, & l = 0 \\ \prod\limits_{k=1}^{l} \mathcal{R}_{l-k\to l-k+1}, & l = 1, 2, \ldots, L^+ \end{cases},$$

$$\mathcal{P}_{l\to 0} = \begin{cases} I, & l = 0 \\ \prod\limits_{k=1}^{l} \mathcal{P}_{k\to k-1}, & l = 1, 2, \ldots, L^+ \end{cases},$$

where I is unit operator.

4 CONVERGENCE ANALYSIS AND COMPLEXITY

Approximation of some linear Σ-modified BVP on the multigrid structure results to series of the system of linear equations of the same size

$$A_l \boldsymbol{c}_l = \mathcal{R}_{0\to l}\big(\boldsymbol{b}_0 - A_0 \hat{\boldsymbol{\varphi}}_0^{(q)}\big), \qquad l = 0, 1, \ldots, L^+,$$

with problem-independent transfer operators $\mathcal{R}_{0\to l}$ and $\mathcal{P}_{l\to 0}$. Here $\hat{\boldsymbol{\varphi}}_0^{(q)}$ is an approximation to the solution $\hat{\boldsymbol{\varphi}}_0 = A_0^{-1}\boldsymbol{b}_0$ after q multigrid iterations and subscript indicates the level index.

Let the smoothers be abbreviated as

$$W_l\big(\boldsymbol{c}_l^{(\nu_l+1)} - \boldsymbol{c}_l^{(\nu_l)}\big) = \mathcal{R}_{0\to l}\big(\boldsymbol{b}_0 - A_0 \hat{\boldsymbol{\varphi}}_0^{(q)}\big) - A_l \boldsymbol{c}_l^{(\nu_l)}$$

or

$$\boldsymbol{c}_l^{(\nu_l+1)} = S_l \boldsymbol{c}_l^{(\nu_l)} + W_l^{-1}\mathcal{R}_{0\to l}\big(\boldsymbol{b}_0 - A_0 \hat{\boldsymbol{\varphi}}_0^{(q)}\big),$$

where $S_l = I - W_l^{-1}A_l$ is the smoothing iteration matrix. We assume that the smoother is convergent iterative method ($\|S_l\| < 1$) and the same number of the smoothing iterations (ν_l) is performed on all grids G_k^l, $k = 1, 2, \ldots, 3^{dl}$ of the level l ($l = 0, 1, 2, \ldots, L^+$, $d = 2, 3$).

Multigrid iterations of RMT is given by

$$\hat{\boldsymbol{\varphi}}_0^{(q+1)} = Q_0 A_0 \hat{\boldsymbol{\varphi}}_0^{(q)} + \big(A_0^{-1} - Q_0\big)\boldsymbol{b}_0,$$

where the matrix Q_0 is defined as

$$Q_l = \begin{cases} S_l^{\nu_l}\Big(d_l\mathcal{R}_{0\to l} + \mathcal{P}_{l+1\to l}Q_{l+1}\Big), & l = 0,1,2,\ldots,L^+-2 \\ S_l^{\nu_l} d_l\mathcal{R}_{0\to l}, & l = L^+-1 \end{cases},$$

$$d_l = A_l^{-1} - \mathcal{P}_{l+1\to l}A_{l+1}^{-1}\mathcal{R}_{l\to l+1}.$$

It is clear that the multigrid iteration matrix is $M = Q_0A_0$, for the details see [5].

Following W. Hackbusch [1], convergence proof of the multigrid methods is based on the definitions:

1. Smoothing property: the matrix S has the smoothing property if there exist a monotonically decreasing function $\eta(\nu_l) : \mathbb{R}_+ \to \mathbb{R}_+$ independent of the mesh size h such that $\eta(\nu_l) \to 0$ at $\nu_l \to \infty$ and

$$\|A_lS_l^{\nu_l}\| \leqslant \eta(\nu_l)\|A_l\|. \tag{16}$$

2. Approximation property: the approximation property holds if there exists a constant $C_A > 0$ independent of the mesh size h such that

$$\|A_l^{-1} - \mathcal{P}_{l+1\to l}A_{l+1}^{-1}\mathcal{R}_{l\to l+1}\| \leqslant C_A\|A_l\|^{-1}. \tag{17}$$

Approximation and smoothing properties should be proved for each problem.

Now we use the multigrid iteration matrix $M = Q_0A_0$ to derive the convergence theorem:

Convergence theorem. Assume the smoothing (16) and approximation (17) properties hold and $\|\mathcal{R}_{0\to l}\| \leqslant C_{\mathcal{R}}$. Then we have h-independent convergence of RMT and

$$\|M\|_2 \leqslant C_A\eta(\nu_0) + C_AC_{\mathcal{R}}\sum_{l=1}^{L^+-1} C^l\eta(\nu_l).$$

The theorem shows that the number of the multigrid iterations is independent of the mesh size h, but computational cost of the iterations is not optimal, i.e. the number of arithmetic operations per the multigrid iteration step is proportional to $N(L^+ + 1)$, where N is the number of grid points and $L^+ + 1$ is the number of grid levels. Since $L^+ + 1 \sim \lg N$, the number of arithmetic operations needed for one multigrid cycle of RMT is $O(N \lg N)$.

We represent results of a standard multigrid solver (V-cycle) and RMT applied to the model problem of the 2D Poisson equation:

$$-\Delta u = f \quad \text{in} \quad \Omega = (0,1)^2,$$

$$u = 0 \quad \text{on} \quad \partial\Omega,$$

with exact solution

$$u(x,y) = Q(x)Q(y), \quad \text{where} \quad Q(\beta) = 10\big(e^\beta + (1-e)\beta - 1\big), \quad \beta = (x,y).$$

In the experiment we use a starting guess $\hat{\boldsymbol{u}}^{(0)} = 0$ and the computational grids 129×129 ($h = 1/128$) and 1025×1025 ($h = 1/1024$). The rate of convergence is measured by looking the residual norm

$$\|\boldsymbol{b}_0 - A_0\hat{\boldsymbol{u}}^{(q)}\|_\infty = \|\boldsymbol{r}^{(q)}\|_\infty, \tag{18}$$

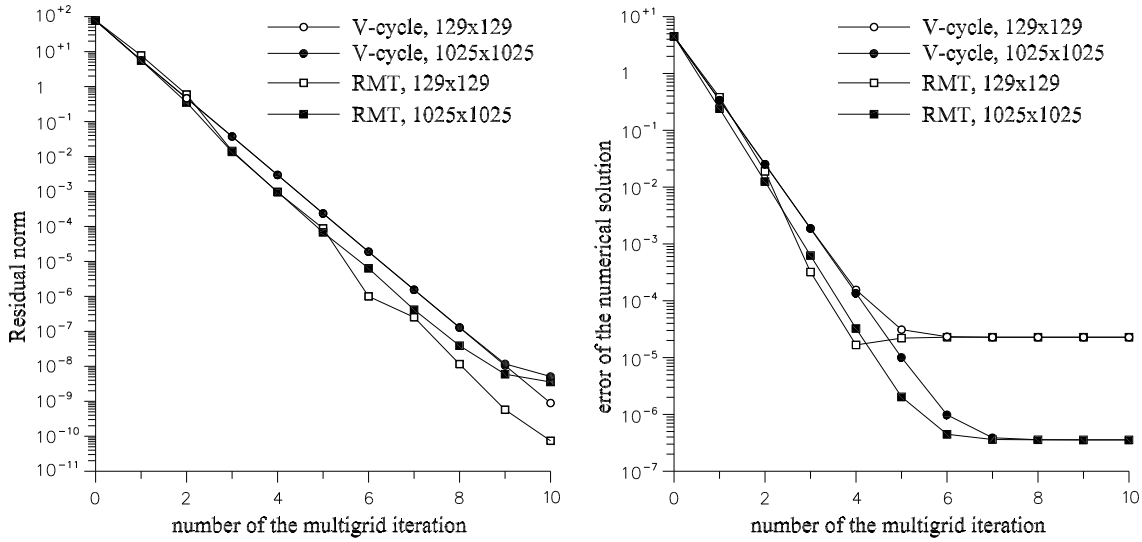

Figure 10: Comparison of the multigrid convergence: V-cycle vs. RMT

and error of the numerical solution defined as maximum of the absolute difference between the exact and numerical solutions, i.e.

$$\max_{ij} |Q(x_i)Q(y_j) - \hat{u}_{ij}^{(q)}|.$$

Fig. 10 represents reduction of the residual norm (18) and the error of the numerical solution as a function of the multigrid iteration q. Point Gauss–Seidel method is used as a smoother. V-cycle is implemented with three pre- and post-smoothing iterations and RMT is implemented with six (post)smoothing iterations. It can be seen that there is no noticeable difference between the convergence behaviors of the multigrid algorithms, but implementation of RMT iterations requires more computational work. Assuming that the computational grid consists of $(2^k + 1)^d$ vertices ($d = 2, 3$) and the computational costs of the transfer operators are negligible compared to the cost of the smoothing iterations, augmentation of the computational efforts in RMT as compared with V-cycle is estimated as

$$\approx \left[\frac{\lg(2^k + 1)}{\lg 3}\right] \frac{2^d - 1}{2^d + 1} \approx \Big[0.63k\Big] \frac{2^d - 1}{2^d + 1}$$

arithmetic operations, where square brackets mean integer part.

5 ROBUST SMOOTHER

Robustness and efficiency of a multigrid method are strongly influenced by the smoother used. Users of multigrid methods employ many different smoothing methods because efficiency of the smoothers are problem-dependent.

Let 2D discrete boundary value problem arising from a five-point discretization be abbreviated as

$$A_{ij}^w u_{i-1j} + A_{ij}^e u_{i+1j} + A_{ij}^s u_{ij-1} + A_{ij}^n u_{ij+1} + A_{ij}^p u_{ij} = b_{ij}.$$

Firstly, we consider a point Gauss-Seidel iteration, which consists of updating the unknown corresponding to the nodes located at the center of the five-point stencil:

$$u_{ij} = \frac{1}{A_{ij}^p}\Big(b_{ij} - A_{ij}^w u_{i-1j} - A_{ij}^e u_{i+1j} - A_{ij}^s u_{ij-1} - A_{ij}^n u_{ij+1}\Big).$$

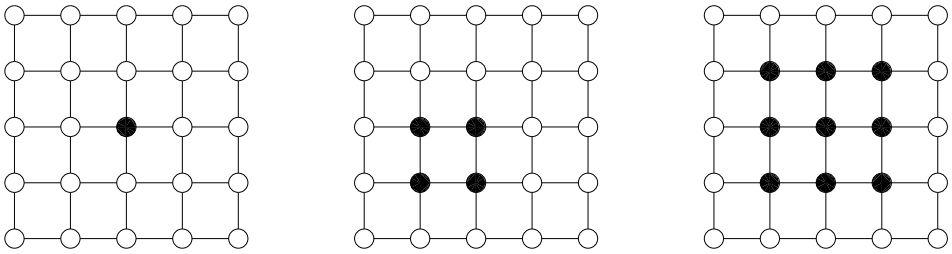

Figure 11: Block ordering of unknowns: blocks 1×1, 2×2 and 3×3

There are many variants of point-type smoothers, that can differ in the choice of the unknown ordering. Simple point-wise smoothers (with any ordering) are not appropriate for the saddle point type problems (i.e. if $A^p_{ij} = 0$).

In general the block ordering of the unknowns can be used for construction of the Gauss–Seidel smoothers. It consists of decomposing the grid into small overlapping subdomains and looping over all of them solving the system arising from the equations corresponding to the points in the subdomain as shown on Fig 11. Point-wise smoothers can be considered as a block iterative methods with blocks 1×1.

Application of blocks 2×2 give the following system

$$\begin{pmatrix} A^p_{ij} & A^e_{ij} & A^n_{ij} & 0 \\ A^w_{i+1j} & A^p_{i+1j} & 0 & A^n_{i+1j} \\ A^s_{ij+1} & 0 & A^p_{ij+1} & A^e_{ij+1} \\ 0 & A^s_{i+1j+1} & A^w_{i+1j+1} & A^p_{i+1j+1} \end{pmatrix} \begin{pmatrix} u_{ij} \\ u_{i+1j} \\ u_{ij+1} \\ u_{i+1j+1} \end{pmatrix} = \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \end{pmatrix},$$

where

$$\begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \end{pmatrix} = \begin{pmatrix} b_{ij} - A^w_{ij}\, u_{i-1j} - A^s_{ij}\, u_{ij-1} \\ b_{i+1j} - A^e_{i+1j}\, u_{i+2j} - A^s_{i+1j}\, u_{i+1j-1} \\ b_{ij+1} - A^w_{ij+1}\, u_{i-1j+1} - A^n_{ij+1}\, u_{ij+2} \\ b_{i+1j+1} - A^e_{i+1j+1}\, u_{i+2j+1} - A^n_{i+1j+1}\, u_{i+1j+2} \end{pmatrix}.$$

Solution of the system by a direct method (for example, Gaussian elimination) gives the updated values of u_{ij}, u_{i+1j}, u_{ij+1} and u_{i+1j+1}.

Also there are many variants of box-type smoothers, they can differ in the choice of the subdomains which are solved simultaneously, and in the way in which the local systems are solved. In addition, the different subdomains can be visited in different ways yielding to a wide variety of the box-smoothers.

Note that computational cost of the system solution is g^3 arithmetic operations, where g is the number unknowns in the block. In limit $g \to N$ the ordering results in a direct solver, where N is the number unknowns.

Compared to point-wise smoothers, the crucial advantage of the block smoothers is the ability to deal with zero elements appearing on the diagonal of the coefficient matrix. Very important partial case is a Vanka smoother developed to solve the saddle-point systems arising in the field of computational fluid dynamics, particularly for incompressible flow problems [9]. Other reasons are that it is not too difficult to implement and at the same time it is efficient and robust for a wide class of problem configurations.

6 NUMERICAL EXPERIMENTS

We consider the following partial differential equation (PDE)

$$\frac{\partial}{\partial x}\left(\lambda^x \frac{\partial w}{\partial x}\right) + \frac{\partial}{\partial y}\left(\lambda^y \frac{\partial w}{\partial y}\right) + \frac{\partial}{\partial z}\left(\lambda^z \frac{\partial w}{\partial z}\right) = -f(x,y,z)$$

in the unit cube $\Omega = (0,1)^3$ for the illustration of the RMT convergence. Zero starting guess and uniform grid $101 \times 101 \times 101$ ($h = 1/100$) is used in the numerical experiments. Convergence rate of RMT is illustrated by the average reduction factor of the residual

$$\rho^{(q)} = \left(\frac{\|A_0 \hat{\boldsymbol{u}}^{(q)} - \boldsymbol{b}_0\|_\infty}{\|A_0 \hat{\boldsymbol{u}}^{(0)} - \boldsymbol{b}_0\|_\infty}\right)^{1/q}$$

with q is the number of the multigrid iteration, $\hat{\boldsymbol{u}}^{(0)} = 0$ is a starting guess. The average reduction factor is computed after five multigrid iterations ($q = 5$).

Example 1: Poisson equation ($\lambda^x = \lambda^y = \lambda^z = 1$). Exact solution of the Poisson equation is taken as

$$w_e = e^{x+y+z}, \tag{19}$$

and the error of the numerical solution is defined as

$$e^{(q)} = \max_{ijk} |w_e(x_i, y_j, z_k) - \hat{u}_{ijk}|.$$

Results of the numerical test are given in Table 1. Point Gauss-Seidel (blocks $1 \times 1 \times 1$) and block Gauss-Seidel (blocks $3 \times 3 \times 3$) methods with natural unknown ordering are used as the smoothers in the test. Execution time of the multigrid algorithm with four point Gauss-Seidel smoothing iterations ($\nu = 4$) is taken as a work unit (WU=1) in the numerical experiments. It is clear that RMT works efficiently for the Poisson equation.

Block	ν	$e^{(q)}$	$\rho^{(q)}$	WU
$1 \times 1 \times 1$	4	$7.30 \cdot 10^{-6}$	0.084	1.00
$1 \times 1 \times 1$	5	$7.30 \cdot 10^{-6}$	0.033	1.16
$1 \times 1 \times 1$	6	$7.30 \cdot 10^{-6}$	0.023	1.35
$3 \times 3 \times 3$	3	$7.30 \cdot 10^{-6}$	0.055	1.03
$3 \times 3 \times 3$	4	$7.30 \cdot 10^{-6}$	0.033	1.32
$3 \times 3 \times 3$	5	$7.30 \cdot 10^{-6}$	0.021	1.62

Table 1: Convergence of RMT in **Example 1** ($q = 5$)

Example 2: anisotropic equation ($\lambda^x = \lambda^y = 1, \lambda^z = 0.1$) with the same exact solution (19). Table 2 represents results of the test. Point Gauss-Seidel smoother does not work efficiently because the discrete variables are linked weakly in Z direction ($\lambda^z \ll \lambda^x = \lambda^y$). Application of Gauss-Seidel smoother with the block ordering of the unknowns (blocks $5 \times 5 \times 1$) shows the best results because the smoother takes into account the problem anisotropy.

Example 3: equation with discontinuous coefficients. Assume that unit cube Ω consists of two materials with heat conductivity coefficients λ_i (internal subdomain $\tilde{\Omega}$) and λ_e (external

Block	ν	$e^{(q)}$	$\rho^{(q)}$	WU
$1 \times 1 \times 1$	4	$3.29 \cdot 10^{-3}$	0.752	1.00
$1 \times 1 \times 1$	5	$1.44 \cdot 10^{-3}$	0.637	1.16
$1 \times 1 \times 1$	6	$6.25 \cdot 10^{-4}$	0.538	1.35
$3 \times 3 \times 3$	3	$1.77 \cdot 10^{-4}$	0.522	1.03
$3 \times 3 \times 3$	4	$2.73 \cdot 10^{-5}$	0.333	1.32
$3 \times 3 \times 3$	5	$9.90 \cdot 10^{-6}$	0.216	1.62
$5 \times 5 \times 1$	3	$1.08 \cdot 10^{-5}$	0.268	0.81
$5 \times 5 \times 1$	4	$7.87 \cdot 10^{-6}$	0.142	1.01
$5 \times 5 \times 1$	5	$7.81 \cdot 10^{-6}$	0.081	1.21

Table 2: Convergence of RMT in **Example 2** ($q = 5$)

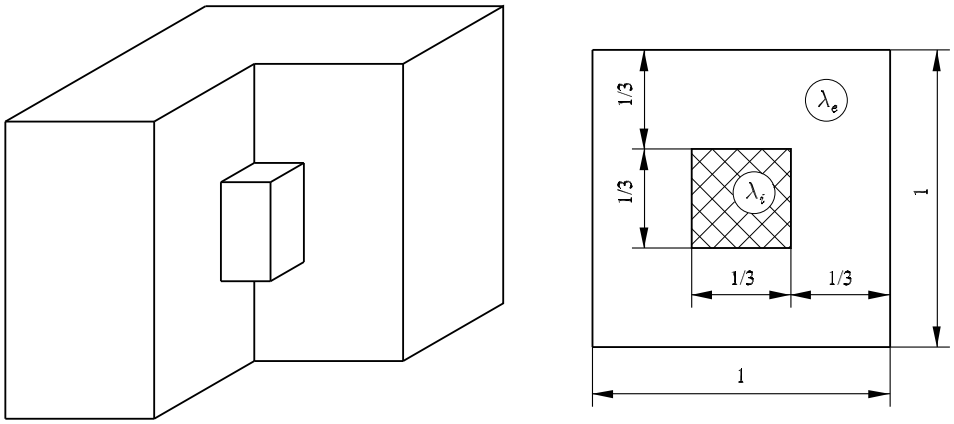

Figure 12: Geometry of the problem with the discontinuous coefficients

subdomain) as shown on Fig. 12, i.e.

$$\lambda^x(x,y,z) = \lambda^y(x,y,z) = \lambda^z(x,y,z) = \begin{cases} \lambda_i, & (x,y,z) \in \tilde{\Omega} \\ \lambda_e, & (x,y,z) \notin \Omega \end{cases}.$$

In case of $\lambda_i = \lambda_e = 1$, the exact solution is given by (19).

Results of the test presented in Table 3 show that the block ordering of the unknowns gives more efficient smoother as compared with the point ordering.

Example 4: nonlinear equation. Let the coefficients λ^x, λ^y and λ^z are defined as

$$\lambda^x = w^{-a}, \quad \lambda^y = w^{-b}, \quad \lambda^z = w^{-c},$$

where a, b and c are some positive constants. Linearization of the discrete nonlinear equation is based on the computation of the coefficients using correction and approximation to the solution obtained on previous level, i.e.

$$\lambda_l^x = (c_{l+1} + \hat{u}_{l+1})^{-a}, \quad \lambda_l^y = (c_{l+1} + \hat{u}_{l+1})^{-b}, \quad \lambda_l^z = (c_{l+1} + \hat{u}_{l+1})^{-c}.$$

Solution of the nonlinear problem with exact solution (19) starts from $\hat{u}^{(0)} = 0$ and $\lambda_x^{(0)} = 1$, $\lambda_y^{(0)} = 1$ and $\lambda_z^{(0)} = 1$. Typical convergence rate of RMT is shown in Table 4. It is easy to see that RMT with the block smoother is an efficient solver for the nonlinear problem.

Block	λ_e	λ_i	ν	$\rho^{(q)}$	WU
$1 \times 1 \times 1$	1	10^{+0}	6	0.023	1.92
$1 \times 1 \times 1$	1	10^{-1}	6	0.228	1.92
$1 \times 1 \times 1$	1	10^{-2}	6	0.198	1.92
$1 \times 1 \times 1$	1	10^{-3}	6	0.227	1.92
$1 \times 1 \times 1$	1	10^{-4}	6	0.273	1.92
$1 \times 1 \times 1$	1	10^{-5}	6	0.410	1.92
$3 \times 3 \times 3$	1	10^{+0}	5	0.021	2.05
$3 \times 3 \times 3$	1	10^{-1}	5	0.193	2.05
$3 \times 3 \times 3$	1	10^{-2}	5	0.170	2.05
$3 \times 3 \times 3$	1	10^{-3}	5	0.150	2.05
$3 \times 3 \times 3$	1	10^{-4}	5	0.149	2.05
$3 \times 3 \times 3$	1	10^{-5}	5	0.149	2.05

Table 3: Convergence of RMT in **Example 3** ($q = 5$)

Block	a	b	c	ν	$e^{(q)}$	$\rho^{(q)}$	WU
$1 \times 1 \times 1$	0.25	0.50	0.75	4	$9.00 \cdot 10^{-6}$	0.317	1.44
$1 \times 1 \times 1$	0.25	0.50	0.75	5	$2.83 \cdot 10^{-6}$	0.169	1.77
$1 \times 1 \times 1$	0.25	0.50	0.75	6	$2.40 \cdot 10^{-6}$	0.115	2.01
$3 \times 3 \times 3$	0.25	0.50	0.75	3	$6.01 \cdot 10^{-6}$	0.256	1.55
$3 \times 3 \times 3$	0.25	0.50	0.75	4	$2.27 \cdot 10^{-6}$	0.110	1.80
$3 \times 3 \times 3$	0.25	0.50	0.75	5	$2.14 \cdot 10^{-6}$	0.075	2.06

Table 4: Convergence of RMT in **Example 4** ($q = 5$)

Example 5: incompressible Navier-Stokes equations. In the application of RMT to the cavity problem shown on Fig. 13, a number of the flow Reynolds numbers (100 and 500) and the computational grids (101×101 and 1001×1001) have been considered. Inlet and outlet velocity components are distributed under the parabolic law:

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

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

on the other boundaries no-slip conditions are given. Detailed description of the Σ-modification of the Navier-Stokes equations and control volume approximation of the modified Navier-Stokes equations on the multigrid structures are given in [5, 6]. Σ-modification makes it possible to solve discrete Navier-Stokes equations without FAS. Full Vanka method is used as a smoother in the test. Fig. 14 and 15 represent the distribution of the stream function isolines and isobars. Results of test shown in Table 5 demonstrate robustness and efficiency of RMT with full Vanka smoother for the given benchmark problem. Obtained convergence rate depends weakly on the Reynolds number and the mesh size of the computational grids.

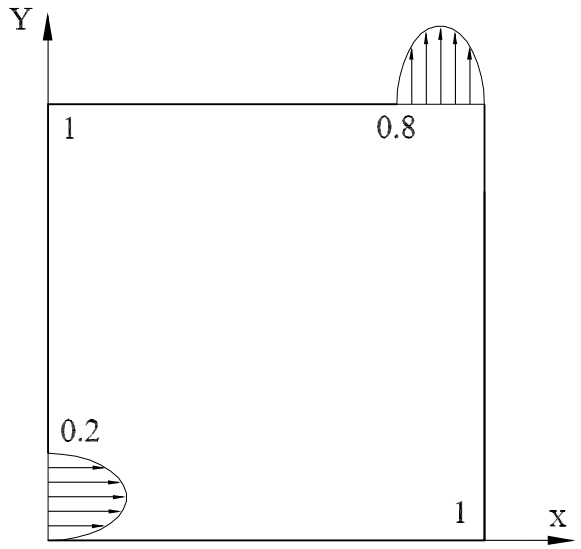

Figure 13: Geometry of the cavity

Grid	Re	ν	$\rho^{(q)}$
101×101	100	4	0.132
1001×1001	100	4	0.151
101×101	500	4	0.167
1001×1001	500	4	0.194

Table 5: Convergence of RMT in **Example 5**

The numerical tests illustrate that the problem-dependent components of RMT are:
1) the number of smoothing iterations;
2) unknown ordering (for anisotropic problems);
3) underrelaxation parameter (for nonlinear problems);
4) stopping criterion of the multigrid iterations.

7 PARALLEL MULTIGRID

Parallelization of classic multigrid methods (CMM) follows in the standard fashion of dividing the domain into subdomains (one for each processor). Each processor is then responsible for updating of the unknowns associated within its subdomain only. However parallel efficiency of CMM can degrade due to coarse grid smoothing. There can be reached situations where the number of processors exceeds the number of coarse grid points. As a result, some processors are idle during these computations. One approaches to overcome poor computation-to-communication ratio on the coarse grids is multiple coarse grid corrections. Nevertheless the choice of appropriate multigrid components and their efficient parallel implementation is highly problem-dependent in CMM [8].

There are two other common measures of parallelism that we now introduce:

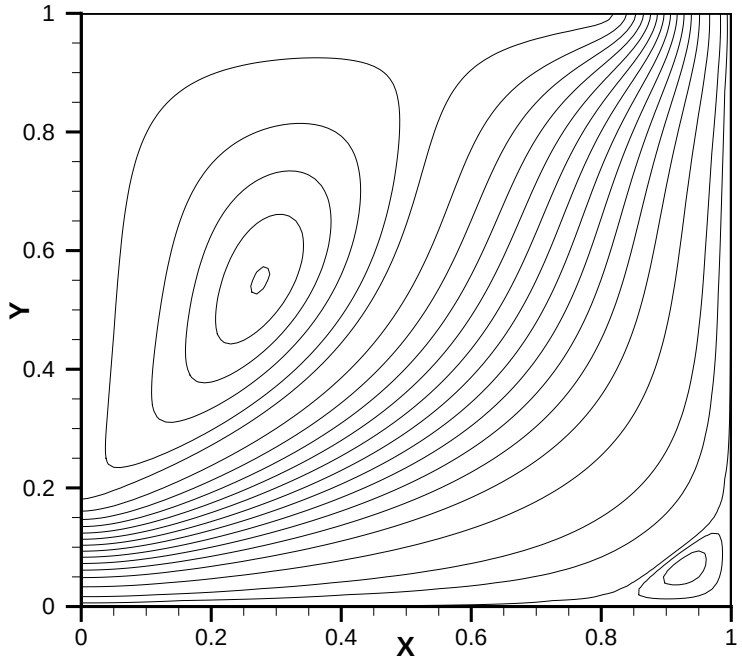

Figure 14: Isolines of the stream functions ($Re = 100$)

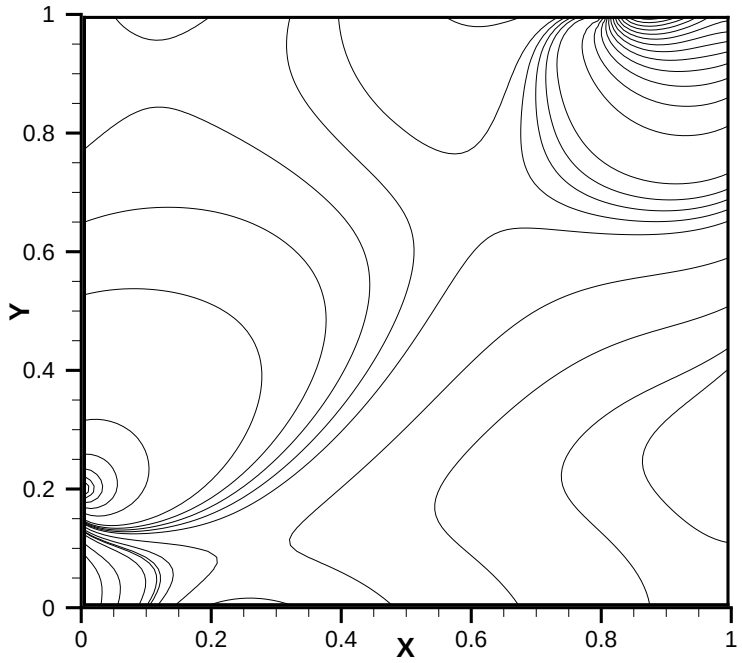

Figure 15: Isobars ($Re = 100$)

Definition 1. The speedup ($\bar{\mathcal{S}}$) and the efficiency ($\bar{\mathcal{E}}$) of a parallel algorithm is

$$\bar{\mathcal{S}} = p\bar{\mathcal{E}} = \frac{T(1)}{T(p)}, \tag{20}$$

where $T(1)$ is the execution time for a single processor and $T(p)$ is the execution time using p processors.

Definition 2. The speedup ($\tilde{\mathcal{S}}$) and the efficiency ($\tilde{\mathcal{E}}$) of a parallel algorithm over the best sequential algorithm is

$$\tilde{\mathcal{S}} = p\tilde{\mathcal{E}} = \frac{\tilde{T}(1)}{T(p)}, \tag{21}$$

where $\tilde{T}(1)$ is the execution time on a single processor of fastest sequential algorithm and $T(p)$ is the execution time of the parallel algorithm on p processors.

General description of the common measures of parallelism is given in [7] and the references therein. In addition to these common measures, we introduce the measures of parallel properties of smoother:

Definition 3. The speedup ($\mathcal{S}$) and the efficiency ($\mathcal{E}$) of a parallel smoother is

$$\mathcal{S} = p\mathcal{E} = \frac{\hat{T}(1)}{\hat{T}(p)}, \tag{22}$$

where $\hat{T}(1)$ is the execution time for a single processor of the smoother and $\hat{T}(p)$ is the execution time using p processors.

Goal of our analysis is to estimate of the speedup $\bar{\mathcal{S}}, \tilde{\mathcal{S}}$ and the efficiency $\bar{\mathcal{E}}, \tilde{\mathcal{E}}$ of a parallel algorithm using the measures of parallelism of the smoothing procedure (22). Let $l = 0$ denote the finest grid, L^+ be number of grid level consisting of 3^{dL^+} coarsest grids ($d = 2, 3$) and l^* ($0 \le l^* \le L^+$) be a depth of the parallelization. On finer grids l ($0 \le l \le l^*$) parallel smoothing causes little overhead and grid partitioning is used for parallelization. To overcome overhead on coarse grids of level l ($l^* < l \le L^+$), the given discrete problem will be decomposed into 3^{dl^*} subproblems without an overlap using properties of the multigrid structure. It is clear that the number of processor (p) should be $p = 3^{dl^*}$ for perfect load balance [4].

Let us consider parallelization of the first depth, i.e. $l^* = 1$ or $p = 3^d$, $d = 2, 3$. Distribution of the coarse grids among the processors are shown on Fig. 16. We assume that the same number of the smoothing iterations are performed on each grid of the multigrid structure. Since the total number of grid points on all grids of the same level is N, the execution time of the smoothing iterations is constant $T_l(1) = const$ ($l = 0, 1, \ldots, L^+$). For $l^* = 1$ we obtain

$$\bar{\mathcal{S}} = 3^d\bar{\mathcal{E}} < 3^d \frac{1 + q^*L^+}{\dfrac{1}{\mathcal{E}_0} + q^*L^+}. \tag{23}$$

Execution time of the sequential V-cycle can be estimated as

$$T(1) \approx \frac{T_0(1)}{1 - 2^{-d}}. \tag{24}$$

Substitution of (24) into (21) gives the estimations of speedup and efficiency of parallel RMT over V-cycle

$$\tilde{\mathcal{S}} = 3^d\tilde{\mathcal{E}} < \frac{3^d}{1 - 2^{-d}} \frac{1}{\dfrac{1}{\mathcal{E}_0} + q^*L^+}. \tag{25}$$

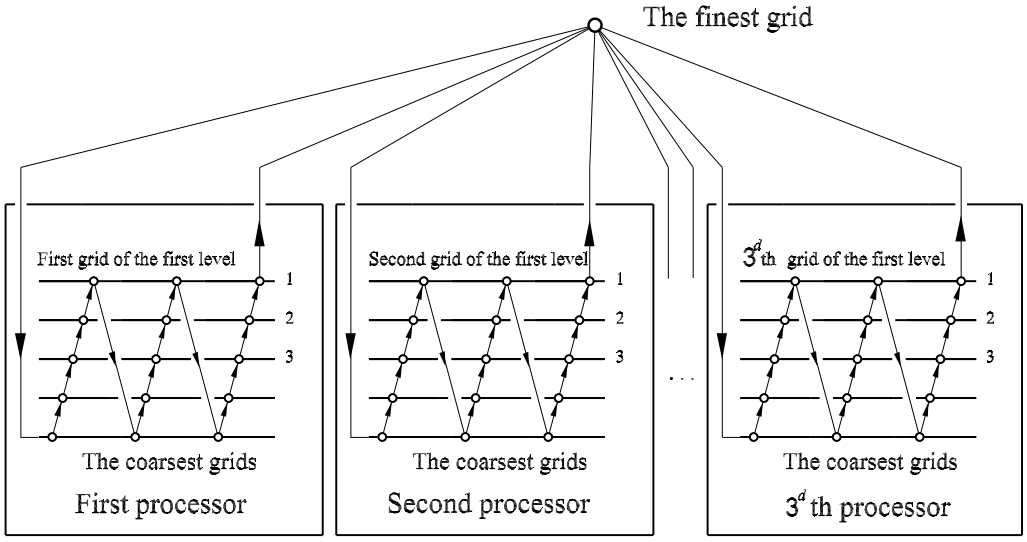

Figure 16: Distribution of subproblems among the processors in case $l^* = 1$ and $q^* = 3$

The estimations (23) and (25) predict that the measures of RMT parallelism depend strongly on efficiency of the parallel smoothing procedure on the finest grid $\mathcal{E}_0$. Since

$$\bar{\mathcal{E}} < \bar{\mathcal{E}}_{\max} = \mathcal{E}_0 \frac{1 + q^* L^+}{1 + q^* L^+ \mathcal{E}_0} \;\Rightarrow\; \frac{\bar{\mathcal{E}}_{\max}}{\mathcal{E}_0} = \frac{1 + q^* L^+}{1 + q^* L^+ \mathcal{E}_0} > 1,$$

the efficiency of parallel RMT will be higher than efficiency of the parallel smoothing procedure on the finest grid.

In the limit case $N \to \infty$ we have:

$$N \to \infty \;\Rightarrow\; L^+ \to \infty \;\Rightarrow\; \begin{cases} \bar{\mathcal{S}} \to 3^d \quad \text{and} \quad \bar{\mathcal{E}} \to 1, \\ \tilde{\mathcal{S}} = O(\lg^{-1} N) \quad \text{and} \quad \bar{\mathcal{E}} = O(\lg^{-1} N), \end{cases}$$

i.e. RMT has full parallelism at $N \to \infty$, but remarkable loss in complexity with classic multigrid is expected.

Eqs. (23) and (25) make it possible to estimate the speedup and efficiency of parallel RMT using only efficiency of parallel smoothing procedure as a function of grid points.

3D Dirichlet boundary value problem for Poisson equation in unit cube is used for numerical experiments. The problem has exact solution e^{x+y+z}, which defines the boundary conditions and source term. Computational grid $245 \times 245 \times 245$ ($N = 14706125 \;\Rightarrow\; L^+ = 4$) is used for the control volume discretization of the problem. OpenMP technology ($l^* = 1$, $d = 3$, $p = 27$) is applied for parallel implementation of RMT. OpenMP is a concurrency platform for multithreaded, shared-memory parallel processing architecture for C, C++ and Fortran. Shared-memory multiprocessor consisting of four processors AMD Opteron 6176 and RAM 128 Gb is used for the computations. Four smoothing iteration steps are performed on each grid with two extra multigrid iterations $q^* = 2$ on coarse levels l ($l^* \le l \le L^+$). The numerical experiments show that obtained efficiencies of parallel smoothing are $\bar{\mathcal{E}}_0 = 0.89$ and $\bar{\mathcal{E}}_* = 0.95$ for the finest and coarse grids, respectively. As a result, obtained efficiency of parallel RMT is 0.92, but estimation (20) predicts $\bar{\mathcal{E}} = \bar{\mathcal{S}} 3^{-d} = 0.98$.

On the other hand, comparison with sequential V-cycle shows that $\tilde{\mathcal{E}} = 0.112$, whereas estimation (21) predicts 0.125. It means that parallel RMT with $p = 27$ only approximately is three time faster than the sequential V-cycle.

Similar estimations for the parallelization of the second depth ($l^* = 2$, $p = 3^{2d}$, $d = 2, 3$) become

$$\bar{\mathcal{S}} = 3^{2d}\bar{\mathcal{E}} < 3^{2d}\frac{2 + q^*(L^+ - 1)}{\dfrac{1}{\mathcal{E}_0} + \dfrac{1}{\mathcal{E}_1} + q^*(L^+ - 1)},$$

$$\tilde{\mathcal{S}} = 3^{2d}\tilde{\mathcal{E}} < \frac{3^{2d}}{1 - 2^{-d}}\frac{1}{\dfrac{1}{\mathcal{E}_0} + \dfrac{1}{\mathcal{E}_1} + q^*(L^+ - 1)}.$$

We summarize the advantages of the multiple coarse grid correction strategy in parallel multigrid:
1) In classical multigrid on the coarse grids algorithms the ratio between communication and computation becomes worse than on fine grids. As a result, a large communication overhead on very coarse grids is expected. This strategy allows us to obtain almost full parallelism on coarse grids by decomposition of the given problem into a fixed number of subproblems without an overlap.
2) The time spent on very coarse grids in W-cycles may become unacceptable. In contrast, the extra multigrid iterations on coarse levels, which are employed in this strategy, lead to further reduction of communication overhead.
3) In classical multigrid algorithms on very coarse grids we may have idle processes. The fixed number of processes in the approach gives almost perfect load balance on the coarse grids.
4) Parallelization of computations on the coarse grids is independent of the smoother.

8 CONCLUSIONS

Application of the essential multigrid principle in single-grid algorithm makes it possible to develop a fast and highly parallel solver of boundary value problems on structured grids. Really RMT can be considered as a problem-independent acceleration technique of Gauss-Seidel iterations. It is expected that RMT will be used in black box software packages where the highest efficiency for a single problem is not so important.

ACKNOWLEDGEMENT

This paper is part of the project “Supercomputer simulation of physical and chemical processes in the high-speed direct-flow propulsion jet engine of the hypersonic aircraft on solid fuels” supported by Russian Science Foundation (project no. 15-11-30012)

REFERENCES

[1] W. Hackbusch, *Multi-grid Methods and Applications*, Berlin, Heidelberg: Springer, 1985.

[2] S.I. Martynenko, Robust multigrid technique for solving partial differential equations on structured grids, *Numerical Methods and Programming*, **1**, 83–100, 2000.

[3] S.I. Martynenko, Robust multigrid technique for black box software, *Computational Methods in Applied Mathematics*, **6**(4), 413–435, 2006.

[4] S.I. Martynenko, Potentialities of the robust multigrid technique, *Computational Methods in Applied Mathematics*, **10**(1), 87–94, 2010.

[5] S.I. Martynenko, *Multigrid Technique: Theory and Applications*, Moscow: FIZMATLIT, 2015. (in Russian)

[6] S.I. Martynenko, Robust multigrid method for solving the Navier-Stokes equations on structured grids, *Modern Applied Science*, **6**(6), 64–72, 2012.

[7] J. Ortega, *Introduction to Parallel and Vector Solution of Linear Systems*, New York: Plenum Press, 1988.

[8] U. Trottenberg, C.W. Oosterlee, A. Schüller, *Multigrid*, London: Academic Press, 2000.

[9] S.P. Vanka, Block-implicit multigrid solution of Navier-Stokes equations in primitive variables, *Journal of Computational Physics*, **65**, 138–158, 1986.