







Parallelization of Robust Multigrid Technique Using OpenMP Technology

Sergey Martynenko^{1,2,3,4} , Weixing Zhou⁵ , İskender Gökalp^{6,7} ,
Vladimir Bakhtin⁸ , and Pavel Toktaliev¹

- ¹ Institute of Problems of Chemical Physics of the Russian Academy of Sciences, Ac. Semenov avenue 1, Chernogolovka, Moscow 142432, Russian Federation
{Martynenko, Toktaliev}@icp.ac.ru
- ² Bauman Moscow State Technical University, ul. Baumanskaya 2-ya, 5/1, Moscow 105005, Russian Federation
- ³ Joint Institute for High Temperatures of the Russian Academy of Sciences, Izhorskaya st. 13 Bd.2, Moscow 125412, Russian Federation
- ⁴ Baranov Central Institute of Aviation Motors, 2, Aviamotornaya Street, Moscow 111116, Russian Federation
- ⁵ Harbin Institute of Technology, 92 West Dazhi Street, Nangang District, Harbin 150001, Heilongjiang Province, China
zhouweixing@hit.edu.cn
- ⁶ Middle East Technical University Ünitersiteler Mahallesi, Dumlupınar Bulvarı No:1, 06800 Çankaya, Ankara, Turkey
igokalp@metu.edu.tr, iskender.gokalp@cnrs-orleans.fr
- ⁷ The Institute of Combustion, Aerothermics, Reactivity and Environment – Centre National de la Recherche Scientifique, 1C, avenue de la Recherche Scientifique, 45071 Cedex 2 Orléans, France
- ⁸ Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences, Miusskaya sq., 4, Moscow 125047, Russian Federation
Bakhtin@keldysh.ru

Abstract. This article represents the parallel multigrid component analysis of Robust Multigrid Technique (RMT). The RMT has been developed for black-box solution of a large class of (non)linear boundary value problems in computational continuum mechanics. Parallel RMT can be constructed by combination of the algebraic and geometric approaches to parallelization. The geometric smoother-independent approach based on a decomposition of the given problem into 3^κ ($\kappa = 1, 2, \dots$) subproblems without an overlap should be used to overcome the problems of large communication overhead and idling processors on coarser levels. The algebraic grid-independent approach based on a decomposition of the given problem into $C3^\kappa$ ($\kappa = 1, 2, \dots$) subproblems with an overlap (multicoloured Vanka-type smoother) should be used for parallel smoothing on finer levels. Standard programming model for shared memory parallel programming OpenMP has been used for parallel implementation of RMT on personal computer and computer cluster.

The activity is a part of the work “Supercomputer modelling of hypervelocity impact on artificial space objects and Earth planet” supported by Russian Science Foundation (project no. 21-72-20023).

This paper represents parallel multigrid cycle, algebraic and geometric approaches to parallelization, estimation of the parallel RMT efficiency and parallel multigrid component analysis.

Keywords: Boundary value problems · Robust multigrid technique · OpenMP · Parallel solvers

1 Introduction

Multigrid algorithms are well known for being the fastest numerical methods for solving elliptic boundary-value problems. There are two trends with respect to the choice of multigrid components [5]:

- 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;
- in robust multigrid algorithms, one tries to choose the components independently of the given problem, uniformly for as large a class of problems as possible.

At the end of the 70ies and at the beginning of the 80ies there was a real boom in research on the multigrid methods. Very interesting multigrid approach had been proposed and developed in Theoretical Division of the Los Alamos National Laboratory. In paper [1], P.O. Frederickson and O.A. McBryan studied efficiency of parallel superconvergent multigrid method (PSMG). The basic idea behind PSMG is the observation that for each fine grid there are two natural coarse grids – the even and odd points of the fine grid. Authors tries to develop optimized multigrid algorithm by combination of these coarse grid solutions for more accurate fine grid correction. The PSMG and related ideas essentially refer to massively parallel computing. To keep all processors of a massively parallel system busy especially on coarse grids, PSMG works simultaneously on many different grids, instead of working only on the standard coarse grid hierarchy [5].

Also in 1990, S.I. Martynenko (Baranov Central Institute of Aviation Motors, Moscow) suggested to use similar multiple coarse grid correction strategy for development of robust multigrid method for black-box software. To avoid terminological confusion, we define software to be black-box if it does not require any additional input from the user apart from the physical problem specification consisting of the domain geometry, boundary and initial conditions, the enumeration of equations to be solved (heat conductivity equation, Navier–Stokes equations, Maxwell equations, etc.) and mediums. The user does not need to know anything about numerical methods, or high-performance and parallel computing [2]. Developed solver is called the Robust Multigrid Technique (RMT), RMT history is given in [3]. For a theoretical description of RMT and corresponding parallel analysis, we refer to [2].

To overcome the problem of robustness, the essential multigrid principle¹ has been used in the single-grid solver. RMT has the least number of problem-dependent components, close-to-optimal algorithmic complexity and high parallel efficiency for large class of boundary value problems. Application field of the RMT is mathematical modelling of complex physical and chemical processes described by the systems of nonlinear strongly coupled partial differential equations. As a result, RMT can use not only a segregated smoothers, but also the coupled Vanka-type smoothers in the multigrid iterations. RMT can be used as efficient solver on structured grids or as a multigrid preconditioner on unstructured grids in black-box software [2]. It should be noted that RMT has close-to-optimal algorithmic complexity: the number of multigrid iterations needed for solving the boundary value problems is independent of the number of unknowns, but computational cost of each multigrid iteration is $O(N \log N)$ arithmetic operations. Loss in computational efforts compared to classic multigrid ($\sim \log N$ arithmetic operations) is a result of true robustness of RMT [2].

In [5], important aspects of parallel classic multigrid are summarized:

- on coarse grids, the ratio between communication and computation becomes worse than on fine grids, up to a (possibly) large communication overhead on very coarse grids;
- on very coarse grids we may have (many) idle processes;
- on coarse grids, the communication is no longer local.

The simplest way to construct a parallel multigrid algorithm is to parallelize all of its components. Although suitable multigrid components may be highly parallel, the overall structure of standard multigrid is intrinsically not fully parallel for two reasons. The first reason is that the grid levels are run through sequentially in standard multigrid. The second reason is that the degree of parallelism of multigrid is different on different grid levels (i.e. small on coarse grids) [5]. In addition, parallelizing the multigrid components will only allow constructing a small-scale granulated algorithm, it means small tasks can be performed in parallel.

The Chan-Tuminaro and the Gannon-van Rosendale algorithms both belong to the class of concurrent methods. The basic approach is to generate independent sub-problems for the different grid levels by projection onto orthogonal sub-spaces. The algorithms differ in the way this decomposition is performed and the way solutions are combined again.

The algorithm of Fredrickson and McBryan follows a completely different approach. Opposed to standard multigrid, the method does not employ a single grid to compute a coarse grid correction, but composes on each level several coarse grid problems. Ideally, the additional information obtained from these multiple coarse grids can be used to improve the convergence rate of the multigrid method, thus improving not only parallel efficiency, but also actual run-time [1]

¹ The essential multigrid principle is to approximate the smooth (low frequency) components of the error on the coarse grids. The nonsmooth (high frequency) components are reduced with a small number (independent of mesh size) of smoothing iterations on the fine grid [6].

Large-scale granularity means large tasks that can be performed independently in parallel. Goal of the paper is to analyse granularity of the basic components of parallel RMT using OpenMP technology. Now the presented approach is suitable for use only with shared memory systems.

2 Parallel RMT

Let N_{G_0} points form the computational grid G_0 . We assume that it is possible to form non-overlapping subgrids $G_i \in G_0$ of N_{G_i} points, such that

$$G_0 = \bigcup_{i=1}^I G_i \quad \text{and} \quad G_n \cap G_m = \emptyset, \quad n \neq m.$$

All subgrids G_i , $i = 1, 2, \dots, I$ form the first grid level and

$$\sum_{i=1}^I N_{G_i} = N_{G_0},$$

but the finest grid G_0 forms zero level. There are a number of the subgrid generation strategy, but we will be interested in an optimal strategy that minimizes the approximation error on the coarse grids G_i .

Let 1D uniform grid for the finite volume discretization is defined as

$$\begin{aligned} x_i^v &= \frac{i-1}{N_x^0} = (i-1)h_x, & i &= 1, 2, \dots, N_x^0 + 1, \\ x_i^f &= \frac{x_i^v + x_{i+1}^v}{2} = \frac{2i-1}{2}h_x, & i &= 1, 2, \dots, N_x^0, \end{aligned}$$

where N_x^0 is a discretization parameter and $h_x = 1/N_x^0$ is a mesh size. Points x_i^v and x_i^f can be vertices or faces in the finite volume discretization of mathematical model equations. Figure 1 represents a triple coarsening used in RMT. In this case, each finite volume on the coarser grids is union of three finite volumes on the finer grid. It is easy to see that the smoothing steps on different grid of the same level are independent of each other and can be performed in parallel.

This grid hierarchy will be called a multigrid structure generated by the finest grid G_0 . Note that each grid of the multigrid structure can generate own multigrid structure. The triple coarsening and the finite volume discretization lead to the problem-independent restriction and prolongation operators [2].

Multigrid cycle of RMT is smoothing on the multigrid structure as shown on Fig. 2. The multigrid schedule of RMT is sawtooth cycle, i.e. a special case of the V-cycle, in which smoothing before the coarse grid correction (pre-smoothing) is deleted [6].

OpenMP is an implementation of multithreading, a method of parallelizing whereby a primary thread (a series of instructions executed consecutively) forks a specified number of sub-threads and the system divides a task among them

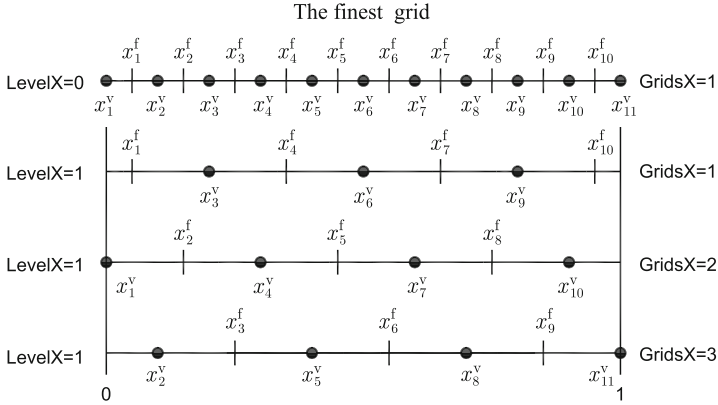


Fig. 1. Triple coarsening in RMT: the finest grid and three coarse grids of the first level.

[8]. The threads then run concurrently, with the runtime environment allocating threads to different processors. OpenMP uses a portable, scalable model that gives programmers a simple and flexible interface for developing parallel RMT for platforms ranging from the standard desktop computer to the supercomputer.

From the parallel point of view, RMT has the following attractive features:

1. All coarse grids of the same level have no common points and the smoothing iterations can be performed in parallel.
2. The number of grids (known in advance) on each level predicts the number of threads for parallel RMT.
3. Almost the same number of points on each grid of the same level leads to almost uniform load balance.
4. The most powerful coarse grid correction strategy used in RMT makes it possible to use the weak smoothers in the parallel multigrid iterations.

Each grid level of the multigrid structure consists of 3^{dl} grids (the problem dimension $d = 1, 2, 3$; $l = 0, 1, \dots, L^+$), where L^+ is the coarsest level. Therefore we should use $p = 3^\kappa$ ($\kappa = 1, \dots, L^+$) threads for uniform load balance. Case $\kappa = 0$ corresponds to sequential implementation.

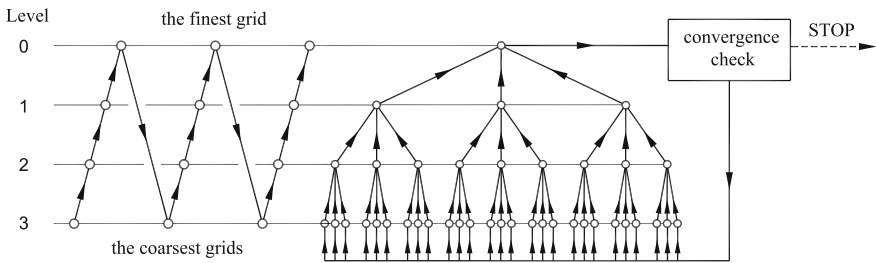


Fig. 2. Sequential multigrid cycle of RMT for solving 1D problems.

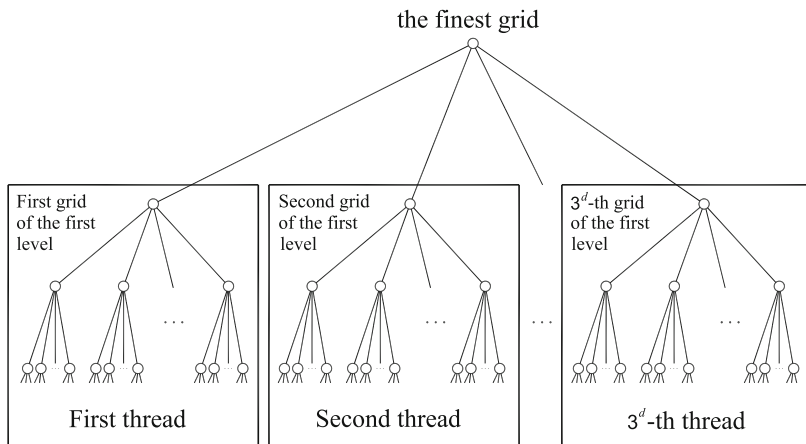


Fig. 3. Distribution of the coarse grids of the first level to $p = 3^d$ threads.

If we use $p = 3^\kappa$ ($\kappa = 1, \dots, L^+$) threads for parallel RMT, we have to distinguish two cases:

- *algebraic parallelism* $\kappa > dl$ (finer levels). In the first case, multicoloured unknown ordering could be viewed as a specific way to parallelize a smoothing iterations independently of the computational grid [4];
- *geometric parallelism* $\kappa \leq dl$ (coarse levels). In the second case, RMT has almost full parallelism independently of the smoothing procedure. Distribution of the coarse grids of the first level to $p = 3^d$ (or $\kappa = d = 2, 3$) threads is shown on Fig. 3.

Remember the common measure of parallelism [4]:

Definition 1. *The efficiency E of a parallel algorithm is*

$$E = \frac{1}{p} \frac{T(1)}{T(p)},$$

where $T(1)$ is an execution time for a single thread and $T(p)$ is an execution time using p threads.

Now we analyse parallel multigrid cycles. Since RMT has almost full parallelism on the coarse levels ($\kappa \geq dl$), it is possible to perform extra smoothing iterations on these levels. Figure 4 demonstrates a parallel multigrid cycle with two extra multigrid iterations on the coarse levels ($q^* = 3$). It is clear that parallel efficiency of the smoothing iterations of the finest grid will be critical for the parallel RMT efficiency.

In addition to abovementioned common measure of parallelism, we introduce measure of parallel properties of the smoothers:

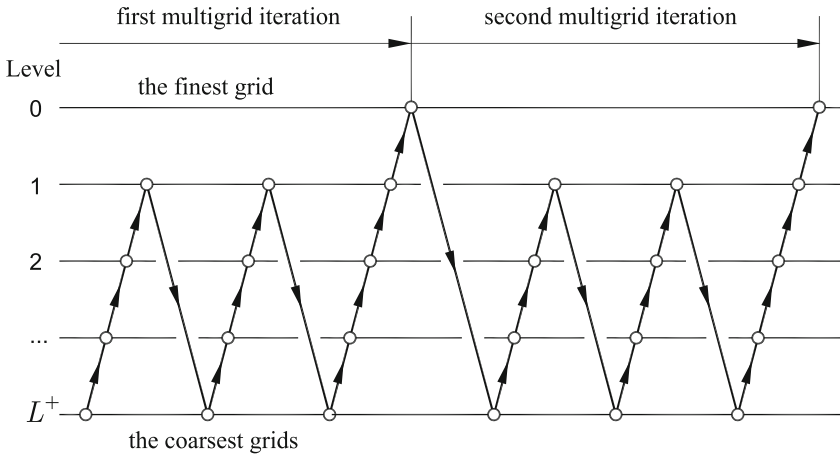


Fig. 4. Parallel multigrid cycle ($q^* = 3$).

Definition 2. The efficiency E_l of a parallel smoother is

$$E_l = \frac{1 T_l(1)}{p T_l(p)}, \tag{1}$$

where $T_l(1)$ is an execution time for a single thread of the smoother, $T_l(p)$ is an execution time using p processors and $l = 0, 1, \dots, L^+$ is serial number of the grid levels.

Since all grids of the same level have the same number of points, it is expected the execution time for the smoothing iterations is l -independent: $T_l(1) = const$. For example, the execution time of the sequential multigrid iteration of RMT becomes

$$T(1) = T_0(1) + q^* \sum_{l=1}^{L^+} T_l(1) = (1 + q^* L^+) T_0(1),$$

where q^* is the number of the multigrid iterations on coarse levels. Efficiency of the parallel multigrid cycle shown of Fig. 4 can be estimated as

$$E \approx \frac{q^* L^+ + 1}{q^* L^+ + \frac{1}{E_0}}.$$

This estimation predicts that $E > E_0$.

For this cycle, solution u of a boundary value problem should be represented as

$$u = c + c_d + \hat{u},$$

where c is the coarse grid correction (defined on the finest grid), c_d is the coarse grid correction (defined on the multigrid structures generated by the coarse

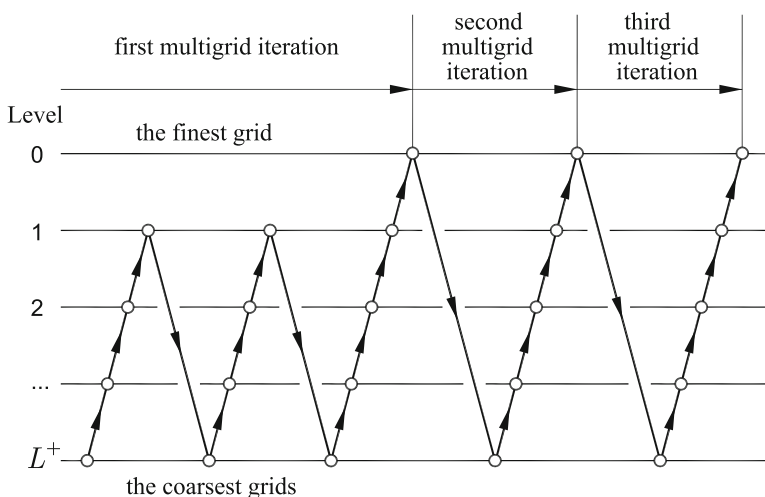


Fig. 5. Simplified parallel multigrid cycle ($q^* = 2$).

grids of the first level) and \hat{u} is approximation to the solution u . This approach is difficult to use for nonlinear boundary value problems.

Simplified parallel multigrid cycle is shown on Fig. 5. This cycle makes it possible to apply standard Σ -modification of the solution used in RMT [2]. In this case, efficiency of parallel RMT depends on the number of the multigrid iterations q :

$$E(+\infty) = \frac{L^+ + 1}{L^+ + \frac{1}{E_0}} < E(q) < \frac{q^*L^+ + 1}{q^*L^+ + \frac{1}{E_0}} = E(1).$$

To illustrate the simplified parallel multigrid cycle, we consider the model Dirichlet boundary value problem for the Poisson equation

$$u''_{xx} + u''_{yy} + u''_{zz} = -f(x, y, z) \tag{2}$$

in domain $\Omega = (0, 1)^3$. Substitution of the exact solution

$$u_a(x, y, z) = \exp(x + y + z), \tag{3}$$

into (2) gives the right-hand side function

$$f(x, y, z) = -3 \exp(x + y + z)$$

and the boundary conditions.

Let us define error of the numerical solution as

$$\|e\|_\infty = \max_{ijk} |u_a(x_i, y_j, z_k) - \hat{u}^h_{ijk}|, \tag{4}$$

where \hat{u}^h_{ijk} is approximation to the solution u .

As smoother, we use $3 \times 3 \times 3$ block Gauss–Seidel iterations (Vanka-type smoother [7]). This means that all 27 unknowns are updated collectively. As a rule, the Vanka-type smoother is used for solving systems of PDEs including saddle points problems. Of course, the discrete Poisson problem on an uniform grid does not require application of the Vanka-type smoother, since for the algebraic system that results from the seven-point discretization a point smoother is efficient, but this algorithm can be used for algorithmic complexity estimation in simulation of the coupled physicochemical phenomena.

Figure 6 represents reduction of the error (4) in the first multigrid iteration of the simplified parallel cycle (Fig. 5) starting the iterand zero. After four multigrid iterations steps on the multigrid structures generated by the coarse grids of the first level, error of approximation to the solution of (2) composed from solutions of 27 independent problems becomes small.

This error can be estimated as follows: for second-order discretization of (2), we have

$$\|u_a - u^h\| = Ch^2.$$

Second-order seven-point discretization of (2) on coarse grids of the first level with mesh size $3h$ results in

$$\|u_a - u^{3h}\| = C(3h)^2 = 9Ch^2.$$

Error estimation becomes

$$\|u^h - u^{3h}\| \leq \|u_a - u^h\| + \|u_a - u^{3h}\| = Ch^2 + 9Ch^2 = 10Ch^2.$$

i.e. difference between numerical solution and approximation to the solution is one significant digit.

Compared to the traditional single-grid Seidel method, RMT has a single extra problem-dependent component – the number of smoothing iterations. Numerical experiments are intended for studying efficiency of the parallel components of RMT as a function of the unknowns number (or the number N of grid points used for the boundary problem approximation).

3 Algebraic Parallelism of RMT

Efficiency of Gauss–Seidel method depends strongly on the ordering of equations and unknowns in many applications. Also, the possibility of parallel computing depends strongly on this ordering. The equations and unknowns are associated in a natural way with blocks of the unknowns. It suffices, therefore, to discuss orderings of the unknown blocks. Figure 7 represents three coloured block ordering in one dimension. Multicoloring allows the parallel execution of Gauss–Seidel relaxation. In d dimensions, such multicoloured block ordering defines the parallel block plane smoothing.

If the number of threads is less than the number of grids forming the given level, unknown blocks partitioning is a natural approach to algebraic parallelism

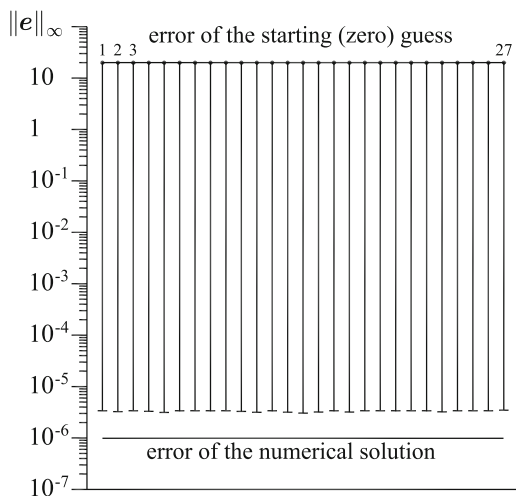


Fig. 6. Reduction of the error (4) in 27 independent subproblems (the first multigrid iteration).

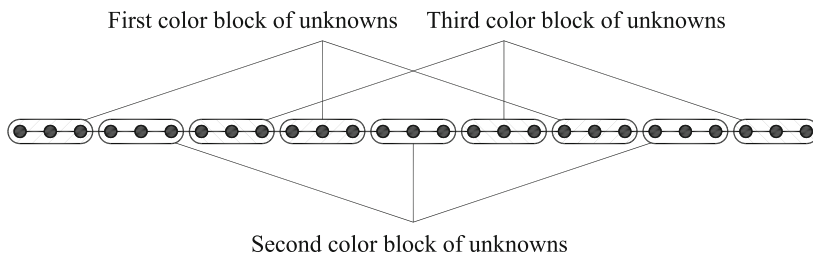


Fig. 7. 1D three coloured ordering of the unknown blocks.

of RMT. In this approach, the set of unknown blocks is split into $C3^\kappa$ ($\kappa = 1, 2, \dots$) subsets, such that 3^κ available threads can jointly solve the underlying discrete problem. Here C is the number of colours in the used multicoloured ordering of the unknown blocks.

Personal computers (Intel(R) Core(TM) i7-4790 CPU@3.60 GHz) and computer cluster K-60 of Keldysh Institute of Applied Mathematics [9] (Russian Academy of Sciences) are used for the computational experiments for study of the parallel smoothing efficiency E_0 (1) on the finest grid.

Figure 8 represents results of the parallel computations. Reduction of the parallel RMT efficiency is observed in 27 thread implementation. Smoothing iterations on the finest grid is small-scale granulated component of RMT, but large algorithmic complexity of the Vanka-type smoother ($\sim N^3$ arithmetic operations, for each box, one has to solve a $N \times N$ system of equations to obtain corrections for the unknowns) leads to almost full parallelism.

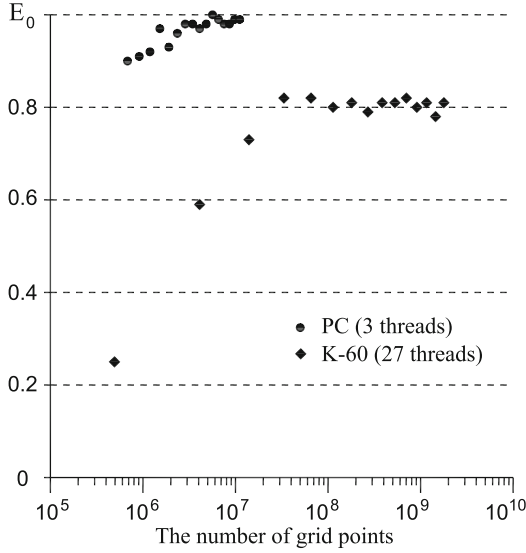


Fig. 8. Parallel smoothing efficiency E_0 (1) on the finest grid.

4 Geometric Parallelism of RMT

Remember that the problem of the very coarse grids leads to multigrid specific parallel complications which do not occur in classical single-grid algorithms. This crucial impact of the coarse grids increases, the more often the coarse grids are processed in each cycle. A parallel W-cycle, for example, has a substantially different parallel complexity from that of a parallel V-cycle [5].

An important advantage of the geometric approach to parallelization is the reduction of the discrete boundary value problem to the set of 3^k independent problems, which can be solved in parallel without data exchange between processors for any used solver. Therefore one aim of parallel RMT is to perform as little work as possible on the finest grid and to do as much work as possible on the coarse levels. Extra multigrid iterations on the coarse levels lead to a considerably better parallel efficiency of RMT. Smoothing iterations on the coarse levels is large-scale granulated component of RMT.

Figure 9 represents results of the parallel solution of the model boundary value problem (2) on the coarse levels. We perform four multigrid iterations on the multigrid structures generated by the coarse grids of the first level. Also reduction of the parallel RMT efficiency is observed in 27 thread implementation. This test demonstrates a significant loss of efficiency for solving 27 independent subproblems. It means that the memory access pattern for computing those multigrid iterations need to be carefully designed, as their performance is very dependent on the memory bandwidth. Our study illustrates that memory bandwidth is a major bottleneck for multigrid. The role of memory and the deepening

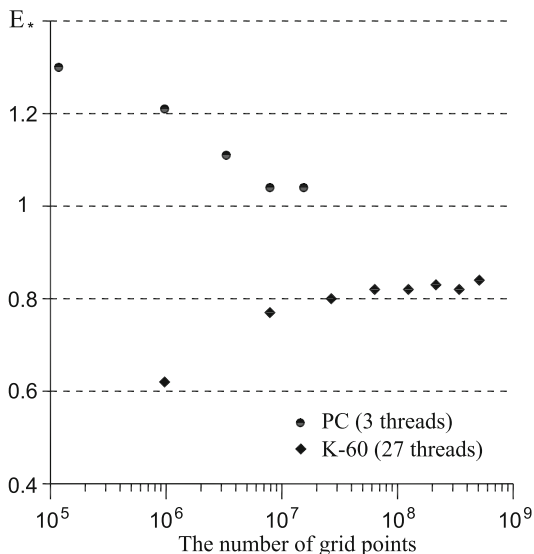


Fig. 9. Parallel smoothing efficiency on the coarse levels.

memory hierarchy on contemporary processors in the performance in numerical codes cannot be overstated.

Parallel RMT allows one to avoid a load imbalance and communication overhead on very coarse grids.

5 Parallel Multigrid Iteration

First parallel multigrid iteration of RMT shown on Fig. 5 consists of four multigrid iterations on the independent multigrid structures generated by the coarse grids of the first level (geometric parallelism of RMT) and parallel smoothing on the finest grid based on the multicoloured block Gauss–Seidel iterations (algebraic parallelism of RMT). Figure 10 represents efficiency of the first parallel multigrid iteration.

6 Remarks on Parallel Implementation

Inefficient memory access is one of the most common performance problems in parallel programs. The speed of loading data from memory traditionally lags behind the speed of their processor processing. The trend of placing more and more cores on a chip means that each core has a relatively narrower channel for accessing shared memory resources. On NUMA computers, accessing remote RAM is slower than accessing local memory. Therefore, to access the RAM of another socket it is necessary to access the QPI or HyperTransport bus, which is slower than the local RAM access bus. The program analysis performed by the

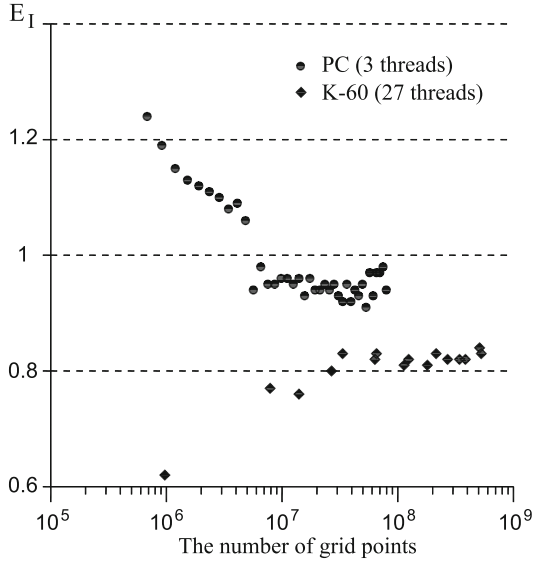


Fig. 10. Efficiency of the first parallel multigrid iteration.

Intel VTune Performance Analyzer shows that when 27 thread implementation using 2 Intel Xeon Gold 6142 v4 processors leads to 15–18% of memory accesses are accesses to remote memory. It results in reduction of the parallel program efficiency. This problem does not arise if all threads go to one processor.

Our future work is development of approaches to made memory-bandwidth efficient for parallel RMT.

7 Conclusions

The RMT has been developed for application in black-box software, because it has the least number of the problem-dependent components. This technique can solve many (non)linear problems to within truncation error at a cost of $CN \log N$ arithmetic operations, where N is the number of unknowns and C is a constant that depends on the problem. Large-scale granularity of the parallel RMT (geometric parallelism) coupled with the multicoloured Gauss-Seidel iterations (algebraic parallelism) lead to almost full parallelism of the multigrid iterations. The geometric approach, based on a decomposition of the given problem into a number of subproblems without an overlap, should be used to overcome the problems of large communication overhead and idling processors on the very coarse grids independent of the smoother. Results of numerical experiments on shared memory architectures show the high parallel efficiency of RMT components.

References

1. Frederickson, P.O., McBryan, O.A.: Parallel superconvergent multigrid. Multigrid Methods. In: McCormick, S. (ed.) Theory, Applications and Supercomputing, pp. 195–210. Marcel Dekker, New York (1988)
2. Martynenko, S.I.: The Robust Multigrid Technique: For Black-Box Software. De Gruyter, Berlin (2017)
3. Martynenko, S.I.: Sequential software for robust multigrid technique. Triumph, Moscow (2020). https://github.com/simartynenko/Robust_Multigrid_Technique_2020
4. Ortega, J.M.: Introduction to Parallel and Vector Solution of Linear Systems. Plenum, New York (1988)
5. Trottenberg, U., Oosterlee, C.W., Schüller, A.: Multigrid. Academic Press, London (2001)
6. Wesseling, P.: An Introduction to Multigrid Methods. Wiley, Chichester (1992)
7. Vanka, S.P.: Block-implicit multigrid solution of Navier-Stokes equations in primitive variables. J. Comput. Phys. **65**(1), 138–158 (1986)
8. OpenMP Homepage. www.openmp.org
9. KIAM Homepage. www.kiam.ru/MVS/resourses/k60.html