Further Development of the Parallel Program Complex of SL-AV Atmosphere Model

Mikhail Tolstykh ✉[1,2,3], Rostislav Fadeev[1,3], Gordey Goyman [3,1],

Vladimir Shashkin [1,2,3]

[1] Institute of Numerical Mathematics Russian Academy of Sciences, Moscow, Russia
[2] Hydrometcentre of Russia, Moscow, Russia
[3] Moscow Institute of Physics and Technology, Dolgorpudny, Russia
mtolstykh@mail.ru,
rost.fadeev@gmail.com,
gordeygoyman@gmail.com,
vvshashkin@gmail.com

Abstract. The SL-AV global semi-Lagrangian atmosphere model is applied to the operational medium-range weather forecast at Hydrometeorological center of Russia. The works on increasing the code scalability and using future computer architectures are described. The scalable parallel multigrid algorithm for solving the linear algebraic equations systems is implemented. It is expected that the multigrid algorithm will be used instead of direct algorithm based on fast Fourier transforms requiring global communications. The results for convergence and strong scalability of the multigrid method are given.

The parallel scalability of the low-resolution versions of the SL-AV model for both seasonal and climate simulation has been evaluated at computer systems based on Intel Xeon Phi 2 (Knights Landing) processors. The results show a practical possibility to use these processors for the global atmosphere modelling with the efficiency comparable to the classical cluster systems.

Keywords: Global atmosphere model · Numerical weather prediction · Climate change modeling · Scalable algorithms for solving elliptic equations · Massively-parallel implementation of the atmosphere model

1 Introduction

Numerical weather prediction (NWP) atmosphere models require huge computer resources. The modern global NWP model has the resolution of about 10 km and about 100 vertical levels, thus the problem dimension is approaching 10^9. Operational application of an NWP model requires the 24-hour forecast be computed in 5-20 minutes. This means that atmosphere model should use efficiently up to 10^5 processor cores.

SL-AV is the global atmosphere model developed at the Institute of Numerical Mathematics, Russian Academy of Sciences (INM RAS) in cooperation with the Hydrometeorological centre of Russia (HMCR) [1]. SL-AV is the model acronym (semi-Lagrangian, based on Absolute-Vorticity equation). The SL-AV model is applied to the operational medium-range weather forecast at Hydrometeorological center of Russia. It is also used as a component of the long-range probabilistic forecast system. The dynamical core of this model applies the semi-implicit semi-Lagrangian approach allowing time steps several times larger than in Eulerian methods [2]. The most part of subgrid-scale parameterizations is adopted from ALADIN/LACE model [3, 4].

SL-AV model uses a combination of MPI and OpenMP technologies. Description of the dynamical core and parallel implementation is given in [2]. Briefly, each MPI process performs computations in the band of grid latitudes during the first phase of the time-step. OpenMP threads are used to parallelize loops along longitude dividing the latitude belt into a number of parts. The second phase of SL-AV time-step consists in solving linear systems of equations using direct solvers in the space of Fourier coefficients obtained after Fast Fourier Transforms in longitude. To apply these direct solvers, the set of Fourier coefficients from all grid latitudes are gathered in the memory of specific MPI-process using data transposition. Each MPI-process performs computations for set of longitude Fourier coefficients from pole to pole. OpenMP parallelization of loops in vertical is applied.

Currently, the SL-AV code runs at 3024 cores with 70% efficiency, at 4536 cores with 63% efficiency, and at 9072 cores with 45% efficiency (while comparing with 512-cores run). This is achieved for the grid of 3024 by 1513 points in longitude and latitude respectively. This grid corresponds to 13 km resolution at the equator and has 51 levels in vertical. The data transpositions before and after the solution of elliptic problems require global communications between the processors. This will become a problem on future massively parallel computers. Therefore, the work has started to implement scalable iterative grid-point solvers. It is known that iterative solvers for elliptic problems can scale up to tens of thousands processors [5]. The second part of the SL-AV dynamical core is the semi-Lagrangian advection that is also known to scale up to 10^4 processors [6]. The replacement of the direct solver for elliptic equations on the sphere based on Fast Fourier Transforms with the multigrid solver is presented in Section 2 of this paper.

Recently, the application of relatively cheap massively parallel accelerators such as GPU or Intel Xeon Phi in different areas of mathematical modelling gained increased popularity in the world, especially in molecular dynamics, chemistry, electrodynamics, astrophysics etc. This was due to growing demand for computer power, from one side, and known limitations of growth for traditional cluster systems, from the other side. However, the applications of such systems in atmosphere modelling so far have been limited. The part of the problem is that the most part of the atmosphere models is written in different dialects of Fortran language and have very complex code (typically, hundreds of thousand lines) so they are not suited for GPUs. Contrary to computer systems with GPU that do not reasonably support Fortran codes, Intel Xeon Phi systems allow using Fortran. The previous generation of Intel Xeon Phi (codename Knights Corner) was designed as a coprocessor so had certain limitations, for example, on memory exchange between host and coprocessor. The recent introduction of cluster systems based on standalone many-core Intel Xeon Phi 2 processors opened a possibility to implement the existing parallel atmosphere model Fortran code directly, without any change. So the second problem considered in this paper is a study on possibility to run the existing SL-AV code at the Intel Xeon Phi processor systems. The first results available today are presented in Section 3.

2 Implementation of the Multigrid Solver in SL-AV Model

2.1 Problem Statement

The SLAV model uses semi-implicit semi-Lagrangian time integration scheme [7] applied to dynamics equations formulated in terms of (vertical component of the) vorticity- (horizontal) divergence [2]. After discretization in space, this approach leads to the set of 2D elliptic equations for each vertical level to be solved at each time step. First, Helmholtz equations are solved to obtain divergence field at the new time level:

$$(K^2 - \nabla^2)S = H \tag{1}$$

K is the constant depending on vertical level, ∇^2 is the horizontal Laplace operator on the sphere, S is a vector variable related to the divergence at the new time level by the linear equation $D = VS$, V being known matrix. Then the relative vorticity ω at the new time level is calculated and Poisson equations are solved to obtain stream-function and velocity potential ψ, χ.

$$\nabla^2 \chi = D\,, \tag{2}$$

$$\nabla^2 \psi = \omega\,. \tag{3}$$

The horizontal wind velocities are then restored using relations

$$u = -\frac{1}{a}\frac{\partial \psi}{\partial \varphi} + \frac{1}{a\cos\varphi}\frac{\partial \chi}{\partial \lambda}, \tag{4}$$

$$v = \frac{1}{a\cos\varphi}\frac{\partial \psi}{\partial \lambda} + \frac{1}{a}\frac{\partial \chi}{\partial \varphi}. \tag{5}$$

Currently, the abovementioned equations are solved in the space of longitudinal Fourier components in the SL-AV model. This means that all the derivatives in longitude are replaced by multiplication by the corresponding coefficients. Compact fourth-order differences are used to approximate the derivatives in latitude. Thus, a 2D equation on the sphere is replaced by the set of 1D linear systems of equations solved by block-tridiagonal version of Thomas algorithm. The parallel implementation of this approach requires data transpositions (hence global communications between MPI processes) before and after these solvers and is a principal obstacle in implementing 2D MPI domain decomposition in this part of the model. The 1D domain decomposition currently used in the SL-AV model code limits the number of the MPI processes by the number of grid points in latitude.

We present here the results of implementing 2D finite-difference approximations for Helmholtz and Poisson equations described above. The arising linear systems of equations are solved with previously implemented multigrid parallel algorithm [8].

2.2 Discretized Equations and the Algorithm

Geometric multigrid with V-cycle [9] is chosen as a base algorithm for the new solver. Intergrid operators are bilinear interpolation and 8-point full weighting. Gauss-Seidel method with red-black ordering is used as a smoother. At the bottom level of the V-cycle, the matrix is inverted with BICGstab solver [10]. We use conditional semi-coarsening approach [11] to account for the anisotropy of the regular latitude-longitude grid near the poles.

Compact finite differences used in solvers in the current version of SL-AV code are not well suited for parallel solver because they imply global dependence of the derivative on values of the function. Thus they are replaced with the local second-order approximation in the solvers. When applying conditional semicoarsening procedure [11], the resolution in latitude becomes irregular. Let us define grid latitudes $\varphi_j, j \in [0, N_\varphi]$ with arbitrary spacing and a constant mesh size in longitude $h_\lambda = \frac{2\pi}{N_\lambda}$ where N_λ , N_φ are grid dimensions in longitude and latitude respectively. Let us denote

$$V_j = a^2\left(\sin\frac{\varphi_{j+1}+\varphi_j}{2} - \sin\frac{\varphi_j+\varphi_{j-1}}{2}\right)h_\lambda, dS_{\varphi,j} = \frac{\varphi_{j+1}-\varphi_{j-1}}{2}, dS_{\lambda,j\pm\frac{1}{2}} =$$

$$= h_\lambda \cos\frac{\varphi_{j\pm1}+\varphi_j}{2}. \qquad (6)$$

The second-order finite-volume approximation from [12] is used to discretize Laplace operator. Except for the pole points, it is written as

$$(\nabla^2\psi)_{i,j} = \frac{1}{V_j}\left(dS_{\varphi,j}\frac{\psi_{i+1,j}-2\psi_{i,j}+\psi_{i-1,j}}{h_\lambda\cos\varphi_j} + \left(\frac{\psi_{i,j+1}-\psi_{i,j}}{h_\varphi}dS_{\lambda,j+1/2} - \frac{\psi_{i,j}-\psi_{i,j-1}}{h_\varphi}dS_{\lambda,j-1/2}\right)\right). \qquad (7)$$

For the pole grid points, the following formulae are used

$$(\nabla^2\psi)_S = \frac{1}{a^2\left(1+\sin\frac{\varphi_1+\varphi_0}{2}\right)h_\lambda}\sum_i \frac{\psi_{i,1}-\psi_S}{h_\varphi} dS_{\lambda,\frac{1}{2}} \qquad (8)$$

$$(\nabla^2\psi)_N = -\frac{1}{a^2\left(1-\sin\frac{\varphi_{N_\varphi}+\varphi_{N_\varphi-1}}{2}\right)h_\lambda}\sum_i \frac{\psi_N-\psi_{i,N_\varphi-1}}{h_\varphi} dS_{\lambda,N_\varphi-1/2} \qquad (9)$$

In order to reconstruct the horizontal velocity field, the standard fourth-order finite difference formulae are used to approximate derivatives in longitude and in latitude, except for latitudinal derivatives near the poles where third-order formulae are used.

2.3 Convergence and Scalability

Convergence of the implemented multigrid solver for equations described in Section 2.1 is studied for different grid resolutions between 128x64 and 2048x1024, and different number of iterations for smoothing operator. There is practically no convergence dependence on the problem size when using two iterations of pre- and post-smoothing.

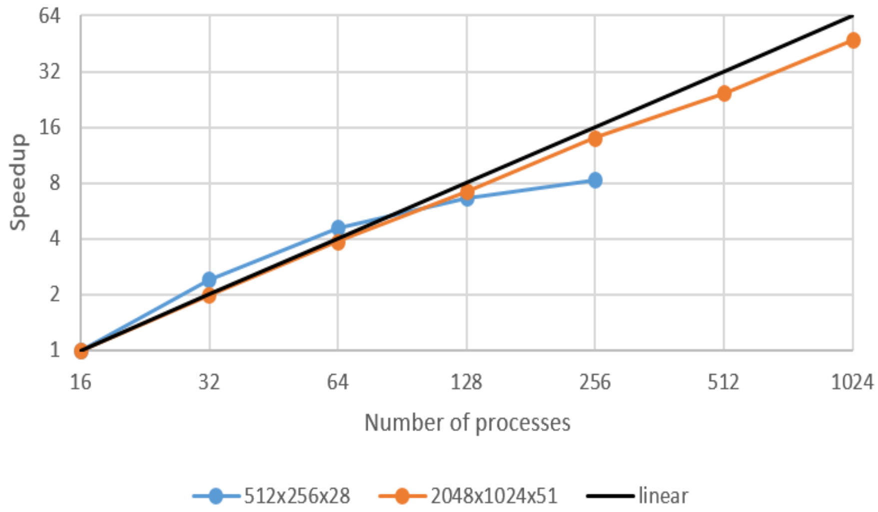

Fig. 1 Strong scalability of the multigrid solver.

The algorithm scalability is studied at MVS10P cluster system installed at Joint Supercomputer Center (Moscow, Russia). This system is based on two-processor nodes with Intel Xeon E5-2690 8-core processors. The strong scalability for problems with grid sizes 512x256x28 and 2048x1024x51 is studied. These grids approximately correspond for different version of the SL-AV model. Parallel speedup with respect to 16 processor cores is presented in Fig.1. One can see that the problem with the size of 512x256x28 scales up to 256 cores with the efficiency more than 50 %. The problem with the size of 2048x1024x51 scales efficiently up to at least 1024 processor cores.

2.4 Implementation in the SL-AV Model

The discretization and the algorithm described above are implemented in the SL-AV model. To test the accuracy of the new algorithm, a series of 31 numerical weather 72-hour forecasts starting with initial data of each day of January 2014 at 12 hours UTC is calculated. The model version with the resolution 0.9 degrees in longitude, 0.72 degrees in latitude and 28 vertical levels is used. The grid dimension is 400x251x28. The averaged over series root mean squared errors for forecast of geo-potential heights at 850, 500 and 250 hPa are depicted in Fig.2 for forecast lead times of 24, 48 and 72 hours. The errors are averaged over Northern extratropics (20-90N). One can see that the new solver presented above slightly reduces forecast errors of the SL-AV model as compared with the 'standard' direct solver. Similar results are obtained for other regions. Also, the tests have revealed that it is sufficient to reduce the norm of the residual by 10^4 times that corresponds to 1 to 3 V-cycle iterations.

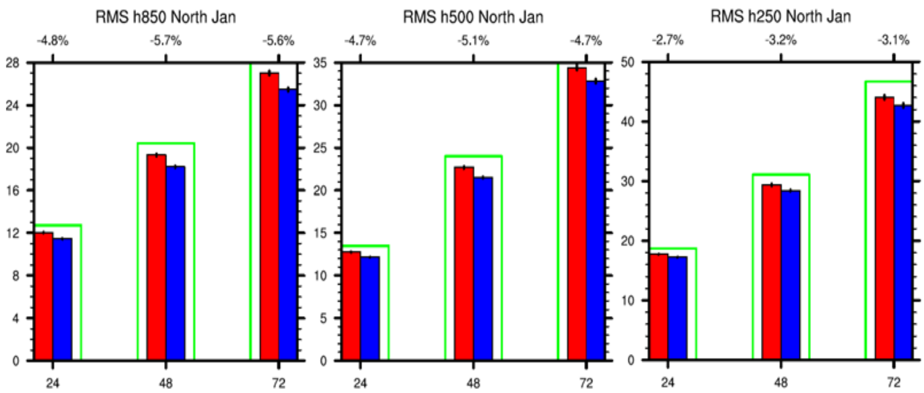

Fig. 2. Root-mean squared geopotential errors averaged over 31 forecasts and Nortern extra-tropics for the standard SL-AV version (red) and the SL-AV model with multigrid solvers (blue). Green boxes mean that the results are statistically significant.

3 Testing SL-AV Model Code at Intel Xeon Phi 2 Systems

3.1 Model Configurations and System Description

Based on growing requirements for computer resources related to the ongoing development of the climate version of the SL-AV model, we have studied possibility to run the existing SL-AV code at the Intel Xeon Phi many-core processor systems. It is essential that no changes were made to the parallel program complex working at `traditional' x86-based clusters. The two model versions having different resolution are tested. The first version has the horizontal resolution 0.9 degrees in longitude, 0.72 degrees in latitude and 28 vertical levels. The horizontal resolution of the second version is 0.56 degrees in longitude, 0.45 degrees in latitude, 50 vertical levels. The problem dimensions are thus 400x251x28 and 640x401x50 respectively.

The cluster system based on Intel Xeon Phi 2 processors is used. Each node contains processor 7250 (codename Knights Landing or KNL) with 68 cores allowing up to 272 hyperthreads. There are 16 Gbytes of fast MCDRAM memory and 48 Gbytes of DDR4 memory, Intel Omnipath interconnect allowing up to 100 Gbytes per second transfer rate. The peak node performance is about 3.04 Tflops. At the time of writing this paper, only three-node cluster was available for tests. Currently, more nodes with such processors are being installed at Joint Supercomputer Center RAS in Moscow, so such tests allow preparing for a proper use of these resources.

3.2 Results and Conclusions

The elapsed times for the SL-AV model time step as a function of hyperthreads number are depicted in Fig.3 for 400x251x28 version and in Fig.4 for 640x401x50 version. The model time step here comprises one time step calling solar and longwave radiation computations and three time steps without radiation computations. The number of hyperthreads is the product of the MPI processes number by OpenMP threads number. Red dots correspond to one 68-core processor, blue dots correspond to two processors, and green ones correspond to the use of three processors. The comparison with the results obtained with RSC Tornado Intel Xeon E2690 cluster installed at Roshydromet's Main Computing Center is presented at the same plots in black lines. Vast variety of dots illustrates a strong dependence of the elapsed time on chosen combination of MPI processes and OpenMP threads. The optimum configurations are marked with solid lines. The black dashed line corresponds to linear scalability. The following conclusions can be made upon inspecting Figs. 3 and 4.

The SL-AV model code for tested versions scales at KNL processor systems similar to classical x86-based systems if an optimum combination of MPI processes and OpenMP threads is used. The absence of significant jumps in the scalability curve when increasing the number of KNL processors shows sufficient interconnect exchange rate.

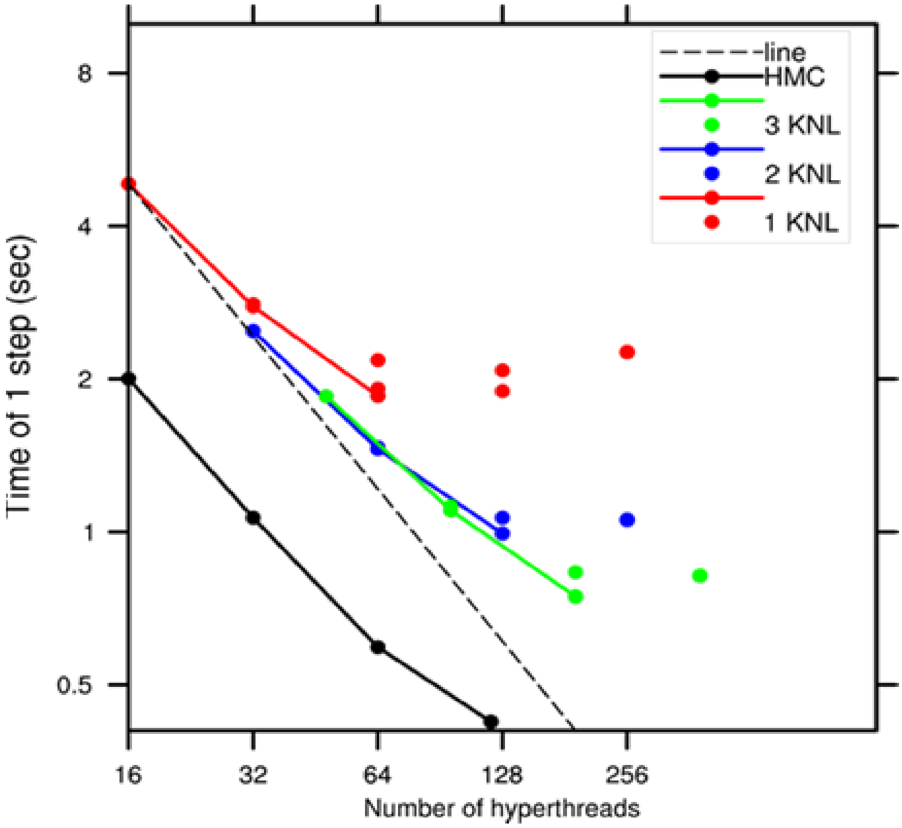

Fig.3. Scalability of SL-AV model code with 400x251x28 grid for different combinations of MPI processes and OpenMP threads. See text for details.

The efficient use KNL requires good code vectorization. One can see from Figs. 3 and 4 that increasing the number of hyperthreads at single KNL processor slows down the execution of 28-level version while 50-level version continues to accelerate. The last result can be explained by the specifics of SL-AV parallel code. Indeed, 1D MPI decomposition in latitude is supplemented with OpenMP parallelization along longitude. The innermost vectorizable loops in many computationally demanding model blocks are in longitude or in the vertical and increasing both dimensions improves vectorization.

The results demonstrate the possibility to use Intel Xeon Phi 2 processors and cluster systems based on these processors for SL-AV model computations. Important points needed to achieve good performance are the choice of proper combination of numbers of MPI processes and OpenMP threads and good code vectorization.

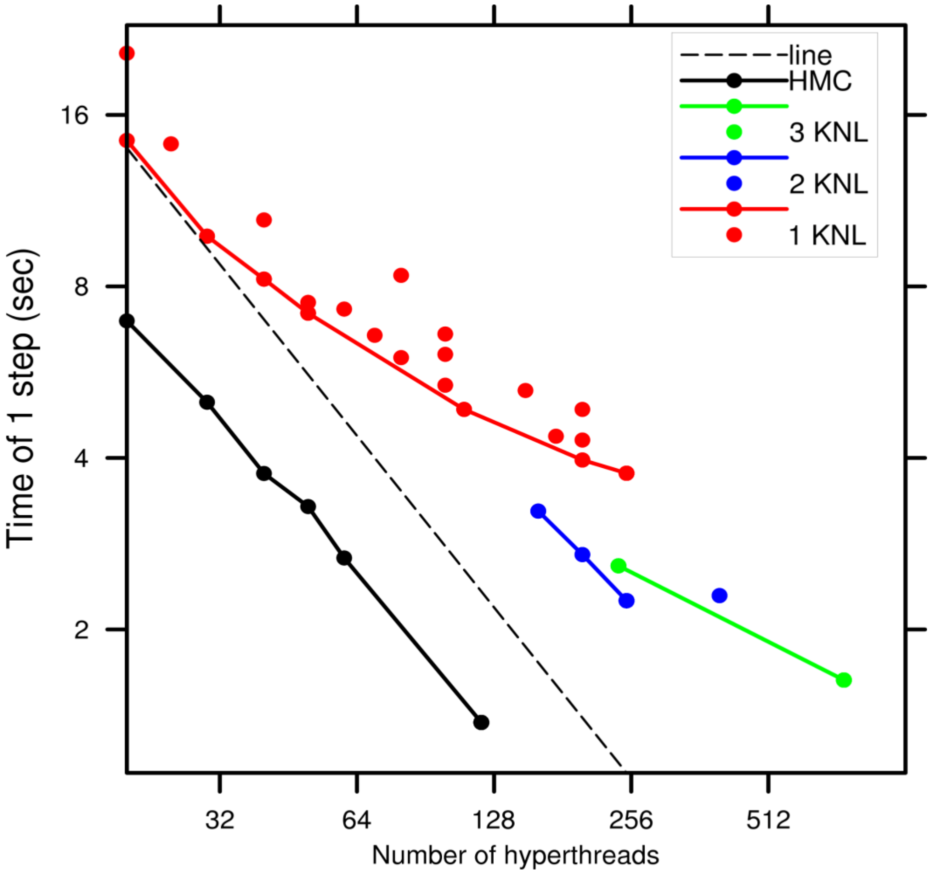

Fig.4. Scalability of SL-AV model code with 640x401x50 grid for different combinations of MPI processes and OpenMP threads. See text for details.

Acknowledgements. This study was carried out at the Institute of Numerical Mathematics, Russian Academy of Sciences. The study presented in Section 2 was supported with the Russian Science Foundation grant No. 14-27-00126, the work described in Section 3 was supported with the Russian Academy of Sciences Program for Basic Researches I.33P.

The authors thank Joint Supercomputer Center RAS (Moscow), RSC Company for giving access to their computer resources.

References

1. Tolstykh, M.A., Geleyn, J.-F., Volodin, E.M., Kostrykin, S.V., Fadeev, R.Y., Shashkin, V.V., Bogoslovskii, N.N., Vilfand, R.M., Kiktev, D.B., Krasjuk, T.V., Mizyak, V.G.,

Shlyaeva, A.V., Ezau, I.N., Yurova, A.Y.: Development of the multiscale version of the SL-AV global atmosphere model. Russ. Meteor. and Hydrol. **40**, 374-382 (2015). doi: 10.3103/S1068373915060035

2. Tolstykh, M., Shashkin, V., Fadeev, R., Goyman, G.: Vorticity-divergence semi-Lagrangian global atmospheric model SL-AV20: dynamical core. Geosci. Model Dev. **10**, 1961-1983 (2017), doi:10.5194/gmd-10-1961-2017.
3. Geleyn, J.-F., Bazile, E., Bougeault, P., Deque, M., Ivanovici, V., Joly, A., Labbe, L., Piedelievre, J.-P., Piriou, J.-M., Royer, J.-F.: Atmospheric parameterization schemes in Meteo-France's ARPEGE N.W.P. model. In: Parameterization of subgrid-scale physical processes, ECMWF Seminar proceedings, pp. 385–402, RCMWF, Reading, UK (1994).
4. Gerard, L., Piriou, J.-M., Brožková, R., Geleyn, J.-F., and Banciu, D.: Cloud and Precipitation Parameterization in a Meso-Gamma-Scale Operational Weather Prediction Model, Mon. Weather Rev. **137**, 3960–3977 (2009). doi:10.1175/2009MWR2750.
5. Müller, E., Scheichl, R.: Massively parallel solvers for elliptic partial differential equations in numerical weather and climate prediction, Q. J. Roy. Meteorol. Soc. **140**, 2608–2624 (2014). doi:10.1002/qj.2327
6. White III, J., Dongarra, J.: High-performance high-resolution tracer transport on a sphere. J. Comput. Phys. **230**, 6778 – 6799 (2011). doi:10.1016/j.jcp.2011.05.008
7. Robert, A.: A stable numerical integration scheme for the primitive meteorological equations. Atmos. Ocean, **19** 35–46, (1981). doi: 10.1080/07055900.1981.9649098
8. Goyman G.S.: Development of the parallel iterative Helmholtz problem solver for the SL-AV global atmospheric model. In: Proceedings of the 2nd Russian Conference on Supercomputing - Supercomputing Days. Pp. 959-966 (2016) [Goiman G. S. Razrabotka parallelnogo iterativnogo bloka reshenia tipa Gelmgoltca dlia globalnoi modeli atmosferi PLAV - Trudi mezhdunarodnoi konferencii 'Superkomp'iuternie dni Rossii' 2016. Izdatelstvo MGU im. M.V.Lomonosova P. 959-966]
9. Trottenberg, U., Oosterlee, C. W., Schuller, A.: Multigrid. – Academic press, 2000. 631 p.
10. Van der Vorst, H. A.: Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **13**, 631-644 (1992). doi: 10.1137/0913035
11. Larsson, J., Lien, F. S., Yee, E.: Conditional semicoarsening multigrid algorithm for the Poisson equation on anisotropic grids. J. Comput. Phys. **208**, 368-383 (2005). doi: 10.1016/j.jcp.2005.02.020
12. Barros, S. R. M.: Multigrid methods for two-and three-dimensional Poisson-type equations on the sphere. J. Comput. Phys. **92**, 313-348 (1991). doi:10.1016/0021-9991(91)90213-5