Parallel ADI Preconditioners for All-Scale Atmospheric Models

Zbigniew P. Piotrowski[1(✉)], Bartlomiej Matejczyk[2], Leszek Marcinkowski[3], and Piotr K. Smolarkiewicz[4]

[1] Institute of Meteorology and Water Management - National Research Institute, Podlesna 61, 01-673 Warsaw, Poland
zbigniew.piotrowski@imgw.pl
[2] Johann Radon Institute for Computational and Applied Mathematics, Linz, Austria
matejczyk.bartlomiej@gmail.com
[3] Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Warsaw, Poland
lmarcin@mimuw.edu.pl
[4] European Centre for Medium-Range Weather Forecasts, Reading, UK
piotr.smolarkiewicz@ecmwf.int

Abstract. Effective preconditioning lies at the heart of multiscale flow simulation, including a broad range of geoscientific applications that rely on semi-implicit integrations of the governing PDEs. For such problems, conditioning of the resulting sparse linear operator directly responds to the squared ratio of largest and smallest spatial scales represented in the model. For thin-spherical-shell geometry of the Earth atmosphere the condition number is enormous, upon which implicit preconditioning is imperative to eliminate the stiffness resulting from relatively fine vertical resolution. Furthermore, the anisotropy due to the meridians convergence in standard latitude-longitude discretizations becomes equally detrimental as the horizontal resolution increases to capture nonhydrostatic dynamics. Herein, we discuss a class of effective preconditioners based on the parallel ADI approach. The approach has been implemented in the established high-performance all-scale model EULAG with flexible computational domain distribution, including a 3D processor array. The efficacy of the approach is demonstrated in the context of an archetypal simulation of global weather.

Keywords: Deflation preconditioners · ADI · EULAG

1 Introduction

Modern numerical solvers, integrating complex and multiscale problems at the frontiers of geo- and astro-physics, constitute indispensable virtual laboratory for investigating inherently irreplicable phenomena, where traditional experimental approach becomes either impractical, infeasible or even entirely inapplicable. Many of these solvers target a particular range of scales and are essential for

R. Wyrzykowski et al. (Eds.): PPAM 2015, Part II, LNCS 9574, pp. 607–618, 2016.
DOI: 10.1007/978-3-319-32152-3_56

industrial or specialized scientific applications, such as Earth System Models for climate research, numerical forecasts suites from global- to meso-scale weather, down to small-scale CFD type models for cloud turbulence and wind engineering. In turn, multiscale solvers capable of integrating PDEs at scales from micro to stellar advance fundamental research in geo and astrophysics, contributing to understanding fundamental physical laws governing nonlinear phenomena. With the continuous advancement in supercomputing and the Big Data, spatial and temporal resolution increases and specialized applications are capable of capturing increasingly broader range of scales. This motivates further development and application of robust and accurate numerical techniques, capable of exploring finer scales admitted by improved resolution.

A notable example of such multiscale, multiphysics solver is an established model EULAG [12,19],[1] recently extended to include a consistent soundproof-compressible formulation of PDEs governing atmospheric dynamics [20]. Highly efficient parallel formulation of EULAG with a 3D MPI domain distribution has been reported in [10].

Explicit integration of the fully compressible atmospheric equations is impractical due to the inherent stability limits related to rapid propagation of acoustic modes. Various numerical techniques, often relying on the operator splitting, are used to mitigate this problem. Effectively, they employ specialized implicit integration schemes applied to selected terms in the flow PDEs. Adopted mathematical simplifications may require solution filtering, possibly leading to numerical artefacts [9]. In turn, fully implicit techniques like Newton-Krylov solvers may be prohibitively expensive. Among the variety of existing methods, the recently reported consistent formulation of the EULAG numerics appear especially prospective as it allows for efficient multiscale integrations using robust forward-in-time integrators and preconditioned nonsymmetric Krylov-subspace solvers.

In realistic multiscale applications, the elliptic Poisson and Helmholtz boundary value problems (corresponding to soundproof and compressible formulation of the governing PDEs) are nonsymmetric, semi-definite, poorly conditioned and complicated. This may result from the anisotropy of grid resolution, planetary rotation, ambient large scale gradients and stratification, the use of curvilinear coordinates, or the imposition of partial-slip conditions along irregular lower boundaries. Despite enormous advancements in numerical methods for sparse linear systems, the effective solution of large linear problems depends (at least for the class of problems discussed) not necessarily on the choice of the best available iterative method, but rather on the artful preconditioning.

In global models, anisotropy of the Earth's atmosphere results in the condition number $\kappa \sim O(10^{10})$, or larger. Because the asymptotic convergence rate of conjugate gradient type methods is proportional to $\sqrt{\kappa}$, the use of directly invertible preconditioner that removes the stiffness associated with a relatively fine vertical resolution is imperative [17]. To date, the Generalized Conjugate Residual (GCR) Krylov solver of EULAG was supported by a deflation technique, in

[1] For comprehensive list of EULAG publications see model webpage at http://www2.mmm.ucar.edu/eulag/.

which splitting the preconditioner into implicit and explicit counterparts in the vertical and horizontal, respectively, lead to subsequent direct inversion in the vertical of the explicit horizontal part [22]. Variants with spectral decomposition in the horizontal were also considered [18,23].

ADI methods per se date back to the nineteen fifties [2], and their use as preconditioners for Krylov solvers in atmospheric models was advocated nearly two decades ago [15,22]. However, with the transition to massively parallel computing, parallel implementation of tridiagonal algorithms underlying the ADI approach has been dismissed (e.g., in favour of the deflation technique as specified above), especially that vertical direction in atmospheric codes remained standardly serial — for the sake of radiation and precipitation processes, inherently sequential in nature and difficult to parallelize. Notwithstanding, more recently it become clear that at the exascale computing mere parallelization in the horizontal may be insufficient. The recent development of the 3D parallelization in EULAG brought parallel tridiagonal algorithms and paved the way for high-performance ADI preconditioners. Further acceleration of variational Krylov solvers is important, in order to minimize their global reductions anticipated at the exascale computing. In this context ADI preconditioners appear promising and worthy exploring. This seems to be reflected in their ceaseless popularity in the current literature.

2 Model Framework

2.1 Analytic Formulation

The consistent formulation of EULAG's governing equations casts (and solves) the governing PDEs in generalized time-dependent curvilinear coordinates $(\bar{t}, \overline{\mathbf{x}}) \equiv \big(t,\ F(t,\mathbf{x})\big)$, where the coordinates $(t,\mathbf{x})$ of the physical space are orthogonal and stationary, but not necessarily Cartesian. In particular, global simulations employ the standard anholonomic latitude-longitude (lat-lon) spherical framework (Sect. 7.2 in [3]) for the physical space [12,19], in which components of the physical velocity vector are aligned at every point of the spherical shell with axes of a local Cartesian frame tangent to the lower surface of the shell; cf. Fig. 7.7 in [3]. For simplicity of the presentation, here we dismiss the time-dependency of the model coordinates; so, $(\bar{t}, \overline{\mathbf{x}}) \equiv \big(t,\ F(\mathbf{x})\big)$ in the formulae that follow. The governing equations for the physical velocity $\mathbf{u} = (u, v, w)$ and the potential temperature θ can compactly be written as

$$\frac{d\mathbf{u}}{d\bar{t}} = -\Theta\,\widetilde{\mathbf{G}}\overline{\nabla}\varphi - \mathbf{g}\Upsilon_B\frac{\theta'}{\theta_b} - \mathbf{f}\times(\mathbf{u}-\Upsilon_C\mathbf{u}_e) \ \ + \mathcal{M}'(\mathbf{u},\mathbf{u}) + \overline{\mathcal{D}}_{\mathbf{v}}, \tag{1}$$

$$\frac{d\theta'}{d\bar{t}} = -\overline{\mathbf{u}}^{*}\cdot\overline{\nabla}\theta_e + \overline{\mathcal{D}}_{\theta}, \tag{2}$$

$$\frac{d\varrho}{d\bar{t}} = -\frac{\varrho}{\overline{\mathcal{G}}}\overline{\nabla}\cdot\overline{\mathcal{G}}\overline{\mathbf{u}}^{*}. \tag{3}$$

Here, the generalized density and pressure variables ϱ and φ are defined, respectively, for the [anelastic, compressible] PDEs as

$$\varrho := [\rho_b(z),\ \rho(\mathbf{x},t)]\ ,\quad \varphi := [c_p\theta_b\pi',\ c_p\theta_0\pi'], \tag{4}$$

where ρ and π denote the fluid density and the Exner pressure. The dimensionless coefficients are defined as

$$\Theta := \left[1,\ \frac{\theta(\mathbf{x},t)}{\theta_0}\right],\quad \Upsilon_B := \left[1,\ \frac{\theta_b(z)}{\theta_e(\mathbf{x})}\right],\quad \Upsilon_C := \left[1,\ \frac{\theta(\mathbf{x},t))}{\theta_e(\mathbf{x})}\right]. \tag{5}$$

On the LHS of (1) and (2), the total derivative $d/d\bar{t} = \partial/\partial\bar{t} + \overline{\mathbf{u}}^* \cdot \overline{\nabla}$, where $\overline{\mathbf{u}}^* = d\overline{\mathbf{x}}/d\bar{t}$ is the contravariant velocity in the computational space. The nabla operator $\overline{\nabla} \equiv \partial/\partial\overline{\mathbf{x}}$ represents the vector of partial derivatives corresponding to elementary finite differences in the model code. In the momentum Eq. (1), $\widetilde{\mathbf{G}}$ symbolizes a renormalized Jacobian matrix of the metric coefficients $\propto (\partial\overline{\mathbf{x}}/\partial\mathbf{x})$. The terms $\mathcal{M}(.,.)$ in (1) denote metric forces (viz. the generic Christoffel terms in the physical space), whereas $\mathbf{f}$ is the Coriolis acceleration, and $\mathcal{D}$ symbolizes the dissipation/diffusion operator. Subscript e refers to an ambient state, a particular solution to (1)–(3), while subscript b marks the anelastic base state [6]. The contravariant and physical velocities are related via

$$\overline{\mathbf{u}}^* = \widetilde{\mathbf{G}}^T\mathbf{u}, \tag{6}$$

and $\overline{\mathcal{G}}^2$ is the determinant of the metric tensor that defines the fundamental metric in the computational space.

2.2 Highlights of Numerical Approximations

Here we provide an outline of the nonoscillatory forward-in-time (NFT) numerical approximation strategy of EULAG. For comprehensive discussion and algorithmical details the interested reader is referred to [20] and references therein. The numerics of EULAG are unique, in that they rely on the Lagrangian/Eulerian congruence of governing equations integrated with the trapezoidal rule along flow trajectory in the 4D time-space continuum. In technical terms, however, the hydrodynamical solver of EULAG is reminiscent of the projection approach [1], where evaluation of an explicit part of the velocity $\mathbf{u}_{exp}$ is followed by the implicit completion of the solution, tantamount to the formulation and solution to the discrete boundary value problem (BVP) for pressure.[2] In particular, Eqs. (1)–(3) are cast in the conservation-law form of generalized transport equation for specific variable ψ (e.g. velocity component or potential temperature):

$$\frac{\partial\mathcal{G}\varrho\psi}{\partial\bar{t}} + \overline{\nabla}\cdot(\mathcal{G}\varrho\overline{\mathbf{u}}^*\psi) = \mathcal{G}\varrho R, \tag{7}$$

[2] This second step can be iterated when nonlinear terms resulting from, e.g., metric forces are present; cf. [19] for illustrative examples.

are integrated in time and space using MPDATA approach,[3] that provides a robust integrator for the entire system of the governing PDEs, including consistent formulation of a discrete BVP. The latter constraints the complete solution such as to assure that the updated solution satisfies the discrete mass continuity Eq. (3). In particular for the anelastic system, (3) takes an incompressible-like form

$$\frac{1}{\rho^*}\overline{\nabla}\cdot\rho^*\overline{\mathbf{u}}^* = 0, \tag{8}$$

where $\rho^* = \overline{\mathcal{G}}\rho_b$. therefore implying the Poisson BVP for φ

$$-\frac{\Delta t}{\rho^*}\frac{\partial}{\partial\overline{x}^j}\left[\rho^*\mathcal{E}\left(\widetilde{\mathcal{V}}^j - \widetilde{\mathcal{C}}^{jk}\frac{\partial\varphi}{\partial\overline{x}^k}\right)\right] = 0. \tag{9}$$

The problem in (9) can be though of as $\mathcal{L}(\varphi) - \mathcal{R} = 0$, wherein $\mathcal{E}$ and $\widetilde{\mathcal{C}}^{jk}$ denote 10 fields of known coefficients that generally vary in time and space; $\mathcal{E}\widetilde{\mathcal{V}}^j$ is the jth component of the explicit part of contravariant velocity solution; and repeating k indices imply the summation over the components of $\overline{\nabla}\varphi$; see [11] for the exposition. The multiplicative factor $-\Delta t/\rho^*$ assures the formal negative semi-definiteness of $\mathcal{L}$ and expresses the residual error $r = \mathcal{L}(\varphi) - \mathcal{R} \neq 0$ as the divergence of a local Courant number on the grid. Notably, for the compressible solver, the resulting Helmholtz BVP is composed of three Poisson-like operators and the term proportional to φ [20]. In either case, the resulting BVP is solved (subject to appropriate boundary conditions) using the GCR approach [4,21] — a robust preconditioned nonsymmetric Krylov-subspace solver akin to GMRES [14]. Given the updated pressure, and hence the updated contravariant velocity, the updated physical velocity components are constructed from (6).

A key element of our GCR machinery is the (left) operator preconditioning providing an estimate of the solution error $q = \varphi - \varphi_{exact}$ as

$$q = \mathcal{P}^{-1}(r), \tag{10}$$

where the preconditioner $\mathcal{P} \approx \mathcal{L}$ but is easier to invert than the $\mathcal{L}$. In EULAG, $\mathcal{P}$ closely matches $\mathcal{L}$ by only neglecting the cross derivative terms with coefficients $\widetilde{\mathcal{C}}^{jk}\big|_{k\neq j}$. In the remainder of the paper we elaborate on technical aspects of (10).

3 Implicit Inversion Preconditioning

3.1 Principles

Standard EULAG preconditioning relies on the direct inversion of the vertical component of the implicit operator, while evaluating horizontal part of the operator explicitly using a stationary Richardson iteration

$$\frac{q^{\mu+1} - q^{\mu}}{\Delta\tau} = P^z q^{\mu+1} + P^h q^{\mu} - r^{\nu}, \tag{11}$$

[3] MPDATA (for multidimensional positive definite advection transport algorithm) is a class of nonoscillatory forward-in-time flow solvers, widely documented in the literature; for a recent overview see [20] and references therein.

where P^z and P^h symbolize vertical and horizontal parts of $\mathcal{P} = P^z + P^h$; $\Delta\tau$ is a fixed pseudo-time step (selected such as to assure convergence of the iterative process; μ numbers preconditioner's iterations (usually a few to several), and r^ν is the residual error of the Krylov solver's νth iteration. Gathering all $q^{\mu+1}$ terms on the LHS, leads to the tridiagonal problem

$$(\Delta\tau^{-1}\mathcal{I} - P^z)q^{\mu+1} = \Delta\tau^{-1}q^\mu + P^h q^\mu - r^\nu := \widetilde{r}, \tag{12}$$

readily invertible with the Thomas algorithm concisely symbolized as

$$q^{\mu+1} = (\Delta\tau^{-1}\mathcal{I} - P^z)^{-1}\widetilde{r} \tag{13}$$

The developed ADI preconditioners enable extending the standard preconditioner (11) to admit implicitness in P^h and, thus, accelerate the convergence by increasing $\Delta\tau$. In particular, a two-dimensional ADI design leaves only single explicit direction. When operating on the global lat-lon grid, the natural choice is to treat the longitudinal direction implicitly, as the meridians converging towards poles can introduce considerable anisotropy in the coefficients of the horizontal operator P^h. While numerous formulations of a 2D ADI are possible, a particularly simple algorithm was provided by Peaceman and Rachford [8],

$$\begin{aligned}
\frac{q^{\mu+\frac{1}{2}} - q^\mu}{\Delta\tau_2} &= P^x q^{\mu+\frac{1}{2}} + P^z q^\mu &&+ P^y q^\mu - r^\nu,\\
\frac{q^{\mu+1} - q^{\mu+\frac{1}{2}}}{\Delta\tau_2} &= P^x q^{\mu+\frac{1}{2}} + P^z q^{\mu+1} &&+ P^y q^\mu - r^\nu,
\end{aligned} \tag{14}$$

where P^x and P^y symbolize, respectively, the longitude and latitude counterparts of $P^h = P^x + P^y$. To extend (14) to all three directions, we adopt the unconditionally stable 3D ADI Douglas [2] algorithm

$$\begin{aligned}
\tfrac{q^{\mu+\frac{1}{3}} - q^\mu}{\Delta\tau_3} &= P^x \tfrac{(q^{\mu+\frac{1}{3}} + q^\mu)}{2} + P^y q^\mu &&+ P^z q^\mu &&- r^\nu,\\
\tfrac{q^{\mu+\frac{2}{3}} - q^\mu}{\Delta\tau_3} &= P^x \tfrac{(q^{\mu+\frac{1}{3}} + q^\mu)}{2} + P^y \tfrac{(q^{\mu+\frac{2}{3}} + q^\mu)}{2} &&+ P^z q^\mu &&- r^\nu,\\
\tfrac{q^{\mu+1} - q^\mu}{\Delta\tau_3} &= P^x \tfrac{(q^{\mu+\frac{1}{3}} + q^\mu)}{2} + P^y \tfrac{(q^{\mu+\frac{2}{3}} + q^\mu)}{2} &&+ P^z \tfrac{(q^{\mu+1} + q^\mu)}{2} &&- r^\nu.
\end{aligned} \tag{15}$$

Grouping all implicit terms of (14) and (15) on the LHS, as in (12), shows that all three preconditioners share a common template algorithm

$$q^{\mu*} = (\Delta\tilde{\tau}_i^{-1}\mathcal{I} - P^I)^{-1}\widetilde{r}^* \tag{16}$$

for the tridiagonal solver in $I = x, y, z$ directions at some intermediate step $\mu*$. $\widetilde{r}^*$ denotes the corresponding explicit elements of (14) and (15).[4] This substantiates the earlier assertion that the parallel implementation of the tri-diagonal inversion in one direction paves the way for the family of ADI preconditioners.

[4] Note that $\tilde{\tau}_i = \tau_i$ for (11)–(14) but $\tilde{\tau}_i = \tau_i/2$ in the (15).

3.2 Parallel Implementation

At the highest level, the implementation of algorithms (11)–(15) in EULAG follows the mathematical notation. Notably, as the first guess $q^{\mu=0} = 0$, the initlaisation of the algorithms needs bespoke script to avoid unnecessary memory references. In particular, since $\forall^I \; P^I q^{\mu=0} = 0$ in the first preconditioner iteration, it is only necessary to evaluate half of the explicit members of (11)–(15). Furthermore, the algorithms (11)–(15) can be judiciously mixed, whereupon we evaluate all members of the preconditioner family as they all can be useful in different classes of applications.

The BVP problems in EULAG rarely can be solved in single GCR iteration. For compressible equations the problem matrix is constant within given timestep. For anelastic equations in stationary coordinates it remains constant throughout the entire integration time. Inspection of the standard tridiagonal algorithm suggests that it is possible to precompute a good part of its forward step. Consequently, all variants of preconditioners in EULAG begin with a suitable initialization executed either once per simulation or time step, depending on the application at hand. This requires relatively large additional storage that is, however, rarely an issue, on the modern CPU clusters.[5]

The parallel tridiagonal inversion, if implemented in the form of naive parallel recurrence, suffers from the adverse load balance characteristics, i.e. only one core in the column may be active at the time. However, it is fairly easy to mitigate this problem if using two cores in vertical. With the given horizontal subdomain, at half of the gridpoints the forward tridiagonal sweep starts from one boundary while the remaining columns are evaluated from the forward sweep starting from the opposite boundary. Importantly, this technique does not noticeably alter the physical results. If the use of larger number of cores in the direction of tridiagonal inversion is needed and the cost of computations per core is significant, it is possible to use Pipelined Thomas Algorithm aiming at quickest possible fill of pipeline of computing cores in given direction. This is done by splitting the processor subdomain into chunks and communicating the results to the next core as soon as recurrence (acting in direction normal to the chunk surface) is executed; see Sect. 2 in [10] and references therein. While this strategy seems to be efficient for low speed computing cores, such as employed by IBM Bluegene/L architecture, we found the strategy in the spirit of recursive doubling [24] to be more effective on modern CPU clusters. The Thomas algorithm as employed for the A-grid discretization of EULAG for nonperiodic boundary conditions has the general form

$$e_k = A_k / \left(B_k - C_k e_{k-2}\right) \tag{17}$$

$$f_k = D_k(e_k) f_{k-2} + Q_k(e_k, rhs_{ADI}) \tag{18}$$

$$p_k = e_k p_{k+2} + f_k \tag{19}$$

[5] For the majority of geophysical applications, the main performance bottleneck is the memory bandwidth and access time, and not the main memory size. The latter is usually much larger than needed, as large number of timesteps and good scalability lead to routine use of hundreds of computing cores and tens of supercomputer nodes.

where k is indexes either x,y or z direction, A, B, C are known 3D matrices, D is a function of e_k and Q is a function of instantaneous right hand side rhs_{ADI} at a given step of an algorithm at hand from the suite (11)–(15). Because e_k can be precomputed, this allows to express last element of the recurrence by the first or the second element of the recurrence, for the last k index allowed being odd and even, respectively. For example, for the recurrence split in four-element segments

$$a_k = b_k a_{k-2} + c_k = b_k b_{k-2} a_{k-4} + b_k c_{k-2} + c_k. \tag{20}$$

The auxiliary product $b_k b_{k-2}$ and the remaining $b_k c_{k-2} + c_k$ term can be precomputed at additional cost, but in fully parallel way. This reduces the cost of purely sequential part of the algorithm (per "chunk surface" gridpoint in computational subdomain) to single multiply/add operation along with the three memory loads and single store and we will later refer to it as LFF (Last From First) operation. For the whole computational domain the parallel tridiagonal algorithm employs two possible forms, on the sequence of *nproc* computational cores denoted as subscripts in a given direction

$$LFF_1 \rightarrow SEND_{1\ to\ 2} \rightarrow LFF_2 \rightarrow ... \rightarrow SEND_{nproc-1\ to\ nproc} \rightarrow LFF_{nproc}$$

or a variant employing one-dimensional MPI Gather/Scatter operations

$$MPIgather_{input\ for\ LFF} \rightarrow \forall_{1,nproc} LFF \rightarrow MPIscatter_{LFF\ results}.$$

Ultimately, the choice of optimal parallel inversion strategy depends on the problem and the computing architecture at hand.

4 Computational Efficiency and Performance

The suite of the preconditioners has been benchmarked for the baroclinic instability experiment (BAROC) [5], an archetype of the global weather. Details of BAROC implementation in EULAG are presented in [13] together with the grid convergence study and the comparison against the hydrostatic solutions in [5]. Further studies of BAROC included simulations with anelastic, pseudo-incompressible and fully compressible equations [20] using several explicit and implicit formulations of either flux-form Eulerian of semi-Lagrangian EULAG integrators.

The model setup assumes the analytically prescribed ambient state consisting of two mid-latitude zonal jets symmetric about the equator that are in unstable thermal wind equilibrium with the corresponding meridional distribution of the potential temperature. Initial velocity field is locally perturbed at the northern hemisphere, leading to the development of the instability manifested with the fastest growing eastward propagating Rossby mode of wavenumber 6. For illustration, the top panels of Fig. 1 show surface θ' solutions using anelastic and compressible PDEs. Evident frontogenesis is similar to the observed weather systems at the planetary scales.[6]

[6] For a discussion of the differences in various soundproof and compressible solutions see [20] and references therein.

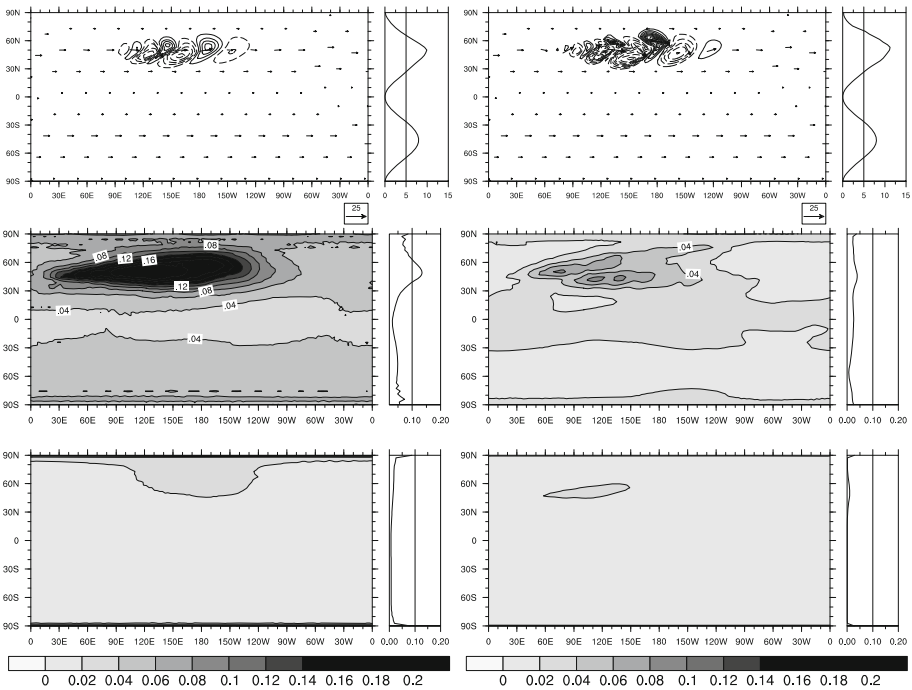

Fig. 1. Anelastic (left) and compressible (right) solutions for the baroclinic instability benchmark at day 8. Top panels display the isolines of surface potential temperature perturbation, overlaid with the flow vectors, on a regular longitude-latitude grid. Contour intervals are 4 K, dashed/solid lines correspond to negative/positive contour values, and zero contour lines are not shown. Zonally averaged profiles to the right of the contour plots depict mean kinetic energy over the full 10-day integration. Middle panels refer to the model run employing standard preconditioner (11) with 3 preconditioner iterations per each GCR iteration. The contour plots show the vertically averaged 10-day-mean residual errors, with their zonally averaged profiles appearing on the right. The bottom panels convey corresponding measures for the runs employing ADI preconditioner (14) with one preconditioner iteration per each GCR iteration. Contour values in the middle and bottom panels refer to the fraction of maximum allowed residual error $r_{max} = 10^{-7}$; cf. [9] and the accompanying discussion.

The reported calculations employ a coarse lon-lat "research" grid 128×64 (corresponding to 2.8^o horizontal resolution) and 23 Km deep atmosphere resolved with 48 equally spaced vertical levels. The implicit solver employs physically-based stopping criteria [16] and iterates until the L_∞ norm of the residual error is no larger than 10^{-7}; i.e., orders of magnitude smaller than maximal Courant number $\lesssim 1$. All calculations use 2880 timesteps with $\delta t = 300$ s. The resulting residual errors displayed in Fig. 1, for the runs using three iterations of the standard preconditioner (11) and the runs using one iteration of the two directional ADI preconditioner (14), document substantial improvement

in the solution accuracy for the latter. In particular, mean residual error in the runs with the standard preconditioner evinces a signature of nonlinear processes occurring in the northern hemisphere, which is dramatically reduced in the runs with one ADI x-z pass per each GCR iteration. Similarly, effects due to meridian convergence and the refined Δx are practically removed, except in the immediate proximity of the poles. Interestingly, tests conducted with the fully implicit ADI preconditioner (15) essentially show no further improvement compared to ADI x-z [7].

Table 1. Performance of the standard (11) and ADI (14) preconditioners for the anelastic and compressible BAROC experiment, presented in the middle and bottom panels of Fig. 1, respectively. First column identifies the governing PDEs and the preconditioner used. Subsequent columns list: time-to-solution (TTS); time spent in elliptic solver (TES); time spent in the preconditioner (TPR), all in seconds; and the total number of the linear operator evaluations (NLE).

PDE; preconditioner	TTS	TES	TPR	NLE
Anelastic; standard	102	80	35	175386
Anelastic; ADI x-z	48	34	13	105245
Compressible; standard	61	36	20	52593
Compressible; ADI x-z	52	30	12	58225

Table 1 summarizes computational performance of the runs illustrated in Fig. 1. All experiments were performed on the Cray XC30 "Piz Daint" at CSCS, using Cray Fortran compiler with strong optimization options, "-O aggress, cache3, scalar3, fp3, ipa5, vector3, thread0". Standard preconditioner runs were distributed in a $16 \times 8 \times 1$ processor array, whereas ADI x-z runs used a $2 \times 16 \times 4$ array. The latter distribution was motivated by the fact that the tridiagonal solver is more expensive for periodic x direction than in the non-periodic z direction. The total cost of model integration is reflected in the second column, showing that the ADI x-z runs are altogether 50 % cheaper in the anelastic simulations, and 15 % cheaper in the compressible simulations. This discrepancy in the performance improvement results from the much lower overall number of the linear operator evaluations in the compressible case, regardless of the preconditioner used. Although the GCR performs 10 % more iterations in the compressible run with the ADI x-z than with the standard preconditioner, the overall performance is improved due to the lower cost of evaluating the preconditioner itself.

5 Remarks

It is often emphasized that regular lat-lon grids are disadvantageous for global modeling, because the meridians convergence near the poles necessitates small

timesteps to assure computational stability and impairs conditioning of the linear operator in semi-implicit flow solvers (thus resulting in a poor performance of iterative elliptic solvers). The first aspect can be circumvented by resorting to semi-implicit semi-Lagrangian integrators. Here we addressed the second aspect and showed that it can be mitigated with ADI preconditioners employing implicit inversions in x and z directions. In particular, for highly anisotropic grids, this permits much larger preconditioner's pseudo-timestep τ (e.g., 40 and 10 larger with ADI x-z in the discussed anelastic and compressible runs, respectively), leading to significantly faster convergence of the elliptic solver and consequently to reduction of the time-to-solution. Benefits of ADI preconditioning are expected to be even more pronounced when MPI parallelization scheme of EULAG will be supplemented by the shared memory decomposition.

Acknowledgements. This work was supported by "Towards peta-scale numerical weather prediction for Europe" project realized within the "HOMING PLUS" programme of Foundation for Polish Science, co-financed from European Union, Regional Development Fund. Selected code optimizations were supported by the Polish National Science Center (NCN) under the Grant no.: 2011/03/B/ST6/03500. Piotr K. Smolarkiewicz is supported by funding received from the European Research Council under the European Union's Seventh Framework Programme (FP7/2012/ERC Grant agreement no. 320375). This work was supported by a grant from the Swiss National Supercomputing Centre (CSCS) under project ID d25 and by the Interdisciplinary Centre for Mathematical and Computational Modelling (ICM) University of Warsaw under grant no. G49-15.

References

1. Chorin, A.J.: Numerical solution of the Navier-Stokes equations. Math. Comp. **22**, 742–762 (1968)
2. Douglas, J.J.: On the numerical integration of $u_t = u_{xx} + u_{yy}$ by implicit methods. SIAM J. **3**(1), 42–65 (1955)
3. Dutton, J.A.: The Ceaseless Wind. McGraw-Hill, New York (1976)
4. Eisenstat, S.C., Elman, H.C., Schultz, M.H.: Variational iterative methods for nonsymmetric systems of linear equations. SIAM J. Numer. Anal. **20**(2), 345–357 (1983)
5. Jablonowski, C., Williamson, D.L.: A baroclinic instability test case for atmospheric model dynamical cores. Q. J. Roy. Meteorol. Soc. **132**(621C), 2943–2975 (2006)
6. Lipps, F.B., Hemler, R.S.: A scale analysis of deep moist convection and some related numerical calculations. J. Atmos. Sci. **39**(10), 2192–2210 (1982)
7. Matejczyk, B.: Preconditioning in mathematical weather forecasts. Master's thesis, University of Warsaw (2014)
8. Peaceman, D.W., Racheford, H.H.: The numerical solution of parabolic and elliptic numerical differential equations. SIAM J. **3**(1), 28–41 (1955)
9. Piotrowski, Z.P., Smolarkiewicz, P.K., Malinowski, S.P., Wyszogrodzki, A.A.: On numerical realizability of thermal convection. J. Comp. Phys. **228**(17), 6268–6290 (2009)

10. Piotrowski, Z.P., Wyszogrodzki, A.A., Smolarkiewicz, P.K.: Towards petascale simulation of atmospheric circulations with soundproof equations. Acta Geophys. **59**(6), 1294–1311 (2011)
11. Prusa, J.M., Smolarkiewicz, P.K.: An all-scale anelastic model for geophysical flows: dynamic grid deformation. J. Comput. Phys. **190**(2), 601–622 (2003)
12. Prusa, J.M., Smolarkiewicz, P.K., Wyszogrodzki, A.A.: EULAG, a computational model for multiscale flows. Comput. Fluids **37**(9), 1193–1207 (2008)
13. Prusa, J.M., Gutowski, W.J.: Multi-scale waves in sound-proof global simulations with EULAG. Acta Geophys. **59**(6), 1135–1157 (2011)
14. Saad, Y., Schultz, M.H.: GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **7**(3), 856–869 (1986)
15. Skamarock, W.C., Smolarkiewicz, P.K., Klemp, J.B.: Preconditioned conjugate-residual solvers for Helmholtz equations in nonhydrostatic models. Mon. Weather Rev. **125**(4), 587–599 (1997)
16. Smolarkiewicz, P.K., Grubisic, V., Margolin, L.G.: On forward-in-time differencing for fluids: stopping criteria for iterative solutions of anelastic pressure equations. Mon. Weather Rev. **125**(4), 647–654 (1997)
17. Smolarkiewicz, P.K., Margolin, L.G., Wyszogrodzki, A.A.: A class of nonhydrostatic global models. J. Atmos. Sci. **58**(4), 349–364 (2001)
18. Smolarkiewicz, P.K., Temperton, C., Thomas, S.J., Wyszogrodzki, A.A.: Spectral preconditioners for nonhydrostatic atmospheric models: extreme applications. In: Proceedings of the ECMWF Seminar Series on Recent Developments in Numerical Methods for Atmospheric and Ocean Modelling, Reading, UK, pp. 203–220 (2004)
19. Smolarkiewicz, P.K., Charbonneau, P.: EULAG, a computational model for multiscale flows: an MHD extension. J. Comput. Phys. **236**, 608–623 (2013)
20. Smolarkiewicz, P.K., Kühnlein, C., Wedi, N.P.: A consistent framework for discrete integrations of soundproof and compressible PDEs of atmospheric dynamics. J. Comput. Phys. **263**, 185–205 (2014)
21. Smolarkiewicz, P., Margolin, L.: Variational solver for elliptic problems in atmospheric flows. Appl. Math. Comp. Sci **4**(4), 527–551 (1994)
22. Smolarkiewicz, P., Margolin, L.: Variational methods for elliptic problems in fluid models. In: Proceedings of ECMWF Workshop on Developments in Numerical Methods for Very High Resolution Global Models, pp. 137–159 (2000)
23. Thomas, S.J., Hacker, J.P., Smolarkiewicz, P.K., Stull, R.B.: Spectral preconditioners for nonhydrostatic atmospheric models. Mon. Weather Rev. **131**(10), 2464–2478 (2003)
24. Zhang, Y., Cohen, J., Owens, J.D.: Fast tridiagonal solvers on the GPU. ACM Sigplan Not. **45**(5), 127–136 (2010)