ENGINEERING MATHEMATICS
AND COMPUTING LAB

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Nested Schur-Complement Solver for a Low-Mach Number Model: Application to a Cyclone-Cyclone Interaction

Gawlok, S., Heuveline, V.

Preprint No. 2018-02

Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)

www.emcl.iwr.uni-heidelberg.de

Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)
ISSN 2191–0693
Preprint No. 2018-02

Affiliation of the Authors

Gawlok, S.[a], Heuveline, V.[a]

[a] *Engineering Mathematics and Computing Lab (EMCL), Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Germany*

Impressum

Heidelberg University
Interdisciplinary Center for Scientific Computing (IWR)
Engineering Mathematics and Computing Lab (EMCL)

Im Neuenheimer Feld 205,
69120 Heidelberg
Germany

Nested Schur-Complement Solver for a Low-Mach Number Model: Application to a Cyclone-Cyclone Interaction

Gawlok, S., Heuveline, V.

September 27, 2018

Abstract

Forecasting the dynamical evolution of interacting tropical cyclones poses a computationally challenging problem. An attractive approach for modeling and simulating such type of dynamics is given by Low-mach models. A formulation of a Low-mach model is presented along with the description of a benchmark scenario of two interacting tropical cyclones. For the discrete nonlinear equations of this model, an improved variant of the nested Schur-complement preconditioner is proposed. A detailed discussion of the numerical results is conducted with a special emphasis on computational costs and scalability. It will be demonstrated that the new preconditioner clearly outperforms a prior version of this kind of preconditioner with respect to computing time by a factor of two, but at the cost of a slightly reduced scalability.

1 Introduction

The task of forecasting the motion and evolution of the interaction of tropical cyclones is a challenging and computationally expensive task [6, 16, 17, 19, 31]. The underlying physical processes interact in complex ways on a wide range of spatial and temporal scales, which needs to be considered in the discretization of the underlying physical models by means of the resolutions of both the temporal and spatial computational grids. Consequently, a numerical simulation of such processes comprises the solution of very large and possibly non-linear systems of equations.

A very well-known method for solving non-linear systems of equations is Newton's algorithm [30]. In each iteration of this algorithm a linearization of the non-linear system of equations needs to be solved, which constitutes the computationally most expensive step in each iteration. Therefore, effective and efficient linear solvers and corresponding preconditioners are needed in order to solve the linearized system robustly and in a feasible amount of time. Especially, the applied preconditioner plays a crucial role and should be adapted to the physics of the underlying physical model, see, e.g., [19] and the references therein.

A possible model to describe the governing physical processes is the so-called Low-mach number approximation of the Compressible Navier-Stokes equations, see, e.g., [7, 19, 25–27]. This model is expressed in the form of a system of non-linear partial differential equations (PDEs). A model-adapted preconditioner for the solution of the linearized system in each iteration of Newton's method is given by the nested Schur-complement preconditioner, which has been introduced in [19].

The idea of the nested Schur-complement preconditioner is to split the linearized system of Newton's method into smaller sub-systems by means of Schur-complements, which can be assigned a physical meaning [19] based on the underlying PDE model. The solution of the resulting Schur-complement equations involves solving linear systems of equations of sub-systems of the linearzied system, which can be solved by standard techniques, e.g., Krylov subspace methods [19, 33] and multigrid techniques [19, 33]. Consequently, the application of a nested Schur-complement preconditioner comprises the approximate solution of several linear systems. In order to achieve a good overall performance and short computing

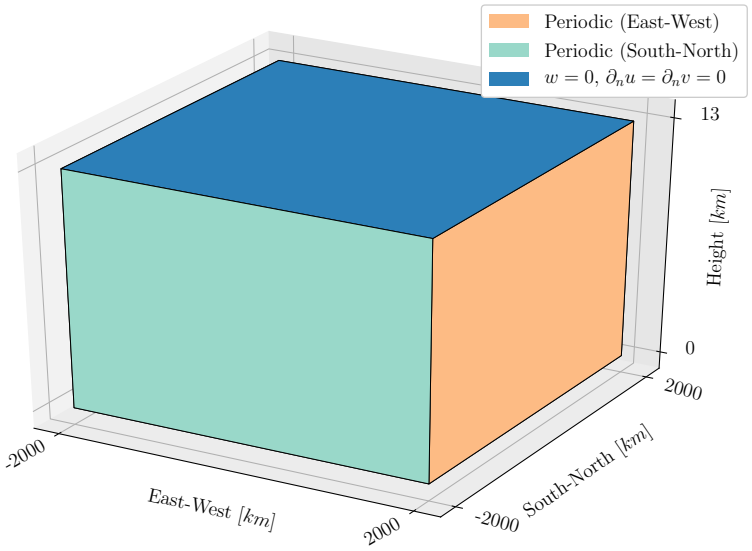

Figure 1: Domain Ω and boundary conditions for CCI scenario.

times, the parameters controlling the corresponding solution algorithms need to be chosen carefully and allow for many possible improvements depending on the number of selectable parameters.

In this article, we present improvements on the parameter choices that have been achieved in comparison to [19]. These improvements lead to a cut of the computational time by a factor of 2.

The remainder of this article is organized as follows: In Section 2, we shortly introduce the Low-mach (LM) model, the setup of the Cyclone-Cyclone interaction (CCI) scenario as well as the discretization of the underlying PDEs. In the following Section 3, the improved nested Schur-complement preconditioner is presented. The numerical results achieved by the improved preconditioner are discussed in Section 4. Section 5 gives a summary as well as an outlook on possible topics for further developments and research.

2 Cyclone-Cyclone Interaction and Low-mach Model

Typically, tropical cyclones have diameters on the scale of several 100 km. In the considered scenario, two cyclones of this type, which interact with each other, are placed in the computational domain with an initial distance of the storm centers of 400 km. Therefore, the horizontal extend of the dynamic evolution of the two cyclones easily reaches the scale of 1000 km. Consequently, the domain Ω needs to be chosen large enough such that the cyclones are still fully contained within the domain on the considered time-interval. In the presented case, the domain extends over 4000 km in both horizontal directions, and 13 km in the vertical. Horizontally, the domain is centered around the origin of the coordinate system, i.e., the domain Ω is defined as

$$\Omega := [-2,000,000; 2,000,000] \times [-2,000,000; 2,000,000] \times [0; 13,000], \tag{1}$$

where the boundaries of the intervals are given in meters $[m]$. The domain Ω as well as the applied boundary conditions for Problem 2.1, respectively, are depicted in Figure 1.

The physical model for the evolution of the fluid dynamics, which is considered here, is the so-called *Low-mach* number approximation of the *compressible Navier-Stokes equations* for a dry atmosphere, see, e.g., [19] and the references therein. The governing equations of this model are given as follows:

Problem 2.1 (Low-mach model [19])

Let $\Omega \subset \mathbb{R}^3$ be as in (1) and $T \geq 0$ a final point in time. Find a velocity field $\mathbf{v} := (u, v, w)^\top : [0, T) \times \Omega \to \mathbb{R}^3$, a density perturbation $\rho^* : [0, T) \times \Omega \to \mathbb{R}$, a temperature perturbation $\theta_v^* : [0, T) \times \Omega \to \mathbb{R}$, a pressure

perturbation $p^* : [0,T) \times \Omega \to \mathbb{R}$ and a thermodynamic pressure $p_{th} : [0,T) \to \mathbb{R}$, which fulfills

$$\int_\Omega p^* \mathrm{d}x = 0, \tag{2}$$

satisfying

$$\partial_t \mathbf{v} + (\mathbf{v} \cdot \nabla)\mathbf{v} + \frac{1}{\rho}\nabla p^* - \nu_a \Delta \mathbf{v} + \left(-fv, fu, \frac{\rho^*}{\rho} g\right)^\top = 0 \tag{3}$$

$$\partial_t p_{th} + w \partial_z p_0 + \frac{p_{th} + p_0}{1-\kappa} \mathrm{div}\ \mathbf{v} = 0 \tag{4}$$

$$\partial_t \theta_v^* + w \partial_z \theta_{v,0} + (\mathbf{v} \cdot \nabla)\, \theta_v^* = 0 \tag{5}$$

$$\frac{\left(\frac{p_0}{p_{th}+p_0}\right)^\kappa p_{th} \theta_{v,0} + \left[\left(\frac{p_0}{p_{th}+p_0}\right)^\kappa - 1\right] p_0 \theta_{v,0} - p_0 \theta_v^*}{\left(-\frac{g\kappa}{R'\theta_z} \ln\left(1 + \frac{\theta_z z}{\theta_0}\right) + 1\right) R'\, (\theta_{v,0} + \theta_v^*)\, \theta_{v,0}} = \rho^* \tag{6}$$

$$\partial_t p_{th} - \frac{\int_\Omega \kappa w \partial_z p_0 \mathrm{d}x}{(1-\kappa)\,|\Omega|} = 0 \tag{7}$$

$$w = 0, \quad \partial_{\mathbf{n}} u = 0, \quad \partial_{\mathbf{n}} v = 0 \quad \text{on} \quad [0,T] \times \Gamma \tag{8}$$

$$\mathbf{v}(0,x) = \mathbf{v}_0(x), \quad \rho^*(0,x) = \rho_0^*(x), \quad \theta_v^*(0,x) = \theta_{v,0}^*(x), \quad p^*(0,x) = p_0^*(x), \quad p_{th}(0) = 0 \tag{9}$$

as well as periodic boundary conditions in both horizontal directions for all variables $\mathbf{v}$, ρ^*, θ_v^* and p^*,

$$\rho_0^* := \frac{(1000\ hPa)^\kappa \left(p_{th}(t) + p_0(x)\right)^{1-\kappa}}{R'\left(\theta_{v,0}^* + \theta_0\right)} - \rho_0$$

and

$$\rho := \rho^* + \rho_0.$$

Equations (3)-(6) are required to hold on $(0,T) \times \Omega$ and (9) is asked to hold on $\{t = 0\} \times \Omega$.

The unknown functions $\mathbf{v}$, ρ^*, θ_v^* and p^* are discretized in space by means of finite elements and by means of finite differences in time [19]. The domain Ω is triangulated admissibly in congruent hexahedra. Based on this triangulation, finite elements of Lagrange type with trilinear basis polynomials are chosen for all six unknown functions, i.e., a $\mathbb{Q}_1/\mathbb{Q}_1/\mathbb{Q}_1/\mathbb{Q}_1/\mathbb{Q}_1/\mathbb{Q}_1$ discretization is chosen in space [19]. Also, all finite dimensional test function spaces are chosen to be defined by the $\mathbb{Q}_1$ discretization of the domain Ω by hexahedra.

For the discretization in time, in the momentum equation all terms are treated in a Crank-Nicolson manner except for the pressure part p^*, which is treated in an implicit Euler manner. The continuity equation is discretized by the implicit Euler scheme in time, whereas the thermodynamic energy equation is discretized by the Crank-Nicolson time-stepping scheme.

The initial conditions for the CCI scenario are depicted in Figure 2. Please refer to [19] for further details and visualizations.

3 Nested Schur-Complement Approach

The resulting discrete nonlinear system of equations is solved with an inexact Newton method [30], where the linearized system in each Newton step is solved with the Flexible Generalized Minimum Residual Method (FGMRES) with projections on the space, where the hydrodynamic pressure p^* incorporates zero mean value. This FMGRES algorithm is preconditioned by a nested Schur-complement approach. In the following, we concentrate on describing the improved nested Schur-complement solver. For the sake of clarity of the presentation, details on the inexact Newton method as well as on the Flexible Generalized Minimum Residual Method (FGMRES) with projections on the space, where the hydrodynamic pressure

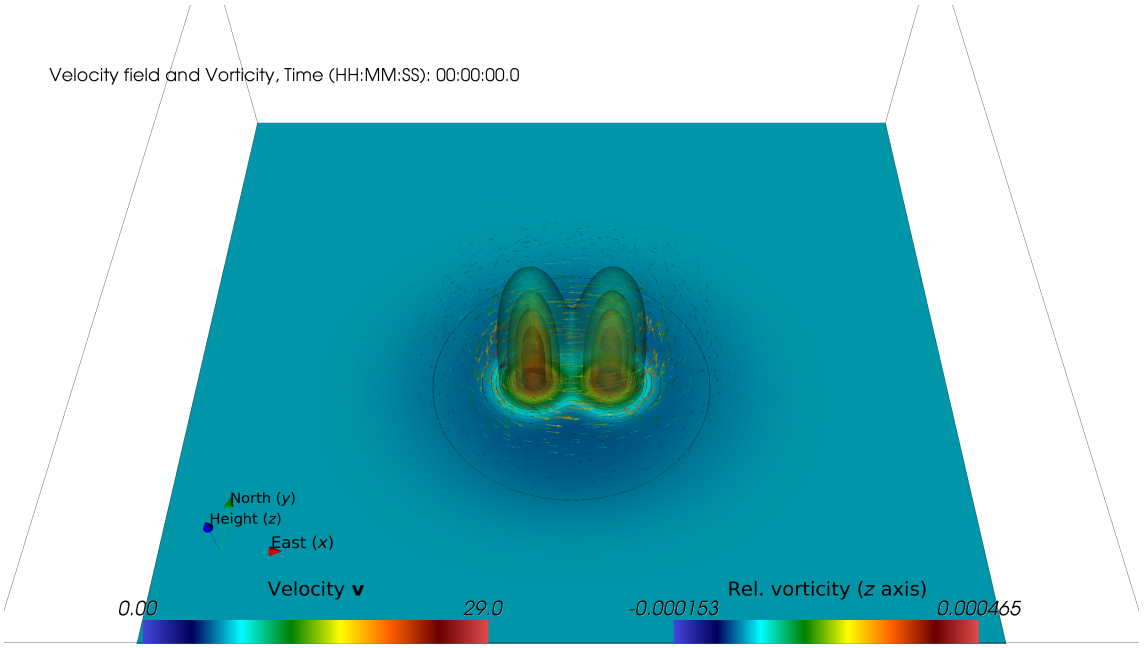

Figure 2: Initial velocity field $\left[\frac{m}{s}\right]$ and vertical vorticity component $\left[\frac{1}{s}\right]$ [19].

p^* incorporates zero mean value, are omitted here. We refer the interested reader to [19]. An overview of the overall solution process is depicted in Figure 3

We will reproduce parts of the presentation of nested Schur-complements for the Low-Mach model given in [19] here in order to precisely describe the changes compared to [19].

The idea of the Schur-complement algorithm is as follows: Let a linear system $\mathcal{A}\xi = \mathbf{b}$ in block matrix form

$$\mathcal{A}\xi = \left(\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right) \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix} \tag{10}$$

be given and assume, that A is regular. By performing a block LU decomposition on (10), this linear system is equivalent to the following two equations:

$$\left(D - CA^{-1}B\right)\mathbf{y} = \mathbf{g} - CA^{-1}\mathbf{f}, \tag{11}$$

$$\mathbf{x} = A^{-1}\mathbf{f} - A^{-1}B\mathbf{y}. \tag{12}$$

The matrix $\Sigma := D - CA^{-1}B \in \mathbb{R}^{N_1 \times N_1}$, $0 \leq N_1 \leq N$, is called the *Schur-complement* of A in the block matrix $\mathcal{A}$ and (11) is called the *Schur-complement equation* for y. The strategy to solve equations (11) and (12) is described in Algorithm 1.

Algorithm 1 Schur-complement solver

Let an initial solution $\xi_0 \in \mathbb{R}^N$, a right hand side vector $(f, g)^\top \in \mathbb{R}^N$, a system matrix $\mathcal{A} \in \mathbb{R}^{N \times N}$, a relative tolerance $\varepsilon_{rel} > 0$, an absolute tolerance $\varepsilon_{abs} > 0$, a maximum iteration number $I_{max} \in \mathbb{N}$ and preconditioning matrices $M_j^{-1} \in \mathbb{R}^{N_1 \times N_1}$, $j \in \mathbb{N}$ for the Schur-complement matrix Σ be given.

1. Solve Schur-complement equation (11) for y by FGMRES with Right Preconditioning [32, 33] and the given parameters ε_{rel}, ε_{abs}, I_{max} and M_j^{-1}.
2. Compute x via (12).

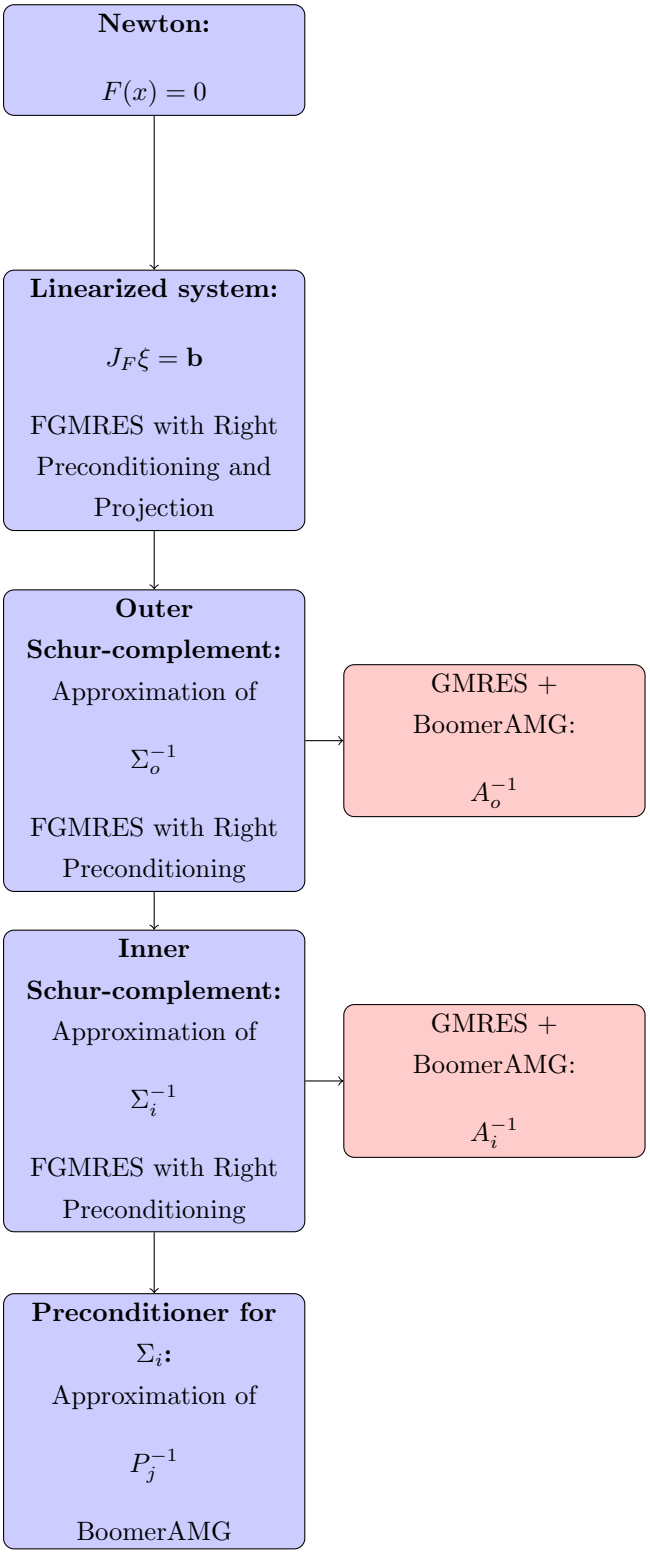

Figure 3: Structure of the proposed solver.

Remark 3.1

1. Observe, that in step 2 of Algorithm 1, in contrast to [19], the right preconditioned FMGRES algorithm *without* projection is used. This is possible due to the change in the preconditioner of the inner Schur-complement equation, see below. The change described there leads to the observation, that for the Schur-complement equations an additional projection does not yield an improvement in the convergence rate.
2. The matrix A^{-1} is approximated by a right preconditioned GMRES method, cf. [28, 33, 34] and Figure 3.

The Jacobian matrix J_F of the Low-Mach model can be written in variable-wise block matrix form as

$$J_F = \begin{pmatrix} A_{\mathbf{v},\mathbf{v}} & A_{\mathbf{v},\rho} & A_{\mathbf{v},\theta} & A_{\mathbf{v},p} \\ A_{\rho,\mathbf{v}} & A_{\rho,\rho} & A_{\rho,\theta} & A_{\rho,p} \\ A_{\theta,\mathbf{v}} & A_{\theta,\rho} & A_{\theta,\theta} & A_{\theta,p} \\ A_{p,\mathbf{v}} & A_{p,\rho} & A_{p,\theta} & A_{p,p} \end{pmatrix}.$$

$A_{i,j}$ corresponds to the matrix block, where the finite element test functions belong to variable i and the finite element trial functions to variable j.

The outer Schur-complement solver (Algorithm 1) operates on the Schur-complement decomposition of the Jacobian matrix J_F of the Low-Mach model and is used as preconditioner for the FMGRES algorithm in the computation of each Newton step, see Figure 3. Since this is the outer Schur-complement solver, all block matrices etc. corresponding to this instance are denoted by the subscript o. Especially, it holds $\mathcal{A}_o = J_F$.

The partition of the matrix $\mathcal{A}_o$ for the application of the outer Schur-complement can be written in block matrix form as

$$\mathcal{A}_o = \left(\begin{array}{cc|cc} A_{\rho,\rho} & A_{\rho,\theta} & A_{\rho,\mathbf{v}} & A_{\rho,p} \\ A_{\theta,\rho} & A_{\theta,\theta} & A_{\theta,\mathbf{v}} & A_{\theta,p} \\ \hline A_{\mathbf{v},\rho} & A_{\mathbf{v},\theta} & A_{\mathbf{v},\mathbf{v}} & A_{\mathbf{v},p} \\ A_{p,\rho} & A_{p,\theta} & A_{p,\mathbf{v}} & A_{p,p} \end{array}\right).$$

As described in Algorithm 1, the Schur-complement equation (11) with the system matrix

$$\Sigma_o := D_o - C_o A_o^{-1} B_o \tag{13}$$

is solved by the right preconditioned FGMRES algorithm. Therefore, efficient preconditioners $M_{j,o}^{-1}$ for Σ_o are needed, where j denotes the iteration counter, i.e., the preconditioner may change in every iteration.

As it is reasoned in [19], the matrix $M_{j,o}^{-1}$ with

$$M_{j,o}^{-1} \approx \mathcal{A}_i^{-1} \quad \text{and} \quad \mathcal{A}_i = D_o \tag{14}$$

constitutes an efficient preconditioner for Σ_o , where $\mathcal{A}_i^{-1}$ is again approximated by a Schur-complement solver (Algorithm 1), which is referred to as the inner Schur-complement and all corresponding block matrices etc. are denoted by the subscript i.

The partition of the matrix $\mathcal{A}_i$ for the application of the inner Schur-complement can be written in block matrix form as

$$\mathcal{A}_i = \left(\begin{array}{c|c} A_{\mathbf{v},\mathbf{v}} & A_{\mathbf{v},p} \\ \hline A_{p,\mathbf{v}} & A_{p,p} \end{array}\right).$$

As described in Algorithm 1, the Schur-complement equation (11) with the system matrix

$$\Sigma_i := D_i - C_i A_i^{-1} B_i \tag{15}$$

is solved by the right preconditioned FGMRES algorithm. Therefore, efficient preconditioners $M_{j,i}^{-1}$ for Σ_i are needed.

As it is reasoned in [19], the matrices

$$M_{j,i}^{-1} \approx P_j^{-1} \quad \text{with} \quad P_j = \delta_p \mathcal{M}_p + D_i \tag{16}$$

Parameter	ε_{rel}	ε_{abs}	I_{max}	η_0	η_{max}	f_d	λ_{min}	α
Value	10^{-6}	$5 \cdot 10^{-14} \cdot \lvert\Omega\rvert$	1000	10^{-4}	10^{-3}	$\frac{3-\sqrt{5}}{2}$	10^{-16}	10^{-6}

Table 1: Parameters for Newton's method with Eisenstat-Walker forcing and Armijo damping [19]. $|\Omega|$ denotes the volume of the computational domain Ω.

constitute efficient preconditioners for the Schur-complement Σ_i, where $\mathcal{M}_p$, defined as

$$\mathcal{M}_p := (M_{i,j})_{i,j=1}^{N_h}, \quad M_{i,j} := (\psi_j, \psi_i), \tag{17}$$

denotes the mass matrix of the hydrodynamic pressure variable p^* and

$$\delta_p := 10^{-16} \cdot \Delta t \tag{18}$$

denotes the regularization parameter. With this regularization, the matrices P_j are symmetric and positive definite and, therefore, the inversion is computed by the BoomerAMG preconditioner [8, 11, 13–15, 29, 37, 38], which is an implementation of Algebraic Multigrid (AMG) methods [1–5, 9, 10, 12–14, 18, 29, 35].

In [19], the Conjugate Gradient (CG) algorithm [21, 28, 33] preconditioned by BoomerAMG was proposed. With that combination, the convergence of the nested Schur-complements preconditioner benefits of the application of FMGRES with Right Preconditioning and Projection in Algorithm 1. The solely usage of BoomerAMG, as it is proposed here, drops the need to project onto the subspace, where p^* incorporates zero mean value, because it is observed that the convergence does no longer benefit of the additional projections due to the fact that AMG is, in contrast to CG, a defect correction algorithm. Therefore, as the parts in the right-hand sides of the linear systems in the Schur-complement equations, which correspond to the hydrodynamic pressure p^*, already fulfill the zero mean-value property, this property is maintained by AMG, whereas the CG algorithm may take projection steps outside this subspace. This observation is one of the key points for the achieved improvements in terms of compute time in comparison to [19].

The nested Schur-complement solver can be configured with a variety of parameters for the occurring (F)GMRES solvers and the corresponding preconditioners. A second key point for the achieved improvements in terms of compute time in comparison to [19] is found in further tuning the parameters, which control the iterations of FMGRES in the two Schur-complement solvers. The new choice of parameters, which is proposed here, is based on the following observation:

The FMGRES algorithms for both the outer and inner Schur-complement equations (11) exhibit a rapid convergence rate within the first iterations. The relative tolerance, down to which the convergence is rapid, is quite constant for most applications of the Schur-complement preconditioner, while the number of iterations to reach this tolerance varies slightly. In order to, on the one hand, achieve this observed tolerance and, on the other hand, keep the computational costs low in those cases, where this observed tolerance is not achieved within few iterations, the maximum number of iterations is chosen a little larger than the empirically observed average value at which the convergence rate deteriorates. At the same time, the relative tolerance is set to the observed relative tolerance, down to which the convergence rate is rapid in most applications of the Schur-complement preconditioner.

The full list of parameters, which are used for the computation of numerical results in Section 4, are given in Tables 1-9. Parameters, which differ from those in [19], are marked with red color. For a description of the meaning of the given values cf. [19] as well as the hypre Reference Manual [22].

4 Numerical Results

The Low-mach model and its respective numerical solver, which have been described in Sections 2 and 3, respectively, are implemented with the aid of the HiFlow^3 software package [20]. Furthermore, HiFlow^3 is compiled with support for the following third-party libraries: hypre 2.12.0 [15], METIS 5.1.0 [23], ParMETIS 4.0.3 [24] and HDF5 1.10.1 [36]. All these libraries as well as HiFlow^3 itself are compiled with

Parameter	ε_{rel}	ε_{abs}	I_{max}
Value	η_{i-1}	$5 \cdot 10^{-14} \cdot \lvert\Omega\rvert$	1000

Table 2: Parameters of FGMRES for the Low-Mach model. η_{i-1} denotes the current forcing term in Newton's method and $|\Omega|$ the volume of the computational domain Ω.

Parameter	ε_{rel}	ε_{abs}	I_{max}
Value	$5 \cdot 10^{-2}$	0.0	10

Table 3: Parameters of FGMRES for the solution of the outer Schur-complement equation.

Parameter	ε_{rel}	ε_{abs}	I_{max}
Value	$5 \cdot 10^{-3}$	0.0	500

Table 4: Parameters of GMRES for the inversion of the matrix A_o in the outer Schur-complement solver (Algorithm 1).

Parameter	**Value**
CoarsenType	10
NumFunctions	2
RelaxType	3
RelaxWt	0.25
InterpType	4
AggNumLevels	25
MaxIter	1
Tol	0.0
StrongThreshold	0.6

Table 5: Parameters of BoomerAMG for preconditioning GMRES in the inversion of the matrix A_o in the outer Schur-complement solver (Algorithm 1).

Parameter	ε_{rel}	ε_{abs}	I_{max}
Value	10^{-1}	0.0	10

Table 6: Parameters of FGMRES for the solution of the inner Schur-complement.

Parameter	ε_{rel}	ε_{abs}	I_{max}
Value	$5 \cdot 10^{-4}$	0.0	500

Table 7: Parameters of GMRES for the inversion of the matrix A_i in the inner Schur-complement solver (Algorithm 1).

Parameter	Value
CoarsenType	10
NumFunctions	3
RelaxType	3
RelaxWt	0.5
InterpType	6
AggNumLevels	25
MaxIter	1
Tol	0.0
StrongThreshold	0.6

Table 8: Parameters of BoomerAMG for preconditioning GMRES in the inversion of the matrix A_i in the inner Schur-complement solver (Algorithm 1).

Parameter	Value
CycleType	2
CoarsenType	10
NumFunctions	1
RelaxType	6
NumSweeps	3
RelaxWt	0.5
InterpType	6
AggNumLevels	25
MaxIter	1
Tol	0.0
StrongThreshold	0.6

Table 9: Parameters of BoomerAMG for the inversion of the matrix P_j (16) in the inner Schur-complement solver (Algorithm 1).

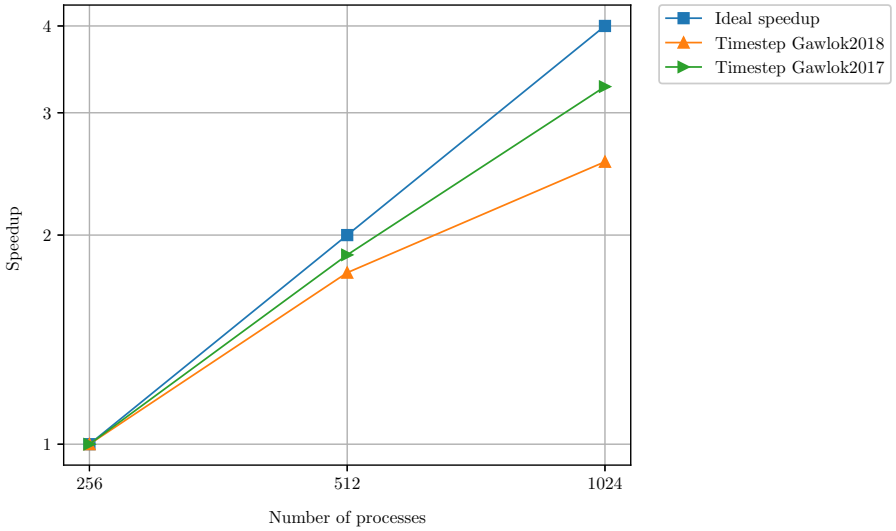

Figure 4: Speedup in strong scaling test for whole time-step on bwForCluster MLS & WISO (Production) relative to 256 processes.

Intel(R) compiler suite version 16.0.4 and Intel Message Passing Interface (mpicc, mpicxx, mpif77 and mpif90) version '5.1.3.258'.

The numerical solution of the cyclone-cylcone interaction problem of Section 2 with the described discretization yields a discrete problem with 31,680,000 *unknowns* or *degrees of freedom (DoFs)* in each time-step.

In the following, the performance in terms scalability and compute time of the solver proposed in Section 3 is investigated and compared to the results obtained in [19]. Therefore, results obtained with the new solver are labeled as Gawlok2018, whereas the results of [19] are labeled as Gawlok2017.

Figures 4 and 5 show the parallel scalability and efficiency of solving a whole time-step, respectively, of the compared solvers with respect to a baseline of 256 MPI processes. In order to filter out variations in runtime, the compute times of the first ten time-steps are accumulated for the presented results. Clearly, the new solver scales worse than the one proposed in [19], but can still maintain an efficiency of more than 65% with 1,024 active MPI processes. The differences are purely due to the changes in the numerical solver, because the assembly of matrices and vectors shows identical and almost perfect scaling behavior, cf. Figures 6 and 7, respectively.

All following results are computed using 1,024 MPI processes.

But when it comes to compute time, the new solver significantly outperforms the one proposed in [19], see Figure 8. Figure 8 shows the accumulated compute times and the development of their ratio plotted over simulated physical time (left plot) as well as the ratio at the final time of $T = 96h$ (right plot). The solver proposed in this article achieves to finish the simulation *2.1 times* faster than the one of [19]. Considering the plot of the ratio over time, the ratio increases significantly up to $30h$ of simulated physical time and in the following stays virtually constant. The reason for this behavior can be found if one inspects the numbers in Table 10. Considering the number of FGMRES iterations per time-step, which are needed to invert the matrices J_F, the new solver needs exactly four iterations per time-step, cf. Figure 10, whereas as the solver of [19] needs in the beginning two and later three iterations, cf. Figure 9. Furthermore, it can be clearly seen that the differences in compute time are due to the improved solver because the times for assembling matrices and vectors, respectively, do not differ virtually. Therefore, the newly proposed nested Schur-complement preconditioner exhibits a significantly more robust convergence behavior in the course of the dynamical evolution of the interacting cyclones. Especially, one application of the improved preconditioner is much cheaper in terms of computational time.

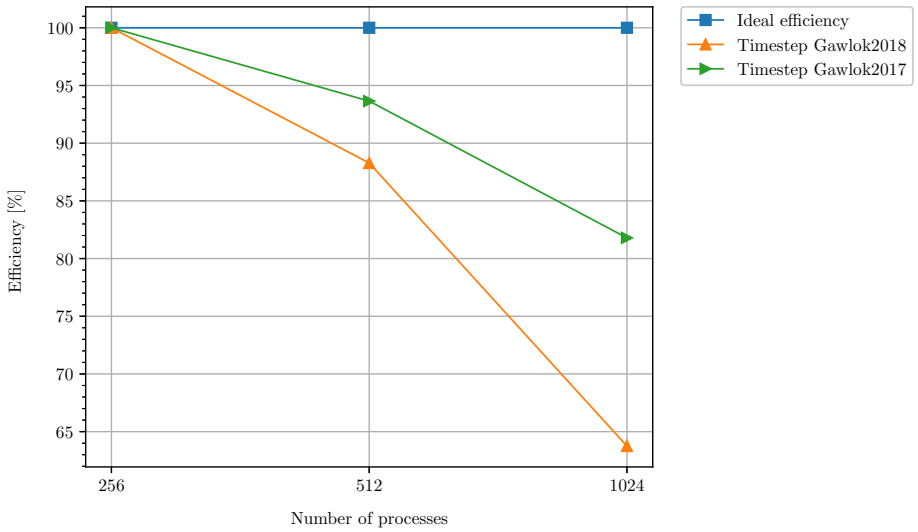

Figure 5: Efficiency in strong scaling test for whole time-step on bwForCluster MLS & WISO (Production) relative to 256 processes.

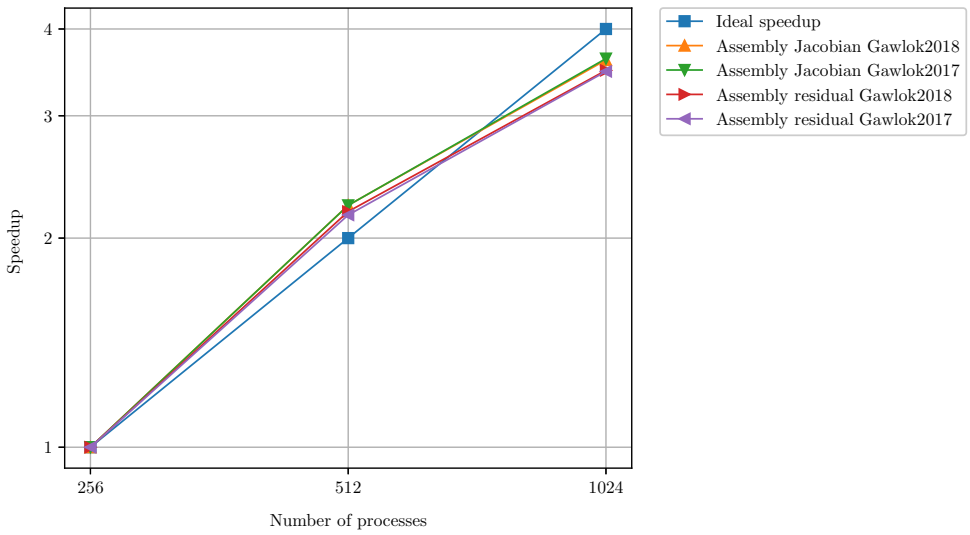

Figure 6: Speedup in strong scaling test for assembly on bwForCluster MLS & WISO (Production) relative to 256 processes.

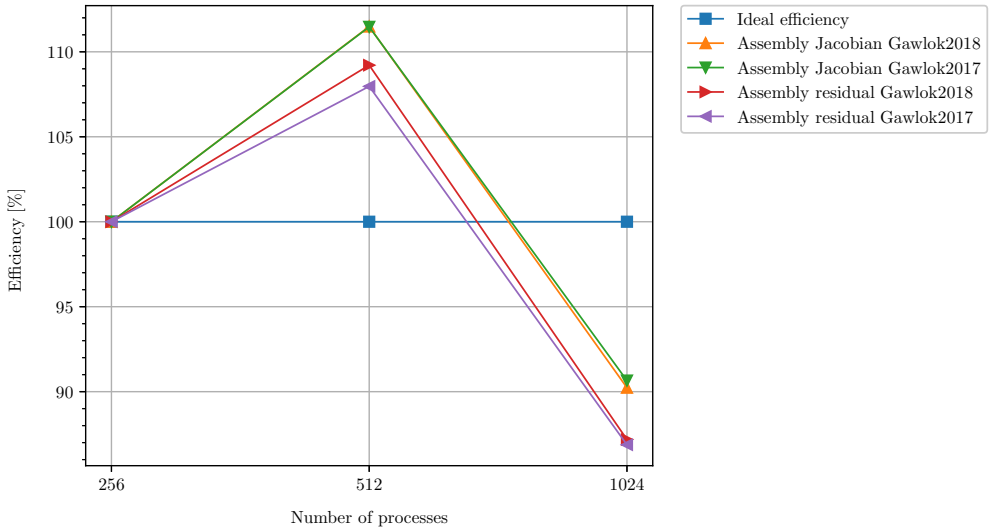

Figure 7: Efficiency in strong scaling test for assembly on bwForCluster MLS & WISO (Production) relative to 256 processes.

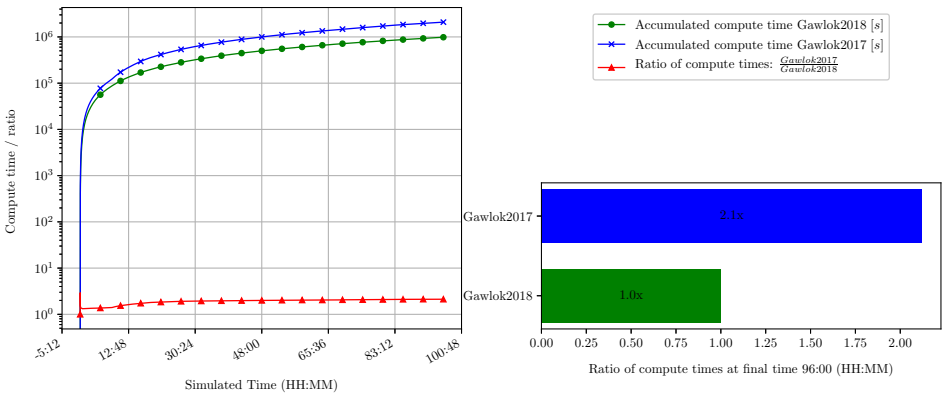

Figure 8: Comparison of the compute times on bwForCluster MLS & WISO (Production).

Model	**Newton iterations**	**Compute time** $[s]$	**Assembly time residual** $[s]$	**Assembly time Jacobian** $[s]$	**FGMRES iterations**
Gawlok2018	2.00014	28.493	1.738	4.587	4.0
Gawlok2017	2.00003	60.428	1.703	4.416	2.914
$\frac{Gawlok2017}{Gawlok2018}$	0.99994	2.121	0.98	0.963	0.729

Table 10: Comparison of averages of quantities, which measure the runtime performance, for one time-step on bwForCluster MLS & WISO (Production).

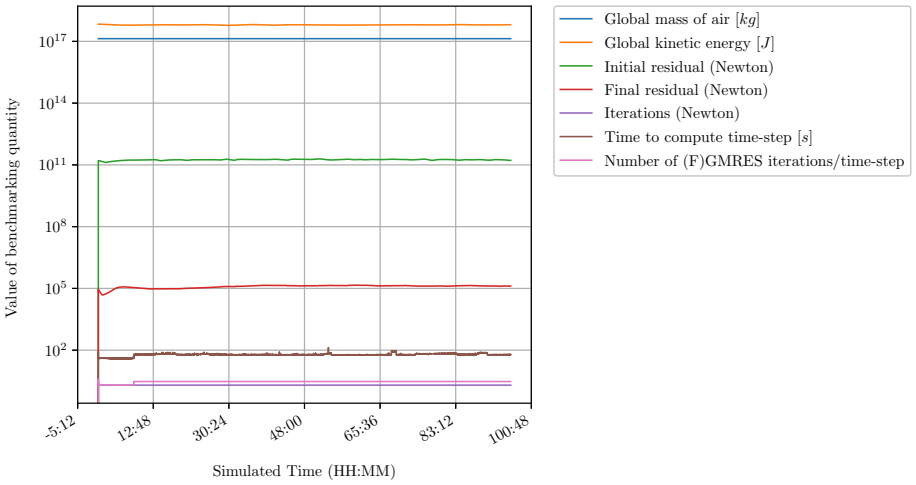

Figure 9: Benchmarking quantities of [19] on bwForCluster MLS & WISO (Production).

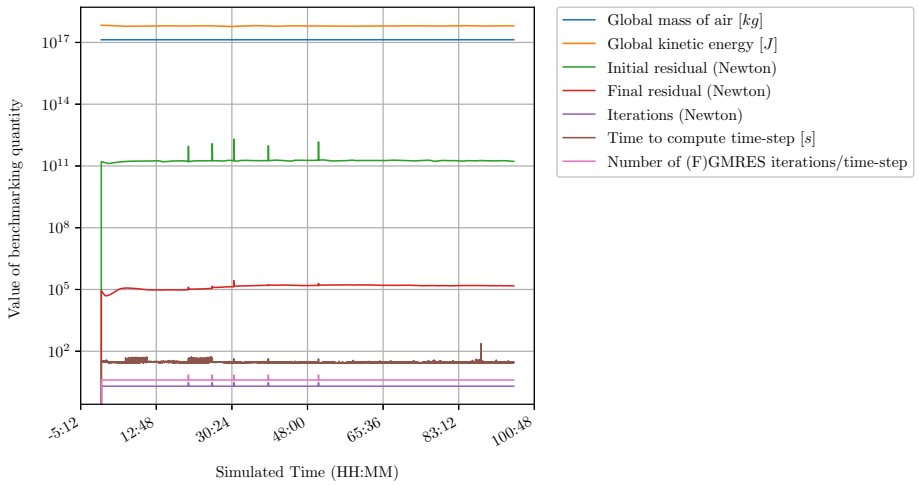

Figure 10: Benchmarking quantities of the new solver on bwForCluster MLS & WISO (Production).

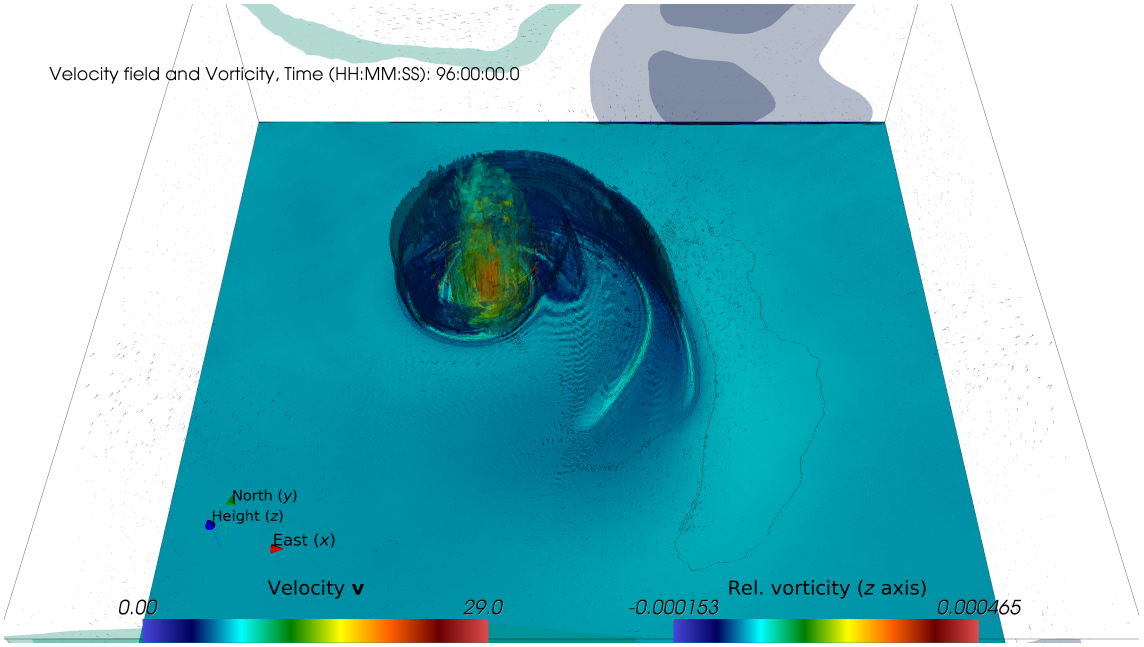

Figure 11: Final velocity field and vertical vorticity component at $T = 96\ h$.

5 Summary and Outlook

A formulation of a Low-mach model has been presented along with the description of a benchmark scenario of two interacting tropical cyclones. For the discrete nonlinear equations of this model, an improved variant of the nested Schur-complement preconditioner has been proposed. A detailed discussion of the numerical results has been conducted with a special emphasis on computational costs and scalability. It has been demonstrated that the new preconditioner clearly outperforms a prior version of this kind of preconditioner with respect to computing time by a factor of two, but for the sake of a slightly reduced scalability.

With the obtained results, the presented work constitutes a major step forward in improving the computational costs of Low-mach flows with the presented numerical model in cyclone-cyclone interaction scenarios. In [19] this has been proposed as one of the major goals for improvement in order to facilitate further studies, especially with respect to computations on finer computational grids, within feasible computational time spans.

Based on the presented research, further efforts need to be taken to improve the scalability. Especially, in the context of larger-scale simulations with finer computational grids and, as a results, higher numbers of MPI processes, the scaling properties of the applied solver are crucial. Furthermore, the current implementation of the presented preconditioner in HiFlow^3, which has been used in this study, is not optimal in the sense that explicit copies of the needed submatrices are extracted from the large Jacobian matrix J_F. On the one hand this significantly increases the memory usage and the demand on the available computer resources and, on the other hand, the copy processes needs computational time. Therefore, refactoring the existing implementation by means of block-matrices and block-vectors promises to be a remedy to both problems.

Acknowledgements

The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1134-1 FUGG. The authors gratefully acknowledge the data storage service SDS@hd supported by the Ministry of Science, Research and the Arts Baden-Württemberg (MWK) and the German Research Foundation (DFG) through grant INST 35/1314-1 FUGG.

References

[1] Allison H. Baker, Martin Schulz, and Ulrike M. Yang. "On the performance of an algebraic multigrid solver on multicore clusters". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 6449 LNCS. 2011, pp. 102–115. ISBN: 9783642193279. DOI: 10.1007/978-3-642-19328-6_12.

[2] Allison H. Baker et al. "Challenges of scaling algebraic multigrid across modern multicore architectures". In: *Proceedings - 25th IEEE International Parallel and Distributed Processing Symposium, IPDPS 2011*. Anchorage, AK, USA: Conference Proceedings, IEEE, 2011, pp. 275–286. ISBN: 9780769543857. DOI: 10.1109/IPDPS.2011.35.

[3] Allison H. Baker et al. "Multigrid Smoothers for Ultraparallel Computing". In: *SIAM Journal on Scientific Computing* 33 (2011), pp. 2864–2887. ISSN: 1064-8275. DOI: 10.1137/100798806.

[4] Allison H. Baker et al. “Scaling Algebraic Multigrid Solvers: On the Road to Exascale”. In: *Competence in High Performance Computing 2010*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 215–226. ISBN: 978-3-642-24024-9. DOI: 10.1007/978-3-642-24025-6_18. URL: http://www.springerlink.com/content/v63j3355468k0471/%20http://link.springer.com/10.1007/978-3-642-24025-6%7B%5C_%7D18.

[5] Allison H. Baker et al. “Scaling hypre’s multigrid solvers to 100,000 cores”. In: *High Performance Scientific Computing: Algorithms and Applications*. Ed. by Michael W. Berry et al. Springer, 2012. ISBN: 1447124367 9781447124368.

[6] Martin Baumann et al. “Goal-oriented adaptivity for idealised tropical cyclones: A binary interaction scenario”. In: *Meteorologische Zeitschrift* 24.3 (Apr. 2015), pp. 269–292. ISSN: 0941-2948. DOI: 10.1127/metz/2015/0591. URL: http://www.schweizerbart.de/papers/metz/detail/24/84720/Goal%7B%5C_%7Doriented%7B%5C_%7Dadaptivity%7B%5C_%7Dfor%7B%5C_%7Didealised%7B%5C_%7Dtropical%7B%5C_%7Dcy?af=crossref.

[7] Malte Braack. *Finite Elemente*. Lecture notes. Mathematisches Seminar, Christian-Albrechts-Universität zu Kiel, 2015.

[8] Edmond Chow, Andrew J. Cleary, and Robert D. Falgout. “Design of the hypre Preconditioner Library”. In: *In SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing*. Ed. by Mike Henderson, Chris Anderson, and Steve Lyons. SIAM, 1998, pp. 106–116.

[9] Edmond Chow et al. “A Survey of Parallelization Techniques for Multigrid Solvers”. In: *Parallel Processing for Scientific Computing*. Ed. by M. A. Heroux, P. Raghavan, and H. D. Simon. SIAM Series on Software, Environments, and Tools, 2006. Chap. 10, pp. 179–195. ISBN: 9780898716191. URL: http://books.google.com/books?hl=en%7B%5C&%7Dlr=%7B%5C&%7Did=AWpK2nkpifcC%7B%5C&%7Doi=fnd%7B%5C&%7Dpg=PA179%7B%5C&%7Ddq=A+Survey+of+Parallelization+Techniques+for+Multigrid+Solvers%7B%5C&%7Dots=oFmixPekds%7B%5C&%7Dsig=XLZ-P0p8q97Tp8JeggZT3rbqd8A.

[10] Andrew J. Cleary et al. “Robustness and scalability of algebraic multigrid”. In: *SIAM J. Sci. Comput.* 21.UCRL-JC-130718 (2000), pp. 1886–1908.

[11] Robert D. Falgout. “A Note on the Relationship Between Adaptive AMG and PCG”. In: *LLNL Technical Report UCRL-TR-205838* (2004).

[12] Robert D. Falgout. *An introduction to algebraic multigrid*. 2006. DOI: 10.1109/MCSE.2006.105.

[13] Robert D. Falgout and Jim E. Jones. “Multigrid on Massively Parallel Architectures”. In: *Multigrid Methods*. Ed. by Erik Dick, Kris Riemslagh, and Jan Vierendeels. VI. Vol. 14. Springer-Verlag, 2000, pp. 101–107. ISBN: 978-3-540-67157-2.

[14] Robert D. Falgout and Panayot S. Vassilevski. “On Generalizing the Algebraic Multigrid Framework”. In: *SIAM Journal on Numerical Analysis* 42.4 (Jan. 2004), pp. 1669–1693. ISSN: 0036-1429. DOI: 10.1137/S0036142903429742. URL: http://epubs.siam.org/doi/abs/10.1137/S0036142903429742.

[15] Robert D. Falgout and Ulrike Meier Yang. "hypre: A Library of High Performance Preconditioners". In: *Lecture Notes in Computer Science*. Ed. by Peter M. A. Sloot et al. Vol. 2331. Springer Berlin Heidelberg, 2002, pp. 632–641. ISBN: 978-3-540-43594-5. DOI: 10.1007/3-540-47789-6_66. URL: http://link.springer.com/10.1007/3-540-47789-6%20http://link.springer.com/chapter/10.1007/3-540-47789-6%7B%5C_%7D10%20http://link.springer.com/10.1007/3-540-47789-6%7B%5C_%7D66.

[16] S. Fujiwhara. "On the growth and decay of vortical systems". In: *Quarterly Journal of the Royal Meteorological Society* 49.206 (Aug. 2007), pp. 75–104. ISSN: 00359009. DOI: 10.1002/qj.49704920602. URL: http://doi.wiley.com/10.1002/qj.49704920602.

[17] S. Fujiwhara. "The natural tendency towards symmetry of motion and its application as a principle in meteorology". In: *Quarterly Journal of the Royal Meteorological Society* 47.200 (Aug. 2007), pp. 287–292. ISSN: 00359009. DOI: 10.1002/qj.49704720010. URL: http://doi.wiley.com/10.1002/qj.49704720010.

[18] Hormozd Gahvari et al. "Systematic Reduction of Data Movement in Algebraic Multigrid Solvers". In: *Proceedings of the 2013 IEEE 27th International Symposium on Parallel & Distributed Processing Workshops and PhD Forum (IPDPSW 2013)* LLNL-CONF-587832 (2013).

[19] Simon Gawlok. "Numerical Methods for Compressible Flow with Meteorological Applications". PhD thesis. Ruprecht-Karls-Universität Heidelberg, 2017. DOI: 10.11588/heidok.00023532. URL: http://www.ub.uni-heidelberg.de/archiv/23532.

[20] Simon Gawlok et al. "HiFlow3 – Technical Report on Release 2.0". In: *Preprint Series of the Engineering Mathematics and Computing Lab* 0.06 (2017). ISSN: 2191-0693. DOI: 10.11588/emclpp.2017.06.42879. URL: https://journals.ub.uni-heidelberg.de/index.php/emclpp/article/view/42879.

[21] Magnus Rudolph Hestenes and Eduard Stiefel. "Methods of conjugate gradients for solving linear systems". In: *Journal of Research of the Natural Bureau of Standards (NBS)* 49.6 (1952), pp. 409–436. ISSN: 0091-0635. DOI: 10.6028/jres.049.044. arXiv: 1102.0183.

[22] *hypre Reference Manual*. URL: https://github.com/LLNL/hypre/blob/master/docs/HYPRE_ref_manual.pdf.

[23] George Karypis and Vipin Kumar. "A Fast and Highly Quality Multilevel Scheme for Partitioning Irregular Graphs". In: *SIAM Journal on Scientific Computing* 20.1 (1999), pp. 359–392.

[24] George Karypis and Vipin Kumar. "Parallel Multilevel k-way Partitioning Scheme for Irregular Graphs". In: *SIAM Review* 41.2 (1999), pp. 278–300.

[25] Bernhard Müller. *Computation of Compressible Low Mach Number Flow*. 1996.

[26] Bernhard Müller. "Low-Mach-Number Asymptotics of the Navier-Stokes Equations". In: *Journal of Engineering Mathematics* 34.1/2 (1998), pp. 97–109. ISSN: 00220833. DOI: 10.1023/A:1004349817404. URL: http://link.springer.com/10.1023/A:1004349817404.

[27] Andrew Majda. *Compressible Fluid Flow and Systems of Conservation Laws in Several Space Variables*. Vol. 53. Applied Mathematical Sciences. Springer New York, 1984. ISBN: 978-0-387-96037-1. DOI: 10.1007/978-1-4612-1116-7. URL: http://link.springer.com/10.1007/978-1-4612-1116-7.

[28] Andreas Meister. *Numerik linearer Gleichungssysteme*. 5th ed. Wiesbaden: Springer Fachmedien Wiesbaden, 2015. ISBN: 978-3-658-07199-8. DOI: 10.1007/978-3-658-07200-1. URL: http://link.springer.com/10.1007/978-3-658-07200-1.

[29] Bram Metsch. "Algebraic multigrid (AMG) for saddle point systems". PhD thesis. Rheinische Friedrich-Wilhelms-Universität Bonn, 2013. URL: http://hss.ulb.uni-bonn.de/2013/3347/3347.pdf.

[30] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. 2nd ed. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006. ISBN: 978-0-387-30303-1. DOI: 10.1007/978-0-387-40065-5. arXiv: NIHMS150003. URL: http://books.google.com/books?hl=en%7B%5C&%7Dlr=%7B%5C&%7Did=VbHYoSyelFcC%7B%5C&%7Doi=fnd%7B%5C&%7Dpg=PR17%7B%5C&%7Ddq=Numerical+Optimization%7B%5C&%7Dots=30P7Bqy8QP%7B%5C&%7Dsig=Kaa0LaXAd1L2qO5Mdgn96umVW-8%20http://link.springer.com/10.1007/BF01068601%20http://link.springer.com/10.1007/978-0-387-40065-5.

[31] Andrea D. F. Richter. "Untersuchungen zur Wechselwirkung tropischer Wirbelstürme mit Hilfe eines idealisierten dreidimensionalen numerischen Modells". Diplomarbeit. Karlsruher Institut für Technologie (KIT), 2012.

[32] Youcef Saad. "A Flexible Inner-Outer Preconditioned GMRES Algorithm". In: *SIAM Journal on Scientific Computing* 14.2 (Mar. 1993), pp. 461–469. ISSN: 1064-8275. DOI: 10.1137/0914028. URL: http://epubs.siam.org/doi/abs/10.1137/0914028.

[33] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. 2nd ed. Society for Industrial and Applied Mathematics, 2003. ISBN: 0898715342. DOI: 10.2113/gsjfr.6.1.30. arXiv: 0806.3802. URL: http://www.stanford.edu/class/cme324/saad.pdf%20http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1231631%7B%5C%%7D5Cnhttp://www.stanford.edu/class/cme324/saad.pdf.

[34] Yousef Saad and Martin H. Schultz. "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems". In: *SIAM Journal on Scientific and Statistical Computing* 7.3 (1986), pp. 856–869. ISSN: 0196-5204. DOI: 10.1137/0907058.

[35] K. Stüben. "Algebraic multigrid (AMG): an introduction with applications". In: *GMD Report* 70 (1999), pp. 1–127. URL: http://en.scientificcommons.org/20203294.

[36] The HDF Group. *Hierarchical Data Format, version 5*. http://www.hdfgroup.org/HDF5/. 1997-2017.

[37] Ulrike Meier Yang. "On the use of relaxation parameters in hybrid smoothers". In: *Numerical Linear Algebra with Applications* 11.2-3 (2004), pp. 155–172. ISSN: 10705325. DOI: 10.1002/nla.375.

[38] Ulrike Meier Yang. *Numerical Solution of Partial Differential Equations on Parallel Computers*. Ed. by Are Magnus Bruaset and Aslak Tveito. Vol. 51. Lecture Notes in Computational Science and Engineering. Berlin/Heidelberg: Springer-Verlag, 2006, pp. 209–236. ISBN: 3-540-29076-1. DOI: 10.1007/3-540-31619-1. arXiv: arXiv:1011.1669v3. URL: http://link.springer.com/10.1007/3-540-31619-1.

Preprint Series of the Engineering Mathematics and Computing Lab

recent issues

No. 2018-01 David John, Michael Schick, Vincent Heuveline: Learning model discrepancy of an electric motor with Bayesian inference

No. 2017-07 Martin Baumann, Fabian Gebhart, Oliver Mattes, Sotirios Nikas, Vincent Heuveline: Development and implementation of a temperature monitoring system for HPC systems

No. 2017-06 Simon Gawlok, Philipp Gerstner, Saskia Haupt, Vincent Heuveline, Jonas Kratzke, Philipp Lösel, Katrin Mang, Mareike Schmidtobreick, Nicolai Schoch, Nils Schween, Jonathan Schwegler, Chen Song, Martin Wlotzka: HiFlow^3 – Technical Report on Release 2.0

No. 2017-05 Nicolai Schoch, Vincent Heuveline: Towards an Intelligent Framework for Personalized Simulation-enhanced Surgery Assistance: Linking a Simulation Ontology to a Reinforcement Learning Algorithm for Calibration of Numerical Simulations

No. 2017-04 Martin Wlotzka, Thierry Morel, Andrea Piacentini, Vincent Heuveline: New features for advanced dynamic parallel communication routines in OpenPALM: Algorithms and documentation

No. 2017-03 Martin Wlotzka, Vincent Heuveline: An energy-efficient parallel multigrid method for multi-core CPU platforms and HPC clusters

No. 2017-02 Thomas Loderer, Vincent Heuveline: New sparsing approach for real-time simulations of stiff models on electronic control units

No. 2017-01 Chen Song, Markus Stoll, Kristina Giske, Rolf Bendl, Vincent Heuveline: Sparse Grids for quantifying motion uncertainties in biomechanical models of radiotherapy patients

No. 2016-02 Jonas Kratzke, Vincent Heuveline: An analytically solvable benchmark problem for fluid-structure interaction with uncertain parameters

No. 2016-01 Philipp Gerstner, Michael Schick, Vincent Heuveline, Nico Meyer-Hübner, Michael Suriyah, Thomas Leibfried, Viktor Slednev, Wolf Fichtner, Valentin Bertsch: A Domain Decomposition Approach for Solving Dynamic Optimal Power Flow Problems in Parallel with Application to the German Transmission Grid

No. 2015-04 Philipp Gerstner, Vincent Heuveline, Michael Schick : A Multilevel Domain Decomposition approach for solving time constrained Optimal Power Flow problems

No. 2015-03 Martin Wlotzka, Vincent Heuveline: Block-asynchronous and Jacobi smoothers for a multigrid solver on GPU-accelerated HPC clusters

No. 2015-02 Nicolai Schoch, Fabian Kißler, Markus Stoll, Sandy Engelhardt, Raffaele de Simone, Ivo Wolf, Rolf Bendl, Vincent Heuveline: Comprehensive Pre- & Post-Processing for Numerical Simulations in Cardiac Surgery Assistance

No. 2015-01 Teresa Beck, Martin Baumann, Leonhard Scheck, Vincent Heuveline, Sarah Jones: Comparison of mesh-adaptation criteria for an idealized tropical cyclone problem

No. 2014-02 Christoph Paulus, Stefan Suwelack, Nicolai Schoch, Stefanie Speidel, Rüdiger Dillmann, Vincent Heuveline: Simulation of Complex Cuts in Soft Tissue with the Extended Finite Element Method (X-FEM)

Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)

www.emcl.iwr.uni-heidelberg.de