Final summary of the CEL-ACCEL Priority Project

Zbigniew Piotrowski

September 9, 2020

Overview of CEL-ACCEL Priority Project

- Overarching goal: provide GPU-ready version of the EULAG dynamics within COSMO
- Realization: March 2017 - March 2020 (including extension)
- FTEs: planned - 3.65, realized in full. This was severely underestimated.
- Fully externally funded by the research grant of Foundation for Polish Science, complementary to the state-funded efforts of CELO Priority Project at IMGW.

Scientific and practical questions addressed

- Would it be feasible to compute COSMO-EULAG on accelerators using COSMO+Gridtools ?
- Is the Gridtools adequate for the specifics of EULAG numerics (prevailing symmetric 3D stencil operators, notorious evaluation of non-trivial boundary conditions, Krylov solver)
- What is the real difference between the (CPU) performance of Gridtools binary and the reference Fortran code optimized such that it remains clean and readable ?
- What is the proper way of encoding EULAG boundary conditions on GPU, to comply with the sequentiality of operations in vertical ?
- Is it really so that the single source code of dynamics for multiple architectures is necessary ?

CEL-ACCEL Task 1 - Accelerator port of COSMO-EULAG

What has been done ?

- Refactoring and full optimization of EULAG diffusion operator to modern Fortran(in connection with EX-CELO Priority Project), benchmarking of computational performance.(Note that diffusion operators are the toughest in terms of clear programming, due to computations on both: A and C grids.)
- Refactoring and continuation of earlier optimization efforts for advection and implicit solver part.
- Complete rewrite of the memory management structure with no locally allocated matrices.
- Clear separation of advection, implicit solver, diffusion and auxiliary code and equipping the first three with idealized tests to ensure correctness, as well as performance assessment tools.
- Establishing Git repo with continuous integration.
- Major investment in new CMake module to manage the Fortran and C++ environment, together with the auxiliary software e.g. Serialbox

What turned out different from the Runge-Kutta porting effort ?

- It appeared necessary to introduce mechanism of line by line (or step by step) testing of the conformance between Fortran and GridTools code, because it may happen that the difference at one step luckily? disappears at the next step, yet the code remains buggy. This is a major infrastucture difference with lots of technical consequences.
- The EULAG numerics constantly evaluates horizontal boundary conditions, either via specialized stencils, or solution of 1D, 2D and 3D linear systems.
- EULAG numerics relies on the iteration over 4D matrices (Krylov subspaces) and global reductions, by design not available in GridTools. Workarounds are ugly.
- Fortran by default relies on the employment of quasi-2D xz and yz matrices, which are not easily supported by GridTools ecosystem.
- The EULAG's implicit solver unit relied on large number of matrices used at once, which for a long time exceeded technical limits of C++ libraries.

What turned out different from the Runge-Kutta porting effort ?

- The support from MeteoSwiss was great, but it is still very different from having GridTools developers in-house.
- Official COSMO 5.07 release with GridTools is dated 21.02.2020, which is five months AFTER the end of the (unextended) CEL-ACCEL. This means shooting at the moving target and dealing with API changes, with all the difficulties resulting from being an external collaborator.
- Rapid COSMO phase-out made the external effort of porting COSMO-EULAG to GridTools unfeasible, turning the focus to the scientific side of the project.
- The dependence on software engineering developments made everything intermittent, and in the final stage there was no human resources to assist myself in the GridTools developments, so I had to split time between CEL-ACCEL and EX-CELO.

Fortran performance improvements

Algorithm	1st order UPWIND	MPDATA standard	MPDATA gauge
32x32x96 grid	1.58	1.46	2.33
16x16x128 grid	1.36	1.27	2.01

Algorithm	1D vertical diffusion	2D vertical diffusion	3D diffusion
32x32x96 grid	1.25	7.06	5.99
16x16x128 grid	1.13	2.71	2.61

Figure: Advection and diffusion performance improvements per core.

Status of GridTools implementation

- All dynamical cores were ported to GridTools and serialized on CPU.
- Most of the stencils serialize on GPU, some of them were not ported to the new API yet.
- The full timeloop on Gridtools has not been run yet, mostly due to the technical obstacles and lack of FTEs in the recent period.
- Reference Fortran application is ready for testing of the full timeloop with GridTools (turned to be a major effort not foreseen before).

Summary and conclusions

- The project success is limited to the complete rewrite and optimization of Fortran core (facilitating further software engineering endeavours), establishing GridTools ecosystem and porting all stencils to GridTools that serialize on GPU. GPU-ready timeloop is not yet composed, as well as the Fortran - C++ interface. The remaining scientific goal is to check the performance of the GridTools implemenation in the meaningfull 3D idealized experiment of convection over heated plane, outside of COSMO framework.
- Notable speedup of Fortran code is achievable, similarly to the Runge-Kutta efforts. This will soon directly benefit COSMO-EULAG.
- The dependence on the software engineering issues is orthogonal to the numerical development of the dynamical cores. The latter should be possible only if complete GridTools ecosystem or its Python frontends will reach maturity, if ever.

Summary and conclusions

- We are not fully happy with GridTools, but as far as I know from personal communications, none of the existing solutions for the single source code on multiple architectures to date seem to find enthusiasm among the domain scientists in various research groups. So the brave attempts to explore the GridTools route are in my opinion fully justified, even if the reward is not immediate.
- I would like to express my deepest gratitude to the MeteoSwiss for their cooperation and support.