Consortium
for
Small-Scale Modelling

Technical Report No. 41

Performance On Massively Parallel Architectures (POMPA): Final report

February 2020

DOI: 10.5676/DWD_pub/nwv/cosmo-tr_41

Deutscher Wetterdienst
MeteoSwiss
Ufficio Generale Spazio Aereo e Meteorologia
ΕΘΝΙΚΗ ΜΕΤΕΩΡΟΛΟΓΙΚΗ ΥΠΗΡΕΣΙΑ
Instytucie Meteorogii i Gospodarki Wodnej
Administratia Nationala de Meteorologie
ROSHYDROMET
Agenzia Regionale Protezione Ambiente Piemonte
Agenzia Regionale Prevenzione Ambiente Energia Emilia Romagna
Centro Italiano Ricerche Aerospaziali
Amt für GeoInformationswesen der Bundeswehr
Israel Meteorological Service

www.cosmo-model.org

Editor: Massimo Milelli, ARPA Piemonte

Performance On Massively Parallel Architectures (POMPA): Final report

Project leaders:

X. Lapillonne[1], O. Fuhrer[1‡]

[1] MeteoSwiss

‡ from 2010 to 2015

Contents

Executive summary

Efficient use of high-performance computing (HPC) systems is key to enable higher resolution, larger computational domains, more complex physical parametrizations, or more ensemble members. Due to the end of Moore's Law, HPC systems achieve higher performance via a massive increase in processor cores accompanied by a stagnant or decreasing clock frequency and by employing specialized hardware such as GPUs. The main outcomes of this project are a GPU-capable version of the COSMO model and the possibility to run the model in single precision. Both developments have been merged into the official release of the model and are already in use by several members of the COSMO community. The port to GPU was achieved by using a domain-specific language (DSL) for the dynamics and OpenACC compiler directives for the other components of the model. The resulting code is about a factor 3-4x faster on current state-of-the-art GPUs as compare to traditional multi-core CPUs.

1 Introduction

Numerical weather prediction can greatly benefit from progress in computational technology by increasing resolution, model complexity or the number of ensemble members in probabilistic prediction system. In the recent years, performance increase in hardware stems from massive increase in the number of processing units (cores) and specialized hardware such as Graphics Processing Units (GPUs). In order to benefit from current hardware, scientific models need to be adapted. The aim of the POMPA project was to prepare the COSMO model for these hardware architectures and render it performance portable across architectures. The project leveraged efforts from the High Performance and High Productivity Computing (HP2C) initiative and the Platform for Advanced Scientific Computing (PASC) in Switzerland which addressed these challenges.

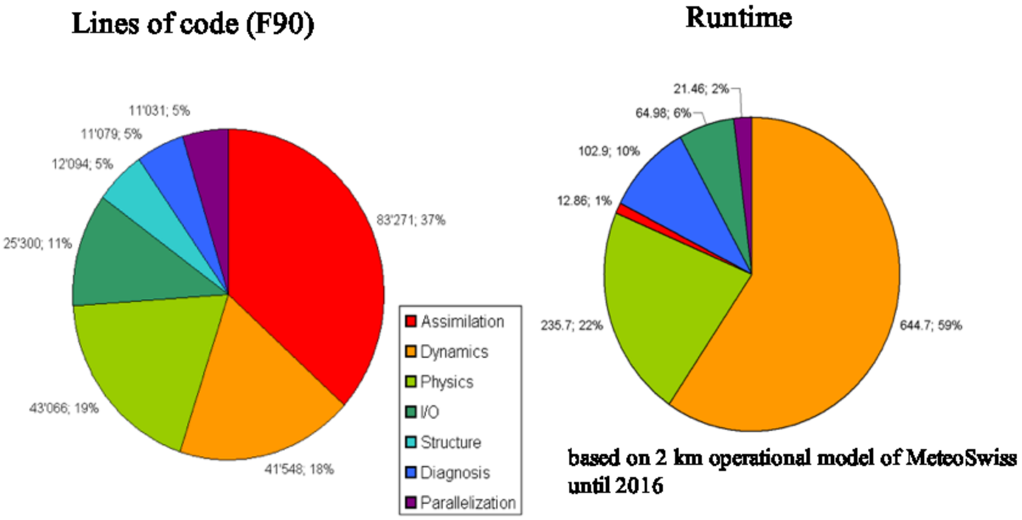
The following actions were planned in the project:

- Holistic performance analysis and documentation (Task 1)
- Improve generic scalability of the current model implementation:
  - Redesign memory layout and data structures for better cache awareness and easier load-balancing (Task 2)
  - Optimization of current parallelization and introduction of a hybrid parallelization employing MPI and OpenMP (Task 3)
  - Improve performance of I/O (Task 4)
  - Develop a switchable singe/double precision model version (Task 8)
- Explore emerging technologies:
  - Design a new implementation of the dynamical core optimized for multi-core processors and GPUs (Task 5)
  - Re-engineering the implementation of the physical parametrizations (Task 6)
  - Develop a GPU-capable version of COSMO (Task 7)

This report provides only a general overview of the main outcomes of the POMPA project, while further details can be found in several papers and reports published during this project which are referenced thereafter in particular. In Sec. 2 we discuss main achievements of the projects associated with the different tasks. In Sec. 3 we present the lessons learned during this work and finally some overview and recommendation for future work are discussed.

2 Main achievements

The performance analysis of the model (Task 1) shows that for a typical application most of the computational time (60%) is spend in the dynamical core, which only represents a relatively small fraction of the code (20%), see Fig. 1. Most of the optimizations therefore focused on this part and a re-write using a Domain Specific Language (DSL) was considered.



based on 2 km operational model of MeteoSwiss until 2016

Figure 1: COSMO components timing.

The DSL approach allows a separation of concerns between the user code and the hardware specific optimizations which are implemented in a so-called back-end. A given user code is in principle portable to and efficient on all architectures supported by the library that implements the DSL. The COSMO dynamical core was re-written using the STELLA Library which was developed as part of the project and for which an x86 CPU and a GPU back-end have been implemented. A detailed description of the STELLA library can be found in [Gysi 2015]. The CPU back-end employs OpenMP for threading (Task 3 and 5) while the GPU back-end relies on the CUDA language (Task 5). The specific data storage layout is hidden from the STELLA user allowing to change the underlying data structures for optimal performance on a given target hardware architecture (Task 2). Furthermore the library relies heavily on cache optimizations (Task 2).

For an application such as COSMO which has low compute intensity, i.e. low number of floating-point operations as compared to memory accesses, having only the dynamical core running on GPU is not a performant solution for running on an hybrid GPU system. This would indeed require costly synchronization between the host CPU and the GPU at every time steps which would counteract all performance gains obtained by using GPUs. It was therefore decided to adopt a so-called full GPU port strategy which means that every component of the model reading or writing three dimensional fields needs to run on the GPU see Fig. 2. The dynamics and communication are ported to GPU using the DSL while a compiler-based directive approach OpenACC (Task 7) is considered [Lapillonne 2014]. A detailed description of the COSMO GPU port can be found in [Fuhrer 2014].

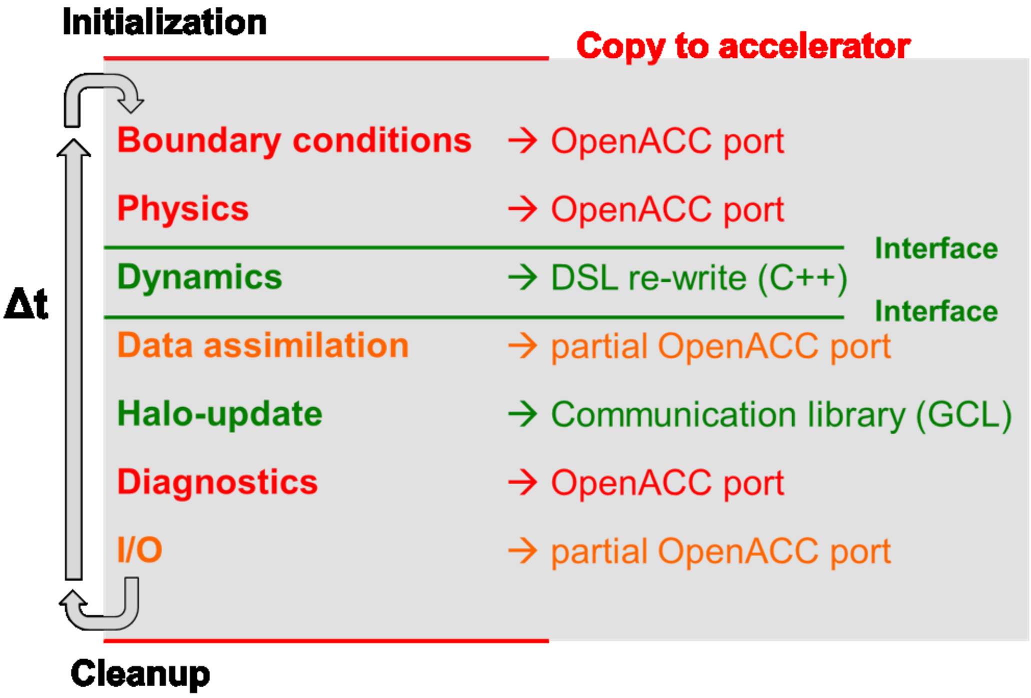


Figure 2: COSMO time loop. Full GPU port strategy.

The GPU capability of COSMO are available from COSMO version 5.6 on for the COSMO community. When comparing the runtime of COSMO on CPU and GPUs of similar generation the code runs typically 3 to 5 time faster on GPU (Fig. 3).

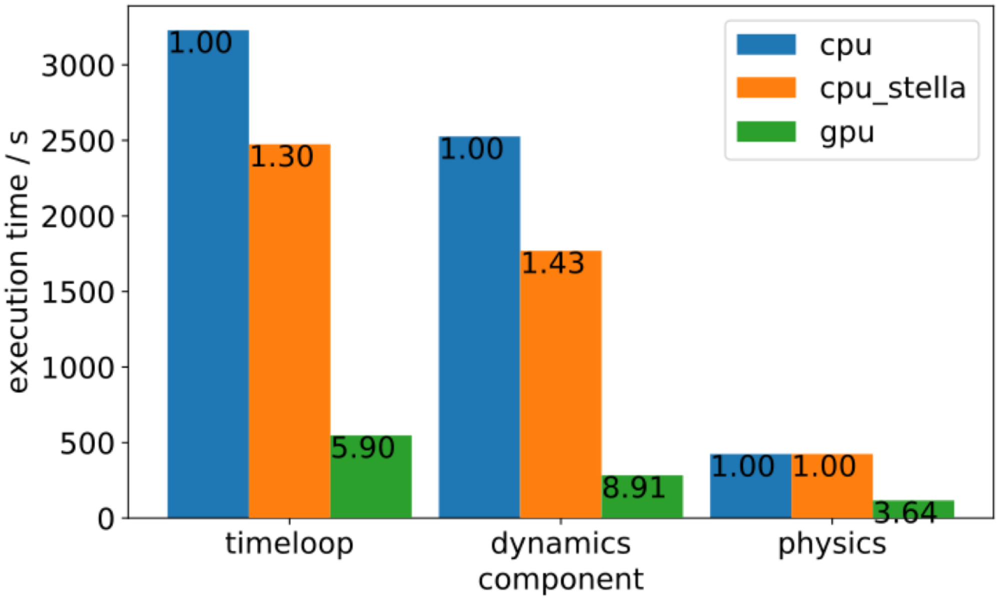


Figure 3: Performance comparison of a COSMO-1 6h benchmark on Piz Daint supercomputer. The CPU benchmark is run on 32 Intel Haswell (12 cores - Xeon E5-2690 v3) CPUs while the GPU is run using 32 P100 Nvidia cards. The speedup as compared to the reference Fortran code (i.e. without the DSL dynamics) is given inside the bars. The code used is COSMO 5.6 release-candidate.

In parallel to the GPU port, the physical parametrizations have been restructured to be shared with the ICON model using a block format (Task 6). This can in principle improve performance on CPU since it could increase cache reuse between the different parametrizations. This gain is however counterbalanced by the fact that one needs to copy the input and output data from the original ijk COSMO data structure to the block format. This change has a neutral effect on performance, but allows to share physics between the ICON and COSMO model.

The COSMO code was furthermore adapted to run using single precision floating point number (Task 8), in addition to double precision. A detailed description of the required adaptation can be found in [Rudisuhli 2014]. This provides an additional performance improvement ranging from a speed up factor of 1.2x to 1.3x depending on the architecture.

Finally, although the C++ dynamical core is making use of OpenMP parallelization on CPU, the rest of the code also had to be ported to OpenMP (Task 3). As compared to a flat MPI parallelization, introducing threading using OpenMP in the Fortran part did not lead to significant performance improvements. Unless more aggressive optimizations such as blocking would be introduced in the Fortran code (which would significantly reduce the maintainability and readability of the code), it was to be expected that a memory bandwidth bound code would not yield significant performance improvements. As a consequence, the changes introduced for hybrid MPI+OpenMP parallelism were not prioritized high enough to be introduced into the official version of the COSMO code.

3 Applications of the GPU and single precision capability

The COSMO model is run operationally at MeteoSwiss on GPU-based hardware and in single precision since 2016. The high performance computing system employs a fat-node design with multiple GPUs per CPU socket. From the operational point of view the use of a GPU system did not bring new issues, and the system is running stably as compared to the previous CPU only operational system. Currently, the system is being upgraded by a system with a similar fat node design but latest-generation hardware. The COSMO-LEPS ensemble forecast run at ECMWF is running in single precision, the performance gain associated with switching from single precision to double precision has enabled to run more ensemble members. Several groups at ETH Zurich are running the COSMO model on GPUs for climate applications on the Piz Daint supercomputer. This allowed researchers to run high-resolution (2-km) simulations on continental scale domain [Leutwyler 2016]

4 Lessons learned and outlook

With the advent of GPU-accelerated systems, the days where vendors would port applications to a specific hardware architecture within the framework of a procurement benchmark are over and the load of maintaining a code base that is ready to run on different hardware architectures has been shifted to the code maintainers and developers. Since hardware architectures are evolving rapidly and the lifetime of a supercomputer typically is between 3-6 years, this requires a tight collaboration with institutions and individuals who have an in-depth understanding of and outlook on developments in the HPC hardware sector.

As a side effect of this project the development infrastructure of COSMO has been greatly improved. The GPU as well as single/double precision capability of the model have increased the maintenance complexity, such that manual testing is not an option anymore. The Jenkins continuous integration platform is now being used to automatically test the main source code as well as development branches on all supported hardware architectures and floating point precision using the COSMO technical testsuite. The code is now hosted on the development infrastructure github which provides a modern interface for software development and in particular allows for online code reviews. Without these tools, the project could not have been such a success, and this type of infrastructure and workflow should be used in the future in the COMO community.

For the port to GPUs different technologies have been used. The OpenACC compiler directives have proven to be very efficient for incrementally porting large existing code bases. It also has the advantage of being moderately intrusive, as compared to a full re-write, and is well accepted by the developer community. On the other hand, adding directives to an existing legacy code make the code much harder to maintain. Indeed, several targets need now to be supported but the Fortran code does not provide component or unit testing functionality such that bugs introduced with changes or associated to a new compiler version require considerable amount of time to be identified. Further, the OpenACC programming model has proven not to be very robust, with the few compilers that support it introducing regressions when new versions come out. Cray has decided to drop support for OpenACC in the future and thus currently the only compiler supporting OpenACC is the PGI compiler owned by NVIDIA. An alternative implementation of OpenACC exists in GCC, but is not in a state that can be used for production. For some components, optimizations for GPU have adverse effects on CPU performance such that either the CPU performance has been given priority, or separate code for GPU has been introduced using ifdef pre-processor macro.

For the most critical component ported with OpenACC a source-to-source translation tool CLAW [Clement 2018], doing automatic code transformation, has been also used.

The STELLA DSL approach has proven successful to achieve performance portability. However, the project has shown that acceptance of this new technology by the end user (dycore developer) is very limited. In fact, there are currently two versions of the COSMO dynamical core in the official code, namely the Fortran and the DSL version. One possible reason for this is that the resulting C++ code is rather complex to handle for someone used to pure Fortran code. In addition, because of time constraints on both the DSL developer and end dycore user the exchange while developing the library was very limited, such that the final product did not meet the expectation of the end user. For a future project we recommend more focus and the usability of such a DSL approach as well as a better integration between the DSL developers and the end-users.

References

Clement, V., Ferrachat, S., Fuhrer, O., Lapillonne, X., Osuna, C. E., Pincus, R., ... & Sawyer, W. (2018, July). The CLAW DSL: Abstractions for Performance Portable Weather and Climate Models. In Proceedings of the Platform for Advanced Scientific Computing Conference (p. 2). ACM

Fuhrer, O., Osuna, C., Lapillonne, X., Gysi, T., Cumming, B., Bianco, M., Arteaga, A. and Schulthess, T.C., 2014. Towards a performance portable, architecture agnostic implementation strategy for weather and climate models. Supercomputing frontiers and innovations, 1(1), pp.45-62

Gysi, T., Osuna, C., Fuhrer, O., Bianco, M. and Schulthess, T.C., 2015, November. STELLA: a domain-specific tool for structured grid methods in weather and climate models. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (p. 41). ACM

Lapillonne, X., Fuhrer, O. (2014). Using compiler directives to port large scientific applications to GPUs: An example from atmospheric science. Parallel Processing Letters, 24(01), 1450003

Leutwyler, D., Fuhrer, O., Lapillonne, X., Lüthi, D., Schär, C. (2016). Towards European-scale convection-resolving climate simulations with GPUs: a study with COSMO 4.19. Geoscientific Model Development, 9(9), 3393-3412

Rudisuhli, S., Walser, A., Fuhrer, O. 2014, COSMO in Single Precision, COSMO Newsletter no. 14, p.18, www.cosmo-model.org

List of COSMO Newsletters and Technical Reports

(available for download from the COSMO Website: www.cosmo-model.org)

COSMO Newsletters

No. 1: February 2001.

No. 2: February 2002.

No. 3: February 2003.

No. 4: February 2004.

No. 5: April 2005.

No. 6: July 2006.

No. 7: April 2008; Proceedings from the 8th COSMO General Meeting in Bucharest, 2006.

No. 8: September 2008; Proceedings from the 9th COSMO General Meeting in Athens, 2007.

No. 9: December 2008.

No. 10: March 2010.

No. 11: April 2011.

No. 12: April 2012.

No. 13: April 2013.

No. 14: April 2014.

No. 15: July 2015.

No. 16: July 2016.

No. 17: July 2017.

No. 18: November 2018.

No. 19: October 2019.

COSMO Technical Reports

No. 1: Dmitrii Mironov and Matthias Raschendorfer (2001):
Evaluation of Empirical Parameters of the New LM Surface-Layer Parameterization Scheme. Results from Numerical Experiments Including the Soil Moisture Analysis.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_1

No. 2: Reinhold Schrodin and Erdmann Heise (2001):
The Multi-Layer Version of the DWD Soil Model TERRA_LM.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_2

No. 3: Günther Doms (2001):
A Scheme for Monotonic Numerical Diffusion in the LM.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_3

No. 4: Hans-Joachim Herzog, Ursula Schubert, Gerd Vogel, Adelheid Fiedler and Roswitha Kirchner (2002):
LLM ¯ the High-Resolving Nonhydrostatic Simulation Model in the DWD-Project LITFASS. Part I: Modelling Technique and Simulation Method.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_4

No. 5: Jean-Marie Bettems (2002):
EUCOS Impact Study Using the Limited-Area Non-Hydrostatic NWP Model in Operational Use at MeteoSwiss.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_5

No. 6: Heinz-Werner Bitzer and Jürgen Steppeler (2004):
Documentation of the Z-Coordinate Dynamical Core of LM.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_6

No. 7: Hans-Joachim Herzog, Almut Gassmann (2005):
Lorenz- and Charney-Phillips vertical grid experimentation using a compressible nonhydrostatic toy-model relevant to the fast-mode part of the 'Lokal-Modell'.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_7

No. 8: Chiara Marsigli, Andrea Montani, Tiziana Paccagnella, Davide Sacchetti, André Walser, Marco Arpagaus, Thomas Schumann (2005):
Evaluation of the Performance of the COSMO-LEPS System.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_8

No. 9: Erdmann Heise, Bodo Ritter, Reinhold Schrodin (2006):
Operational Implementation of the Multilayer Soil Model.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_9

No. 10: M.D. Tsyrulnikov (2007):
Is the particle filtering approach appropriate for meso-scale data assimilation ?
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_10

No. 11: Dmitrii V. Mironov (2008):
Parameterization of Lakes in Numerical Weather Prediction. Description of a Lake Model.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_11

No. 12: Adriano Raspanti (2009):
COSMO Priority Project ”VERification System Unified Survey” (VERSUS): Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_12

No. 13: Chiara Marsigli (2009):
COSMO Priority Project ”Short Range Ensemble Prediction System” (SREPS): Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_13

No. 14: Michael Baldauf (2009):
COSMO Priority Project ”Further Developments of the Runge-Kutta Time Integration Scheme” (RK): Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_14

No. 15: Silke Dierer (2009):
COSMO Priority Project "Tackle deficiencies in quantitative precipitation forecast" (QPF): Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_15

No. 16: Pierre Eckert (2009):
COSMO Priority Project "INTERP": Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_16

No. 17: D. Leuenberger, M. Stoll and A. Roches (2010):
Description of some convective indices implemented in the COSMO model.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_17

No. 18: Daniel Leuenberger (2010):
Statistical analysis of high-resolution COSMO Ensemble forecasts in view of Data Assimilation.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_18

No. 19: A. Montani, D. Cesari, C. Marsigli, T. Paccagnella (2010):
Seven years of activity in the field of mesoscale ensemble forecasting by the COSMO–LEPS system: main achievements and open challenges.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_19

No. 20: A. Roches, O. Fuhrer (2012):
Tracer module in the COSMO model.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_20

No. 21: Michael Baldauf (2013):
A new fast-waves solver for the Runge-Kutta dynamical core.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_21

No. 22: C. Marsigli, T. Diomede, A. Montani, T. Paccagnella, P. Louka, F. Gofa, A. Corigliano (2013):
The CONSENS Priority Project.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_22

No. 23: M. Baldauf, O. Fuhrer, M. J. Kurowski, G. de Morsier, M. Müllner, Z. P. Piotrowski, B. Rosa, P. L. Vitagliano, D. Wójcik, M. Ziemiański (2013):
The COSMO Priority Project 'Conservative Dynamical Core' Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_23

No. 24: A. K. Miltenberger, A. Roches, S. Pfahl, H. Wernli (2014):
Online Trajectory Module in COSMO: a short user guide.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_24

No. 25: P. Khain, I. Carmona, A. Voudouri, E. Avgoustoglou, J.-M. Bettems, F. Grazzini (2015):
The Proof of the Parameters Calibration Method: CALMO Progress Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_25

No. 26: D. Mironov, E. Machulskaya, B. Szintai, M. Raschendorfer, V. Perov, M. Chumakov, E. Avgoustoglou (2015):
The COSMO Priority Project 'UTCS' Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_26

No. 27: J-M. Bettems (2015):
The COSMO Priority Project 'COLOBOC': Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_27

No. 28: Ulrich Blahak (2016):
RADAR_MIE_LM and RADAR_MIELIB - Calculation of Radar Reflectivity from Model Output.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_28

No. 29: M. Tsyrulnikov and D. Gayfulin (2016):
A Stochastic Pattern Generator for ensemble applications.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_29

No. 30: D. Mironov and E. Machulskaya (2017):
A Turbulence Kinetic Energy – Scalar Variance Turbulence Parameterization Scheme.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_30

No. 31: P. Khain, I. Carmona, A. Voudouri, E. Avgoustoglou, J.-M. Bettems, F. Grazzini, P. Kaufmann (2017):
CALMO - Progress Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_31

No. 32: A. Voudouri, P. Khain, I. Carmona, E. Avgoustoglou, J.M. Bettems, F. Grazzini, O. Bellprat, P. Kaufmann and E. Bucchignani (2017):
Calibration of COSMO Model, Priority Project CALMO Final report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_32

No. 33: N. Vela (2017):
VAST 2.0 - User Manual.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_33

No. 34: C. Marsigli, D. Alferov, M. Arpagaus, E. Astakhova, R. Bonanno, G. Duniec, C. Gebhardt, W. Interewicz, N. Loglisci, A. Mazur, V. Maurer, A. Montani, A. Walser (2018):
COsmo Towards Ensembles at the Km-scale IN Our countries" (COTEKINO), Priority Project final report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_34

No. 35: G. Rivin, I. Rozinkina, E. Astakhova, A. Montani, D. Alferov, M. Arpagaus, D. Blinov, A. Bundel, M. Chumakov, P. Eckert, A. Euripides, J. Förstner, J. Helmert, E. Kazakova, A. Kirsanov, V. Kopeikin, E. Kukanova, D. Majewski, C. Marsigli, G. de Morsier, A. Muravev, T. Paccagnella, U. Schättler, C. Schraff, M. Shatunova, A. Shcherbakov, P. Steiner, M. Zaichenko (2018):
The COSMO Priority Project CORSO Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_35

No. 36: A. Raspanti, A. Celozzi, A. Troisi, A. Vocino, R. Bove, F. Batignani (2018):
The COSMO Priority Project VERSUS2 Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_36

No. 37: A. Bundel, F. Gofa, D. Alferov, E. Astakhova, P. Baumann, D. Boucouvala, U. Damrath, P. Eckert, A. Kirsanov, X. Lapillonne, J. Linkowska, C. Marsigli, A. Montani, A. Muraviev, E. Oberto, M.S. Tesini, N. Vela, A. Wyszogrodzki, M. Zaichenko, A. Walser (2019):
The COSMO Priority Project INSPECT Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_37

No. 38: G. Rivin, I. Rozinkina, E. Astakhova, A. Montani, J-M. Bettems, D. Alferov, D. Blinov, P. Eckert, A. Euripides, J. Helmert, M.Shatunova (2019):
The COSMO Priority Project CORSO-A Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_38

No. 39: C. Marsigli, D. Alferov, E. Astakhova, G. Duniec, D. Gayfulin, C. Gebhardt, W. Interewicz, N. Loglisci, F. Marcucci, A. Mazur, A. Montani, M. Tsyrulnikov, A. Walser (2019):
Studying perturbations for the representation of modeling uncertainties in Ensemble development (SPRED Priority Project): Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_39

No. 40: E. Bucchignani, P. Mercogliano, V. Garbero, M. Milelli, M. Varentsov, I. Rozinkina, G. Rivin, D. Blinov, A. Kirsanov, H. Wouters, J.-P. Schulz, U. Schättler (2019):
Analysis and Evaluation of TERRA_URB Scheme: PT AEVUS Final Report.
DOI: 10.5676/DWD_pub/nwv/cosmo-tr_40

COSMO Technical Reports

Issues of the COSMO Technical Reports series are published by the *COnsortium for Small-scale MOdelling* at non-regular intervals. COSMO is a European group for numerical weather prediction with participating meteorological services from Germany (DWD, AWGeophys), Greece (HNMS), Italy (USAM, ARPA-SIMC, ARPA Piemonte), Switzerland (MeteoSwiss), Poland (IMGW), Romania (NMA) and Russia (RHM). The general goal is to develop, improve and maintain a non-hydrostatic limited area modelling system to be used for both operational and research applications by the members of COSMO. This system is initially based on the COSMO-Model (previously known as LM) of DWD with its corresponding data assimilation system.

The Technical Reports are intended

- for scientific contributions and a documentation of research activities,
- to present and discuss results obtained from the model system,
- to present and discuss verification results and interpretation methods,
- for a documentation of technical changes to the model system,
- to give an overview of new components of the model system.

The purpose of these reports is to communicate results, changes and progress related to the LM model system relatively fast within the COSMO consortium, and also to inform other NWP groups on our current research activities. In this way the discussion on a specific topic can be stimulated at an early stage. In order to publish a report very soon after the completion of the manuscript, we have decided to omit a thorough reviewing procedure and only a rough check is done by the editors and a third reviewer. We apologize for typographical and other errors or inconsistencies which may still be present.

At present, the Technical Reports are available for download from the COSMO web site (www.cosmo-model.org). If required, the member meteorological centres can produce hard-copies by their own for distribution within their service. All members of the consortium will be informed about new issues by email.

For any comments and questions, please contact the editor:

Massimo Milelli
Massimo.Milelli@arpa.piemonte.it