HPC-Europa3 Transnational Access Programme

Main field: Earth Sciences & Environment

Dr. Roman Nuterman, University of Copenhagen, Niels Bohr Institute, Copenhagen, Denmark

"Online integrated atmospheric modelling: the Python way"

At the moment, we develop the online coupled meteorology-chemistry limited area modelling system Enviro-HIRLAM (in FORTRAN) and apply it for high-resolution modelling of weather and atmospheric composition. The current project is devoted to joint development of next generation atmospheric modelling system in pure Python. The model should inherit the numerics of dynamical core and physical parametrisations and must be able to predict atmospheric circulation, cloud microphysics and radiative processes as well as Earth's surface-atmosphere interactions. It should combine high-level abstractions and user friendliness on one hand, and grasp modern developments in HPC computing on the other hand. The project pursues the concept of open source software and new model code will be developed within GitHub infrastructure for its testing, documenting and distributing.

Background Information

During the last 40 years, numerical models of different Earth systems, such as atmosphere, ocean, land and ice-sheets, have been actively developed and applied for weather forecasting and climate prediction. Although these models [1-2] were built to efficiently utilize computing resources, such long development history has led to spaghetti source code, written in low-level programming languages like FORTRAN or C, with mixed coding styles. At the same time, most code developers in the environmental sciences are rather scientists than professional programmers, which makes source code modification or debugging even for an experienced scientist, not to mention new MSc and PhD students, a very challenging and time-consuming task.

At present, however, new high-level programming languages like Python, Julia, Scala and MATLAB, are designed with the explicit goal of improving code structure and thus readability. Among them, Python [3] stands out by its enormous popularity in the scientific community, powerful standard library, wide range of learning resources and large third-party package ecosystem. Furthermore, the recent benchmarks of Python's machine learning (ML) libraries [4], for instance Google's JAX [5], show a substantial speed-up of numerical codes on GPU, and to a lesser extent CPU. In addition to that, the same model code can be 2-5 times more energy efficient on GPU in comparison with a traditional FORTRAN model.

In our previous and ongoing joint collaborative research and development projects (EU FPs 6-7 and H2020), we developed the FORTRAN based Enviro-HIRLAM [2] modelling system and applied it for high-resolution meteorological and atmospheric composition predictions over different megapolises in Europen countries and China. Recent successful results of translating FORTRAN Ocean GCM (Python Ocean Model, PyOM [6]) to pure Python (Versatile Ocean Simulator, Veros [7]) encourage us to apply the same approach for the atmospheric model. Therefore, this aims to revamp our conventional research and development routine and to make a first step towards a new generation atmospheric model by initiating the Enviro-HIRLAM model translation, transition and porting to pure Python code as well as utilizing the modern capabilities of the IT Center for Science (CSC HPC, Finland). Such platform independent Python-based modelling system will make research and development more efficient and easier to handle/ transfer/ migrate between different architecture environments, energy utilization more efficient, and help to focus on science rather than source code development and optimization in the future projects/ activities.

References:

1. *Hazeleger et al.: EC-Earth. A Seamless Earth-System Prediction Approach in Action, BAMS, Oct. 2010, https://doi.org/10.1175/2010BAMS2877.1*
2. *Baklanov, A., Korsholm, U. S., Nuterman, R., Mahura, A., Nielsen, K. P., Sass, B. H., ... Gonzalez-Aparicio, I. (2017). Enviro-HIRLAM online integrated meteorology-chemistry modelling system: strategy, methodology, developments and applications (v7.2). Geoscientific Model Development,10(8), 2971-2999. https://doi.org/10.5194/gmd-10-2971-2017*
3. *https://www.python.org*
4. *Nuterman R., Häfner D., Jochum M., and Vinter B.: Next-generation geophysical modelling, EGU General Assembly 2020, https://doi.org/10.5194/egusphere-egu2020-20536; https://github.com/dionhaefner/pyhpc-benchmarks*
5. *Bradbury J., Frostig R., Hawkins P., Johnson M. J., Leary C., Maclaurin D. and Wanderman-Milne S., JAX: composable transformations of Python+NumPy programs, v.0.1.55, 2018, http://github.com/google/jax*
6. *PyOM: https://wiki.cen.uni-hamburg.de/ifm/TO/pyOM2*
7. *Häfner, D., Jacobsen, R. L., Eden, C., Kristensen, M. R. B., Jochum, M., Nuterman, R., and Vinter, B.: Veros v0.1 – a fast and versatile ocean simulator in pure Python, Geosci. Model Dev., 11, 3299-3312, https://doi.org/10.5194/gmd-11-3299-2018, 2018. https://veros.readthedocs.io/en/latest*