Re-engineering and optimization of the GEOtop software package

Elisa Bortoli[a,b], Giacomo Bertoldi[a], Alberto Sartori[c], Stefano Cozzini[d,e], Martin Palma[f], Luca Cattani[f]

[a] *Institute for Alpine Environment, Eurac Research, Bolzano, Italy.* [b] *OGS, National Institute of Oceanography and Applied Geophysics, Trieste, Italy.* [c] *SISSA, Scuola Internazionale Superiore di Studi Avanzati, Trieste, Italy.* [d] *CNR/IOM, Trieste, Italy.* [e] *Exact lab s.r.l. Trieste, Italy.* [f] *ICT department, Eurac Research, Bolzano, Italy.*

Introduction: Integrated hydrological models simulate the complex interactions between groundwater, surface water flows, vegetation and atmosphere. The mathematical representation of these interactions is still a great challenge, because of the composite physical processes involved. Therefore, the developed codes are often not easy portable and difficult to be handled. Moreover, the application to large domains or long time series (i.e. climate research) requires significant computational efforts. **GEOtop** is an integrated model, which describes the three-dimensional water flows in the soil and the energy exchanges with the atmosphere. The core components of the package were presented in the 2.0 version [1], which was written in C, released as free software open-source project, scientifically tested and published [2]. However, despite the scientific quality of the project, a modern software engineering approach was still missing. Moreover, the lack of a parallelization of thealgorithms hindered its use over large domains.

Aims: In this contribution, we present our recent software re-engineering efforts to create a robust and computationally efficient scientific software, easily usable to enable large scale, operational applications.

Results: We present here the new version 3.0 of GEOtop, validated and tested over a set of case studies. The code is documented on GitHub (https://github.com/geotopmodel/geotop) using Travis-CI for continuous integration tests. Now we are using meson and CMake as build system tools. We performed a code refactoring, moving from old and cumbersome data structures to an object oriented approach in C++, (i.e. use of template classes for vectors, matrices and tensors).
We present here the profiling results, obtained with likwid-perfctr , perfstat and callgrind. On the basis of those results, we replaced computational expensive maths operations with more efficient ones.

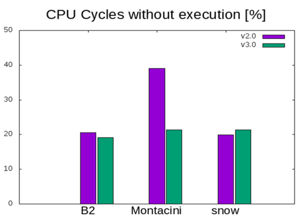

Fig. 1: CPU cycles without execution before and after math optimization for three test cases.

Finally, we pararellized with OpenMP part of the code loops of the most expensive functions. The version 3.0 has now lower value of CPU cycles without execution, as shown for some test cases in Figure 1.

Conclusions and next Steps: This activity is preliminary for a full code parallelization on the Vienna Scientific Cluster (VSC), which will be the next step of this project. The final aim is to develop a powerful HPC modelling infrastructure for real time monitoring systems for water resource management and climatic impact studies.

References

[1] Endrizzi et al.: GEOtop 2.0: simulating the combined energy and water balance at and below the land surface accounting for soil freezing, snow cover and terrain effects, Geosci. Model Dev.,7, 2831-2857, doi:10.5194/gmd-7-2831-2014, (2014). [2] Kollet et al.: The integrated hydrologic model intercomparison project, IH-MIP2: A second set of benchmark results to diagnose integrated hydrology and feedbacks. Water Resources Research. https://doi.org/10.1002/2016WR019191, (2016). [3] Bortoli et al.: Challenges in improving the HPC efficiency of the GEOtop 2.1 Integrated Hydrologic Model. Geophysical Research Abstracts Vol. 20, EGU2018-15892 (2018).