Geophysical Research Abstracts
Vol. 21, EGU2019-9584, 2019
EGU General Assembly 2019

Re-engineering and optimization of the GEOtop 3.0 integrated hydrological model

Giacomo Bertoldi (1), Elisa Bortoli (1), Alberto Sartori (2), Stefano Cozzini (3,4)

(1) Eurac research, Institute for the Alpine Environment, Bolzano, Italy (giacomo.bertoldi@eurac.edu), (2) SISSA, Scuola Internazionale Superiore di Studi Avanzati, Trieste, Italy., (3) CNR-IOM, Consiglio Nazionale delle Ricerche-Istituto Officina dei Materiali, Trieste, Italy., (4) Exact-lab, Trieste, Italy

The scientific software community is facing a challenge created by the confluence of disruptive changes in computing architectures and new opportunities for greatly improved data availability and simulation capabilities.
GEOtop is an integrated hydrological model, which simulates the heat and water budgets at and below the soil surface. It describes the three-dimensional water flow in the soil and the water and energy exchanges with the snowpack, the vegetation and the atmosphere, considering the effects of complex mountain topography.
The core components of the package were presented in the 2.0 version, which was released as Free Software Open-Source project. The code was written in C language. However, despite the high scientific quality of the project, a modern software engineering approach was still missing. Such weakness hindered its computational efficiency, its scientific potential and its use both as a standalone package and in an integrated way with other hydrological software tools and earth system models. Therefore, a software refactoring was needed to keep productivity with increasing amounts of data and to exploit the new available parallel architectures.
In this contribution, we present the new version 3.0 of GEOtop, validated and tested over a set of case studies. The code is documented on a GitHub repository (https://github.com/geotopmodel/geotop) using Travis-CI for continuous integration tests and google test framework for unit tests.
The previous 2.0 version was translated from C to C++ to take advantage of its rich function library, the possibility of using templates and especially to benefit from an Object-Oriented Programming. After a preliminary code cleaning, the old and cumbersome data structures were replaced by more modular and flexible ones, in perspective of an easier linear algebra optimization and code parallelization.
In order to test model performance, three representative experimental test cases were selected among the large suite of possible models configurations; the code was then profiled using likwid-perfctr, callgrind and a newly defined class Timer. On the basis of those results, a code optimization was started, improving the efficiency of most expensive mathematical operation and employing OpenMP parallelism for thread-safe functions. Then, performances and differences of the re-engineered 3.0 code were compared with the original 2.0 version.
The final aim is to develop a powerful HPC modelling infrastructure for real time monitor of water resources and climatic impact studies, that makes the best use of the new generation of hardware and software.