Geophysical Research Abstracts
Vol. 16, EGU2014-14611-1, 2014
EGU General Assembly 2014

How to achieve performance portable code using OpenACC compiler directives?

Xavier Lapillonne (1) and Oliver Fuhrer (2)

(1) ETHZ, C2SM, Zurich, Switzerland (xavier.lapillonne@env.ethz.ch), (2) Federal Office of Meteorology and Climatology MeteoSwiss, Zurich, Switzerland (oliver.fuhrer@meteoswiss.ch)

In view of adapting the weather and climate model COSMO to future architectures a new version of the model capable of running on graphics processing units (GPUs) has been developed. A large part of the code has been ported using compiler directives based on the OpenACC programming model. In order to achieve the best performance on GPUs several optimizations have been introduced for time critical components, mostly in the so-called physical parameterizations. Some of these modifications unfortunately degrade performance on traditional CPUs. Being a large community code, the COSMO model is required to perform well on both hybrid and CPU-only supercomputers. The current practical solution is to have separate source files for GPU and CPU execution, which may in the long-term result in maintenance issues.
Considering the physical parameterization responsible for the atmospheric radiative transfer computations, we first present the restructuring techniques necessary to achieve performance on the GPU. We then show that some parts of the code are compute bound on the CPU while memory bound limited on the GPU, leading to different requirements in terms of optimization. We finally discuss various solutions to achieve a portable and maintainable code, both in terms of possible improvement of the OpenACC standard or in terms of programming strategy.