Geophysical Research Abstracts
Vol. 18, EGU2016-8048, 2016
EGU General Assembly 2016

Generic algorithms for high performance scalable geocomputing

Kor de Jong (1,2), Oliver Schmitz (1), and Derek Karssenberg (1)

(1) Department of Physical Geography, Faculty of Geosciences, Utrecht University, Utrecht, The Netherlands (k.dejong1@uu.nl), (2) Geoneric, Zutphen, The Netherlands

During the last decade, the characteristics of computing hardware have changed a lot. For example, instead of a single general purpose CPU core, personal computers nowadays contain multiple cores per CPU and often general purpose accelerators, like GPUs. Additionally, compute nodes are often grouped together to form clusters or a supercomputer, providing enormous amounts of compute power.

For existing earth simulation models to be able to use modern hardware platforms, their compute intensive parts must be rewritten. This can be a major undertaking and may involve many technical challenges. Compute tasks must be distributed over CPU cores, offloaded to hardware accelerators, or distributed to different compute nodes. And ideally, all of this should be done in such a way that the compute task scales well with the hardware resources. This presents two challenges: 1) how to make good use of all the compute resources and 2) how to make these compute resources available for developers of simulation models, who may not (want to) have the required technical background for distributing compute tasks. The first challenge requires the use of specialized technology (e.g.: threads, OpenMP, MPI, OpenCL, CUDA). The second challenge requires the abstraction of the logic handling the distribution of compute tasks from the model-specific logic, hiding the technical details from the model developer.

To assist the model developer, we are developing a C++ software library (called Fern) containing algorithms that can use all CPU cores available in a single compute node (distributing tasks over multiple compute nodes will be done at a later stage). The algorithms are grid-based (finite difference) and include local and spatial operations such as convolution filters. The algorithms handle distribution of the compute tasks to CPU cores internally. In the resulting model the low-level details of how this is done is separated from the model-specific logic representing the modeled system. This contrasts with practices in which code for distributing of compute tasks is mixed with model-specific code, and results in a better maintainable model. For flexibility and efficiency, the algorithms are configurable at compile-time with the respect to the following aspects: data type, value type, no-data handling, input value domain handling, and output value range handling. This makes the algorithms usable in very different contexts, without the need for making intrusive changes to existing models when using them.

Applications that benefit from using the Fern library include the construction of forward simulation models in (global) hydrology (e.g. PCR-GLOBWB (Van Beek et al. 2011)), ecology, geomorphology, or land use change (e.g. PLUC (Verstegen et al. 2014)) and manipulation of hyper-resolution land surface data such as digital elevation models and remote sensing data. Using the Fern library, we have also created an add-on to the PCRaster Python Framework (Karssenberg et al. 2010) allowing its users to speed up their spatio-temporal models, sometimes by changing just a single line of Python code in their model.

In our presentation we will give an overview of the design of the algorithms, providing examples of different contexts where they can be used to replace existing sequential algorithms, including the PCRaster environmental modeling software (www.pcraster.eu). We will show how the algorithms can be configured to behave differently when necessary.

References

Karssenberg, D., Schmitz, O., Salamon, P., De Jong, K. and Bierkens, M.F.P., 2010, A software framework for construction of process-based stochastic spatio-temporal models and data assimilation. Environmental Modelling & Software, 25, pp. 489-502, Link. Best Paper Award 2010: Software and Decision Support.

Van Beek, L. P. H., Y. Wada, and M. F. P. Bierkens. 2011. Global monthly water stress: 1. Water balance and water availability. Water Resources Research. 47.

Verstegen, J. A., D. Karssenberg, F. van der Hilst, and A. P. C. Faaij. 2014. Identifying a land use change cellular automaton by Bayesian data assimilation. Environmental Modelling & Software 53:121–136.