WEATHER PROJECTIONS AND DYNAMICAL DOWNSCALING FOR THE REPUBLIC OF PANAMA: EVALUATION OF IMPLEMENTATION METHODS VIA GPGPU ACCELARATION

REINHARDT PINZÓN
Centro de Investigaciones Hidráulicas e Hidrotécnicas (CIHH), Universidad Tecnológica de Panamá (UTP), Sistema Nacional de Investigación (SNI), Panama, reinhardt.pinzon@utp.ac.pa

MICHEL MÜLLER
Typhoon Computing, Zug, Switzerland, mmueller.ee@gmail.com

TOSHIYUKI NAKAEGAWA
Meteorological Research Institute (MRI), Tsukuba, Japan, tnakaega@mri-jma.go.jp

JAVIER SÁNCHEZ-GALÁN
Facultad de Ingeniería de Sistemas Computacionales (FISC), Universidad Tecnológica de Panamá (UTP), Sistema Nacional de Investigación (SNI), Panama, javier.sanchezgalan@utp.ac.pa

MANUEL UJALDÓN
Universidad de Málaga, Málaga, Spain, ujaldon@uma.es

JOSÉ FÁBREGA
Centro de Investigaciones Hidráulicas e Hidrotécnicas (CIHH), Universidad Tecnológica de Panamá (UTP), Sistema Nacional de Investigación (SNI), Panama, jose.fabrega@utp.ac.pa

ABSTRACT

Climate change could have a critical impact on the Republic of Panama where a major segment of the economy is dependent on the operation of the Panama Canal. New capabilities to do targeted research around climate change impacts on Panama is therefore being established. This includes anew GPU-cluster infrastructure called Iberogun, based around 2 DGX1 servers (each running 16 NVIDIA Tesla P100 GPUs). This infrastructure will be used to evaluate potential climate models and models of extreme weather events. In this review we therefore present an evaluation of the GPGPU (General Purpose Graphic Processing Unit, here abbreviated GPU) implementation methods for the study of weather projections and dynamical downscaling in the Republic of Panama. Different methods are discussed, including: domain-specific languages (DSLs), directive-based porting methods, granularity optimization methods, and memory layout transforming methods. One of these approaches that has yielded interesting previous results is further discussed, a directive-based code transformation method called 'Hybrid Fortran' that permits a high-performance GPU port for arranged lattice Fortran codes. Finally, we suggest a method akin to previous investigations related to climate change done for the Republic of Panama, but with acceleration via GPU capabilities.

Keywords: Panama, CUDA, OpenACC, Iberogun GPU-cluster, Fortran, weather projections, climate

INTRODUCTION

The Republic of Panama is located in the southernmost range of Central America to the north of the equator (7°–10°N, 77°–83°W). The country is surrounded by the Caribbean Sea in the north and by the Pacific Ocean in the south. As Panama is enclosed by warm tropical oceans, the climate is mostly determined by a warm and moist oceanic atmosphere which is characteristic in most countries of Central America as found in Kusunoki *et al.* (2019). The Panama Canal is one of the utmost significant facilities in Panama. Earnings from the passage of containers pays about 40% of Gross Domestic Product (GDP) of Panama and Canal water source and processes deeply depend on precipitation over the river watershed of the Panama Canal as mentioned in Fábrega *et al*. (2013).

The Meteorological Research Institute—Atmospheric General Circulation Model, version 3.2 (MRI-AGCM3.2S) which has a grid size of 20 km has been used for weather projection and dynamical simulations in Panama. Using this model, our group has investigated future precipitation changes in Central America as is mentioned in Pinzón *et al*. (2017). Also, Nakaegawa *et al*. (2019) has investigated the climatological seasonal changes of the diurnal variation of precipitation at four ground stations in the upper Rio Chagres basin. The upper Rio Chagres basin is a sub-basin on the eastern side of the Panama Canal watershed.

Most of these studies have been carried out using the MRI-AGCM3.2S model, with simulations carried using the MRI's cluster facilities in Tsukuba, Japan. A visualization of the resulting from this simulation is showed in a Figure 2 depicting geographical distribution of precipitation in Panama with NHRCM with 2-km grid spacing.

A point of interest for the local researchers is that this computational capacity now developed in Japan via Global Circulation Model, can be used as a starting point for it to be replicated in Panamanian soil and that a local model is developed. Moreover, it can be interesting to compare results from previous investigations related to climate change done for Panama to results from a GPUs' approach implemented in Panama correcting for local biases.

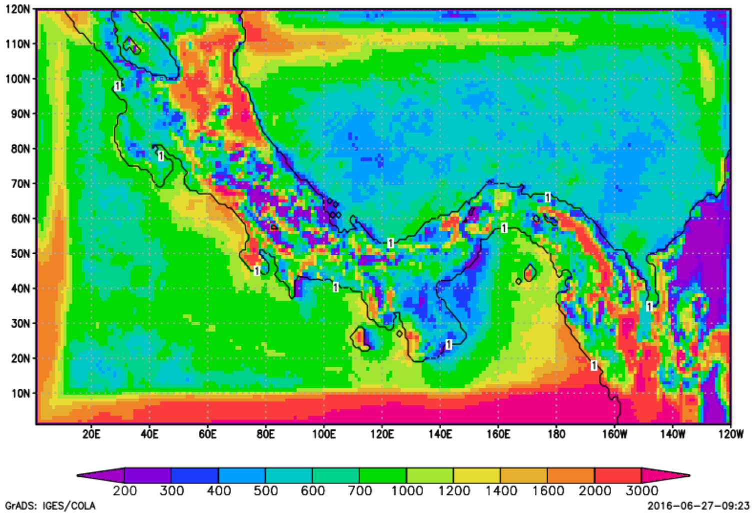

Figure 2. A possible result using a 2km resolution grid of NHRCM with a real-world weather situation depicting geographical distribution of annual precipitation

Regarding such local implementations and specifically using acceleration via GPU, Müller and Aoki (2018) have developed a GPU implementation of ASUCA, the main mesoscale weather prediction model used in operation and developed at the Japan Meteorological Agency, as found in Ishida *et al*. (2010). ASUCA uses a structured grid and is implemented in Fortran. In order to avoid a majority of the required programming changes, a meta-programming and transpiler framework called 'Hybrid Fortran' was first developed and then applied to ASUCA.

In the following review we therefore will investigate methods to re-target structured grid applications via GPU acceleration, with low programming effort for an efficient GPU implementation for the Panama case. Moreover, once our GPU-cluster called Iberogun has been installed and tested, then we will evaluate potential climate models and models of extreme weather events, such as WRF like ASUCA. This implies the following requirements for potential solutions:

(1) In current research the focus is on NVIDIA GPUs architecture, but the switch to others should be possible and streamlined.

(2) The user code should stay as close as possible to the original codebase (for instance WRF, NHRCM, etc.) to ease the transition. Since for some of these models the original code was written in Fortran, which does have some industry support for accelerators, the language should thus be kept if possible.

Finally, we draw conclusions about our preliminary evaluation of the implementation via GPU Acceleration.

RELATED WORK

The following GPU ports of atmospheric models are relevant for the discussion of our investigation related to evaluate potential climate models in our GPUs cluster.

ASUCA. Since 2010, this model has been extensively investigated for GPGPU acceleration. At first a GPU-only CUDA C port of the dynamical core and part of the physical core was implemented. 14 TFLOP/s were achieved in single precision using 528 Tesla S1070 GPUs. In 2011, 24 GFLOP/s was achieved in double precision for dynamical core on a single NVIDIA Fermi based Tesla M2050 GPU on TSUBAME 2.0, a 6-fold speedup when compared to the system's Xeon X5670 CPUs running the original Fortran code. In 2014, the previous work was generalized by employing a stencil library and DSL based on C++ templates that generates either CUDA kernels or CPU code.

COSMO. In 2014, MeteoSwiss' Fuhrer et al. presented a mixed approach for COSMO, a structured grid regional weather and climate model used in operation in several European nations. In this mixed approach, a C++ stencil library and DSL called "STELLA" is used for the dynamical core, while a less disruptive OpenACC-based method is applied to the physical processes in order to retain most of the original Fortran codebase. Thanks to a code refactoring a speedup of 1.5x was achieved on CPU. By employing GPU code generation an additional speedup of 2.9x was achieved (comparing Tesla K20x to 8-core Xeon E5-2670 on Piz Daint).

Weather Research and Forecasting Model (WRF). This joint model by the U.S. National Center for Atmospheric Research (NCAR) and the U.S. National Oceanic and Atmospheric Administration (NOAA) consists of multiple hundreds of thousands of lines of Fortran code. GPU ports known to us have been only partial, leading to heavy pressure on the memory bus between CPU and GPU and thus a lower speedup compared to full GPU ports of other weather models.

Non-hydrostatic Regional Climate Model (NHRCM). NHRCM was developed based on an operational Non-Hydrostatic Model (NHM) produced by the Meteorological Research Institute (MRI) and the Numerical Prediction Division of the Japan Meteorological Agency (NPD/JMA). The model is modified for climate projection. In NHRCM, the MRI-JMA Simple Biosphere model is used for describing the biosphere process, surface temperature, and snow depth. The Kain and Fritsch scheme (Kain and Fritsch 1993) is used to parameterize cumulus convection. NHRCM has been experimentally used to project a late future climate in Panama.

METHOD OUTLINE

In this section we provide an outline of notable implementation methods used in weather- and climate models, as well as methods for GPU acceleration in terms of memory layout- or code granularity optimizations.

1. Domain-Specific Languages.

Domain-specific languages applied to stencil algorithms have been one method to abstract parallelization boiler-plate and memory layout for hybrid GPU/CPU (General Purpose Graphic Processing Unit) code. For this matter, direct data accesses with loop- or thread indices are abstracted in the point-wise stencil code. This generally requires a complete rewrite of existing code.

Shimokawabe et al. (2014) have completed an implementation of the ASUCA dynamical core and a portion of its physical processes using their own C++ library. Memory access patterns are abstracted using a stencil DSL. Kernels are defined as C++ functors, passed to the library for parallelization. Shimokawabe et al. use a combination of MPI processes for multi-node parallelization, OpenMP for multi-GPU or multi-core CPU as well as CUDA for the GPU thread level parallelization.

Jumah et al. (2017) have proposed a general grid definition and manipulation language (GGDML), an extension to Fortran with applicability to other languages, based on the requirements for three existing models: DYNAMICO, ICON and NICAM. This shares some of characteristics of Grid Tools as well as the work by Shimokawabe et al. (2014), however with some important distinctions: firstly, GGDML is designed from the beginning to support icosahedral grids, and secondly, GGDML extends Fortran instead of relying on C++ user code, which facilitates migration from mostly Fortran based climate- and weather models.

The Atlas library by Deconinck et al. (2017) goes a step beyond Shimokawabe et al. (2014) in terms of the abstraction chosen for data structures. Rather than letting the user directly specify stencil o sets, it separates the concerns between function space and the underlying data model, including the chosen grid structure.

2. Directive-Based Porting Methods

Directives are used to steer compilers on how to optimize or parallelize already existing code for a specific hardware architecture:

Govett et al. (2014) have ported the dynamical core of the Fortran-based Nonhydrostatic Icosahedral Model (NIM) to GPU, first using their own directive-based transformation tool F2C-ACC" and later using OpenACC.

Norman et al. (2017) have implemented the “Accelerated Model for Climate and Energy" (ACME) for the U.S. DOE using OpenACC. As with ASUCA, ACME's physical processes are problematic for GPU due to their coarse-grained parallelization. GPU-specific code duplication was the only solution found when using OpenACC.

3. Granularity Optimization Methods

Kernel fusion has been the main approach to granularity optimization applied to GPGPU programming we are aware of. We take note of the following related work:

Gysi and Hoefler (2017) have implemented kernel fusion as well as loop fusion for the aforementioned STELLA framework.

The granularity optimization technique by Müller and Aoki (2018) used in Hybrid Fortran as discussed below is another example of granularity optimization methods.

4. Memory Layout Transforming Methods

While DSLs abstract the memory layout, they also require a full rewrite of the point-wise code. The following work allows for the reuse of existing code while keeping the performance portability gains of an abstracted memory layout:

With the C++ library, Edwards et al. (2014) have demonstrated that existing point-wise code can be reused even when the underlying data structures are converted to an abstracted memory layout.

With a DSL created for the climate model ICON, Torres et al. (2013) have shown that the Fortran syntax can be extended to allow for an abstracted memory layout.

Hybrid Fortran transforms memory access patterns for existing codes as discussed below.

HYBRID FORTRAN

One of the methods we evaluate is a Hybrid Fortran-based implementation, which is a source-to-source transformation and language extension that supports different targets. It uses use Modern Fortran together with higher level parallelization and data specification abstractions as input, and outputs either OpenMP Fortran, CUDA Fortran or OpenACC Fortran sources. Significantly, this tool presents an assisted splitting method for large kernels, a necessity in order to enable GPU compatibility of physical processes within the different climate models to be considered (WRF, NHRCM, etc.) without requiring a rewrite. In addition, it allows to transform the memory layout used within the user code without rewrite.

As Figure 1 shows from an earlier result, by employing Hybrid Fortran directives to both the dynamical core and the physical processes of the different user code, nearly all modules required for an operative weather prediction have already been ported to GPU.

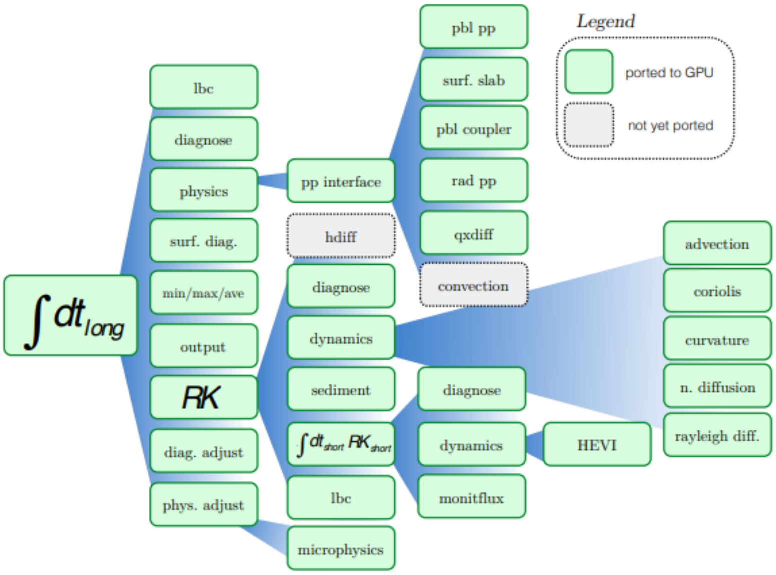

Figure. 1. Simplified call graph and status of Hybrid Fortran based implementation. Image from Müller and Aoki (2018)

GPU-CLUSTER IBEROGUN

We foresee to put together a GPU-cluster infrastructure called Iberogun, which will be based around 2 DGX1 servers (16 NVIDIA Tesla P100) in order to evaluate potential climate models and models of extreme weather events, such as WRF and COSMOS as well.

Iberogun will have have the following output characteristics: peta FLOPS 170 tera FLOPS, GPU Memory 128 GB total system, CPU Dual 20-Core Intel Xeon, E5-2698 v4 2.2 GHz NVIDIA CUDA® Cores 40,960 28,672, NVIDIA Tensor Cores (on V100 based systems) 5,120 N/A, System Memory 512 GB 2,133 MHz DDR4, LRDIMM Storage 4X 1.92 TB SSD RAID 0, Network Dual 10 GbE.

PROJECT IMPLEMENTATION BENEFITS

Regardless of the method used for the weather projections the benefits of such an implementation can be well received both in the research community and the society at large. Models that are fitted to the parameters of the isthmus of Panama can be beneficial for water cycles prediction for the canal basin area, very important information since the recent strict water use policies that have been proposed to the canal and the salinization phenomena that has been reported for the Panama Canal watershed. Moreover, benefits could ripple into other areas of research and commerce, such as: agriculture, hydraulic energy generation, transportation, ecology and conservation.

CONCLUSIONS

In this article, methods for the implementation of weather models accelerated via GPU has been evaluated. Different methods are discussed including: Domain-Specific Languages, Directive-Based Porting Methods, Granularity Optimization Methods, Memory Layout Transforming Methods a there is a special mention to Hybrid Fortran implementation. Characteristics of the Hybrid Fortran are introduced as a new approach to port structured grid Fortran applications to GPU. Benefits of this method over other architecture are discussed, including the possibility of use a combination of OpenMP, OpenACC, and CUDA Fortran in the backend with a combined programming interface. Also, issues about storage order and parallelization granularity are discussed. A new implementation for the different climate models, using Hybrid Fortran to both dynamical core and physical processes is examined. Finally, there is a growing landscape of tools, frameworks and libraries that can assist GPGPU portation of earth system models. However, there are vast differences in productivity when applying these methods to existing codes, which also has impact on performance since crucial engineering time is lost doing more menial work. We are considering Hybrid Fortran to have a good balance of productivity- and performance focused features, however we also note that the final decision will depend on the models chosen for further evaluation. The value of this article, lays in the fact that to the best of our knowledge this is the first time a complete production weather prediction model for the Republic of Panama will be ported to GPU using a single unified programming paradigm.

ACKNOWLEDGMENTS

RP acknowledges a scientific fund from Sistema Nacional de Investigación de Panamá (SNI) and Projects: FID-2016-275 and EIE-2018-16 of Convocatorias públicas of Secretaria Nacional de Ciencia y Tecnología e Innovación (SENACYT). TN acknowledges funds and support from JSPS Grant-in-Aid for Specially Promoted Research 16H06291. IT, and TN acknowledge Theme C of the TOUGOU program granted by the Japanese Ministry of Education, Culture, Sports, Science and Technology. The authors thank the Universidad Tecnológica de Panamá their extensive support, and for the use of their CIHH-group HPC-Cluster-Iberogun. Also acknowledge to NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

Kusunoki, S., Nakaegawa, T., Pinzón, R. *et al*. (2019). Future precipitation changes over Panama projected with the atmospheric global model MRI-AGCM3.2. Climate Dynamics 53, 5019–5034. DOI:10.1007/s00382-019-04842-w.

Fábrega J., Nakaegawa T., Pinzón R., Nakayama K., Arakawa O., and SOUSEI Theme-C modeling group. (2013). Hydrological Research Lett. 7:23–29. DOI: https://doi.org/10.3178/hrl.7.23.

Pinzón, RE., Hibino, K., Takayabu I., and Nakaegawa T. (2017). Virtually experiencing future climate changes in Central America with MRI-AGCM: climate analogues study. Hydrological Research Letters, 11, 106-113. DOI: https://doi.org/10.3178/hrl.11.106.

Nakaegawa, T., Pinzón, R., Fábrega, J., Cuevas. JA., De Lima, HA., Córdoba, E., *et al*. (2019). Seasonal changes of the diurnal variation of precipitation in the upper Río Chagres basin, Panama´. PLoS ONE 14 (12): e0224662. DOI: https://doi. org/10.1371/journal.pone.0224662.

Müller, M., and Aoki, T. (2018). New High Performance GPGPU Code Transformation Framework Applied to Large Production Weather Prediction Code. ACM Transactions on Parallel Computing (TOPC) archive Volume 5 Issue 2.

Shimokawabe, T., Aoki, T. and Onodera, N. (2014). High-productivity framework on GPU-rich supercomputers for operational weather prediction code ASUCA. in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC '14, (Piscataway, NJ, USA), pp. 251-261, IEEE.

Jumah, N., Kunkel, J., Zängl, G., Yashiro, H., Dubos, T., and Meurdesoif, Y. (2017). GGDML: Icosahedral models language extensions.

Deconinck, W., Bauer, P., Diamantakis, M., Hamrud, M., Kühnlein, C., Maciel, P., Mengaldo, G., Quintino, T., Raoult, B., Smolarkiewicz, PK., and Wedi, NP. (2017). Atlas: A library for numerical weather prediction and climate modelling. Computer Physics Communications, vol. 220, no. Supplement C, pp. 188- 204.

Fuhrer, O., Osuna, C., Lapillonne, X., Gysi, T., Cumming, B., Bianco, M., ... & Schulthess, T. C. (2014). Towards a performance portable, architecture agnostic implementation strategy for weather and climate models. Supercomputing frontiers and innovations, 1(1), 45-62.

Lapillonne, X., and Fuhrer, O. (2014). Using compiler directives to port large scientific applications to GPUs: An example from atmospheric science. Parallel Processing Letters, vol. 24, no. 01.

Govett, M., et al. (2014). Directive-based parallelization of the NIM weather model for GPUs, in Accelerator Programming using Directives (WACCPD), First Workshop on, pp. 55-61, IEEE.

Norman, MR., Mametjanov, A., and Taylor, M. (2017). Exascale programming approaches for the accelerated model for climate and energy.

Gysi, T., and Hoefler, T. (2017). Integrating STELLA & MODESTO: Definition and optimization of complex stencil programs.

Edwards, HC., Trott, CR., and Sunderland, D. (2014). Kokkos: Enabling many-core performance portability through polymorphic memory access patterns. Journal of Parallel and Distributed Computing, vol. 74, no. 12, pp. 3202-3216. Domain-specific languages and high-level frameworks for high-performance computing.

Torres, R., Linardakis, L., Kunkel, J., and Ludwig, T. (2013). ICON DSL: A domain specific language for climate modeling. in International Conference for High Performance Computing, Networking, Storage and Analysis, Denver, CO.

Ishida, J., Muroi, Ch., Kawano K., and Yuji Kitamura. (2010). Development of a new non-hydrostatic model ASUCA at JMA. CAS/JSC WGNE Research Activities in Atmospheric and Oceanic Modelling 40 (2010), 0511–0512.

Universidad Tecnológica de Panamá. (2020). CIHH-group HPC-Cluster-Iberogun: http://hpc-simulations.utp.ac.pa/