Sprint documentation #1

ICON-ART Sprint

Enrico Degregori[1] and Ali Hoshyaripour[2]

[1]Deutsches Klimarechenzentrum GmbH, Hamburg, Germany
[2]Karlsruhe Institute of Technology, Karlsruhe, Germany

Contact: info@nat-esm.de

Published on 17.08.2022 on https://www.nat-esm.de/services/accepted-sprints

1 Summary

The focus of the short advisory activity is a preparatory analysis of the ART code for GPU porting. The Aerosol and Reactive Trace gases (ART) is a submodule within the ICON (ICOsahedral Nonhydrostatic) model and it is used for operational forecasts of dust, volcanic ash and pollen.

The natESM sprint has the objective to start preparing ART for GPU-based HPC systems because, while there have been activities in this direction with respect to ICON, ART has remained unexplored.

ICON-ART programming language is Fortran 2003/2008.

2 General Information

Start and end date:	14.04.2022 – 17.06.2022
Intended period:	2 months
Responsible RSE:	Enrico Degregori, DKRZ
Responsible scientist:	Dr. Ali Hoshyaripour, KIT

3 Sprint Objectives

At the beginning of the sprint a set of 4 test cases was provided, 2 of them operational. A performance analysis was conducted and static call graphs of the ART interfaces were created for each test case.

Then, the focal point of the sprint was outlined in the definition of a step-by-step strategy for GPU porting. The ART interfaces are scattered in the main ICON integration loop and this design does not allow to run them on CPUs while everything else is running on GPUs because of the amount of data movement. For the same reason, also alternative approaches such as concurrency would not be straightforward to implement and scientifically validate.

The objectives of the sprint are summarized as follows:

- analysis of ART interfaces and their modularity
- definition of data that need to be moved on GPU for each interface (based on a specific experiment)

- algorithm analysis for GPU porting
- final definition of a roadmap for a meaningful experiment

4 Procedure and Insights

4.1 Technical Approach / Procedure

The main topics covered to define the final roadmap are the following:

- data movement for each interface (based on Volaero experiment) and guidance on deep copies of Fortran derived types
- guidance on NVIDIA compiler function inlining and list of possible subroutines/functions which should be extracted
- first level optimization which might involve some basic refactoring of the code (i.e. loops fusion with gang static, temporary array allocation and reuse, memory coalescing, etc.) with a list of subroutines/functions to be analysed after the initial porting.

4.2 General Insights

During the performance analysis of the test cases, two major hotspots were identified in the chemical reaction interface and in the aerosol dynamics interface. This is related to some legacy code within modules external to ART.

The code design and the algorithm prevent optimal performance on modern CPUs (i.e. no efficient vectorization) and poor performance is also expected on GPUs because of divergent branches and code jumps (GO TO statements).

5 Results

The ART GPU porting can benefit from the work done so far in ICON because it is working on the same domain decomposition (same loops structure and memory layout) and it is sharing some data structures.

The main differences are:

- Fortran derived types defined in ART which require a specific strategy for deep copies
- external modules which contain legacy code not optimized for both modern CPUs or GPUs

The refactoring of legacy code represents the major effort to obtain optimal performance on GPUs.

6 Conclusions and Outlook

The sprint outcome identified the ART external modules as the main possible limitation for optimal GPU performance. Since significant code refactoring might be needed, the isorropia module within the aerosol dynamics was extracted from ART to provide a standalone version that can run on GPUs. The reason why this specific module was extracted is that it is the main hotspot in the Volaero experiment which was the main focus for the GPU porting roadmap.

The standalone isorropia can be used as a development environment during the code refactoring and during the initial performance analysis if a meaningful driver is provided. This represents a possibility for a future natESM sprint.

7 References

A full documentation can be found on Gitlab: https://gitlab.dkrz.de/natESM/pre-artex