論文 / 著書情報
Article / Book Information

題目(和文)	現業気象予報モデルをソースコード変換によりGPU計算に適用させる高生産性フレームワーク
Title(English)	A Framework for High Productivity GPU Acceleration Through Code Transformation Applied to Operational Weather Prediction Model in Japan
著者(和文)	ミュラー ミシェル
Author(English)	Michel Mueller
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第10851号, 授与年月日:2018年3月26日, 学位の種別:課程博士, 審査員:青木 尊之,奥野 喜裕,末包 哲也,肖 鋒,長崎 孝夫
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第10851号, Conferred date:2018/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Category(English)	Doctoral Thesis
種別(和文)	論文要旨
Type(English)	Summary

（博士課程）
Doctoral Program

論 文 要 旨

THESIS SUMMARY

専攻： Department of	創造エネルギー　専攻	申請学位（専攻分野）： Academic Degree Requested	博士　(Engineering) Doctor of
学生氏名： Student's Name	Mueller Michel	指導教員（主）： Academic Supervisor(main)	教授 青木 尊之
		指導教員（副）： Academic Supervisor (sub)	准教授　横田　理央

要旨（和文 2000 字程度）
Thesis Summary（approx.2000 Japanese Characters ）

Numerical weather prediction (NWP) has been a grand challenge for computing going back to its inception at the beginning of the 20th century. Today's NWP models are steadily advancing towards increasingly high grid resolutions. This advancement is accompanied with an increasing demand for memory bandwidth. Graphics processing units (GPU) are very promising in this regard thanks to their highly performant memory system. On the other hand, NWP is a very demanding environment with regards to operational requirements for the codebases in use - maintainability and ease-of-use for domain scientists is paramount for a successful deployment of new hardware architectures. There is a gap in studies and frameworks where productivity plays a central role. In this thesis we therefore want to close this gap.

We have used the Japanese regional weather prediction model "ASUCA as a case study on how to design a maintainable and efficient GPU codebase for NWP. Two technical challenges stand out in terms of their potential impact on productivity and code integrity: The granularity of physical processes tends to be too coarse for GPU, and the memory layout of data structures needs to be reordered, as these codes have been developed with CPUs in mind. Existing approaches to solve these issues have led to varying combinations of limited performance, poor maintainability and invasive rewrites. For ASUCA, an approach has been sought that comes with none of these drawbacks.

Our proposed method to enable GPUs in NWP has been implemented as a language extension and transformation framework called "Hybrid Fortran". This technique requires only minimal changes for a CPU targeted codebase, a significant advancement in terms of productivity. The aforementioned technical challenges are addressed as follows: Parallel loops are abstracted through the means of a new language construct ("parallel region"), allowing to specify multiple granularity levels with the same code, depending on the target architecture (thus keeping support for CPU). Memory layout is transformed at compile-time, enabling the reuse of existing code with zero run-time overhead. In the backend the parallelization is implemented through source-to-source translation for many-core GPUs or multi-core CPUs using an original Python-based framework.

Hybrid Fortran has been applied to ASUCA's dynamical core as well as planetary boundary layer-, surface and radiation processes, cloud microphysics, precipitation. Using the aforementioned memory layout transformation, the new hybrid version of ASUCA is using IJK order for GPU and KIJ order for CPU. Physical processes are transformed at compile-time from coarse-grained- to fine-grained parallelization for GPU. ASUCA's codebase encompasses more than 150k lines of code. By using extensive code transformation techniques, more than 85% of the original codebase was left unchanged and the overall code size has been extended by less than 4%, even though the new codebase targets two very different hardware architectures instead of one. This shows very promising productivity, maintainability and ease-of-adoption of the proposed method in an operational setting.

By means of a minimal weather application that resembles ASUCA's code structure we have compared our method to a performance model as well as today's commonly used method, OpenACC. As a result, the Hybrid Fortran-based implementation is shown to deliver the same or better performance than OpenACC for GPU (many-core) and OpenMP for CPU (multi-core) and its performance agrees with the model both on CPU and GPU. In a full-scale production run, using an ASUCA grid with 1581 x 1301 x 58 cells and real-world weather data in 2km resolution, 24 NVIDIA Tesla P100 running the Hybrid Fortran based GPU port are shown to replace more than 50 18-core Intel Xeon Broadwell E5-2695 v4 running the reference implementation. In terms of kernel performance, a speedup of more than 3x is shown between latest-generation the same CPUs and GPUs.

In order to present a wider view of the software tooling efforts in the field of weather- and climate modelling, a review of relevant hybrid GPU- and CPU implementation methods has been conducted, including a characterization and comparison by means of a productivity model. There are use cases where a higher-level abstraction is useful (e.g. to abstract the underlying grid). On the opposite end, in some cases where the application is already GPU friendly (e.g. dynamical cores that already have a performant memory layout), a more direct directive-based approach can be the most productive. Hybrid Fortran presents a middle ground, providing much of the benefit of higher level abstractions that are highly invasive to existing codebases, yet requiring significantly less code to be changed.

We conclude by listing the contributions of this work to the field of scientific computing. We were able to significantly speed up a major weather prediction model using GPUs, with the crucial additional benefit of retaining a codebase that is easy to maintain for domain scientists. Our proposed method promises to be highly applicable and valuable in the wider field, which is shown by comparing our approach to similar and recently published projects where the problems, solved with our method, were a significant hurdle to productivity and/or efficiency. This shows the potential for a high impact of this work.