

大気大循環モデル GCM に対するディレクティブベース手法を用いた GPU 実装の評価

#迫田 祥司¹⁾, 深沢 圭一郎²⁾, 三好 勉信³⁾, 岩下 武史⁴⁾

⁽¹⁾ 京大院情報学, ⁽²⁾ 総合地球環境学研究所, ⁽³⁾ 九州大学, ⁽⁴⁾ 京都大学学術情報メディアセンター

Evaluation of Directive-Based GPU Implementation for an Atmospheric General Circulation Model

#Shoji Sakoda¹⁾, Keiichiro FUKAZAWA²⁾, Yasunobu MIYOSHI³⁾, Takeshi Iwashita⁴⁾

⁽¹⁾Kyoto University Graduate School of Informatics, ⁽²⁾Research Institute for Humanitics and Nature, ⁽³⁾Kyushu University,

⁽⁴⁾Academic Center for Computing and Media Studies, Kyoto University

The computational cost of numerical simulations is substantial due to the huge number of grid points required for higher resolutions and the complexity of models required for better accuracy. One way to speed up these simulations is to use Graphics Processing Units (GPUs), which offer high computational performance.

CUDA is a programming model for GPUs, which extends the syntax of existing languages like Fortran and C/C++. While CUDA can maximize the performance of GPUs, its application to legacy codes requires a fundamental rewrite of the source code, leading to large development costs. This is a major bottleneck, particularly as many scientific simulation codes are developed and maintained by individual researchers.

To address this challenge, directive-based approaches such as OpenMP and OpenACC have been developed. These methods allow compilers to automatically generate GPU-accelerated code by simply inserting directives into loop structures. This minimizes the need for code modifications, thereby drastically reducing the development costs associated with GPU porting. However, there are a limited number of comprehensive studies that apply OpenMP and OpenACC to large-scale, complex, production-level applications and evaluate their performance, efficiency, and portability.

This study focuses on GAIA (Ground-to-topside model of Atmosphere and Ionosphere for Aeronomy), a simulation model in practical use. In this study, we report on our work to implement GPU acceleration for the General Circulation Model (GCM) component of GAIA, which simulates the neutral atmosphere. We developed two separate GPU-enabled versions using OpenACC and OpenMP. By comparing their performance with the CPU implementation, we evaluated the effectiveness and identify the challenges of both directive-based approaches. The results demonstrate that both GPU implementations achieved a significant speedup over the baseline code. In addition, the OpenACC implementation exhibited higher performance compared to the OpenMP implementation.

数値シミュレーションは高解像度化に伴うグリッド数の増大やモデルの改善に伴う計算量の増加により、計算時間が増大する。計算時間を早くする方法の一つとして、高い計算能力を持つ GPU (Graphics Processing Unit) の活用が進んでいる。CUDA は GPU 利用のための主要な手法であり、Fortran や C/C++ の既存の文法を拡張する。この手法は GPU の高い性能を引き出すことができる一方で、既存のアプリへの適用には、コードの抜本的な書き換えが必要となり、大きな開発コストを要する。特に、シミュレーションコードの多くは研究者がそれぞれに開発しているため、この開発コストが大きな課題となる。

この課題の解決のため、OpenMP や OpenACC を始めとするディレクティブベースの手法が開発された。本手法は、既存コードのループ構造等に指示行（ディレクティブ）を追加するだけでコンパイラが GPU 向けのコードを生成する。そのため、既存コードに対する大幅な改変が不要であり、GPU を利用するための開発コストを大幅に抑制できる。しかし、実運用されている大規模かつ複雑なアプリケーション（アプリ）を OpenMP と OpenACC により GPU 実装し、性能や実行効率、移植性などの評価を行った研究事例は多くない。そこで本研究では、実運用されているアプリである全大気統合シミュレーションモデル GAIA を対象として実験を行う。

本研究では、GAIA のうち中性大気領域を取り扱う大気大循環モデル GCM の部分に GPU 実装を OpenACC と OpenMP を用いてそれぞれ行なった。その後、これらの実装と CPU で実行する元コードと比較することによって、ディレクティブベースである両手法の有効性と課題を評価した。その結果、両方の GPU 実装において CPU 実装時より高い計算性能の実現が確認できた。また、GPU 実装手法としては、OpenACC が OpenMP に対して性能が高いことも確認できた。