

ポストペタスケールシステム向けの並列計算モデルの開発と評価

課題責任者

上原 均 海洋研究開発機構 地球情報基盤センター

著者

上原 均^{*1}, 横川三津夫^{*2}, 村井 均^{*3}, 板倉 憲一^{*1}, 浅野 俊幸^{*1}

*1 海洋研究開発機構 地球情報基盤センター

*2 神戸大学 大学院システム情報学研究科

*3 理化学研究所 計算科学研究機構

3～5年後と予測されるポストペタスケール級計算機の時代では100万並列にまで対応可能な高並列プログラムが必要となるが、そのような高並列プログラムの開発は困難である事も指摘されている。特に地球科学分野のプログラムは時に数十万行以上に及ぶことから事前の移行検討が不可欠である。それらをふまえると、既存の計算機環境でも利用でき、かつポストペタ時代にも通用する並列プログラム開発手段があれば極めて有効といえる。そこで我々は、ポストペタ時代において有望と目されるPGASモデルに着目し、それに基づいたXcalableMP言語を、多くの地球科学分野のプログラムが稼働する地球シミュレータ上で評価した。本研究では、姫野ベンチマークで基本的な性能評価を行った。この評価においてXcalableMPコンパイラでの内部処理に起因する性能低下を確認し、その改善手法の提案と効果を確認した。また実用的な評価としてNICAM-DCの演算カーネル作成を行い、そのXcalableMP言語への書換え、および評価を試みた。

キーワード：ポストペタ, 並列プログラミング言語, PGAS, 性能評価

1. はじめに

近年、計算機ユーザからの計算性能向上に対する強い要望に応じて高性能計算機の性能は著しく向上しており、計算機システムの総Core数は数万から数十万に達している。さらに3～5年後に登場すると予測されるポストペタスケール級計算機の総Core数は100万に達する事が見込まれる。一方で、そのような高並列計算機を活用するために不可欠な高並列プログラムの開発は、開発に要する人的コストの高騰や技術的難易度の上昇、開発期間の長期化などから、現在の並列プログラム開発よりも遥かに難しくなる事が専門家らによって予測されている。特に地球科学分野のプログラムは、時にその行数が数十万行以上に及ぶことから、コストが更に膨大なものになることが推測できるため、早期の移行対策が重要といえる。この早期の移行対策には既存の計算機環境でも利用でき、かつポストペタスケール時代に通用する並列プログラム開発手段があれば極めて有効である。

高並列プログラム開発に資する次世代並列モデルの研究は世界的に進められており、なかでも区分化大域アドレス空間 (Partitioned Global Address Space、以下PGASと略記) [1][2] モデルは普及が見込まれており、Fortran2008の言語規格にも導入され、ECMWFなどの地球科学の研究機関でも検討が進められている。

それらの背景をふまえて本研究では、ポストペタスケール級計算機における高並列プログラム開発に資するための並列計算モデル (並列プログラミング言語) として、理化学研究所が開発しているPGAS言語XcalableMP[3]に注目し、地球シミュレータへの移植および性能評価・分析

などを通じて、その利用可能性を検討する。XcalableMP言語は既存のソフトウェア資産からの移行容易性・可読性の高さなどが国際的にも評価されており[4]、QCDシミュレーションなどでの評価も進められているが、一方で地球シミュレータのようなベクトル型計算機での実績や地球科学分野での評価事例は乏しい。

なお本研究は海洋研究開発機構と神戸大学、理化学研究所による共同研究「ポストペタスケールシステム向けの並列計算モデルの開発と評価」の一環として実施した。

2. 研究計画

本研究は以下の二つの段階に大別して実施した。

1. XcalableMPコンパイラの地球シミュレータへの移植
2. 移植したコンパイラで得られる実行ファイルの性能測定・分析
 - A) 姫野ベンチマークでの基本的性能評価 (Fortran+MPI版[5]とXcalableMP版[6]の比較)
 - B) 実アプリケーションへの適用を見据えた、実用的性能評価としてのNICAM-DC[7]カーネルでの性能評価

3. 今年度の研究成果

3.1 XcalableMPコンパイラの移植

XcalableMP言語の言語仕様は策定済みであるが、コンパイラの実装には日々改良が加えられている。本研究の実施当初は開発版 (Nightly build ver.) を用いたが、最終的には安定版 (Stable ver. 0.9.2) を用いた。安定版は地球シミュレータで用いられているOS (SUPER-UX) に対応してい

るため、そのインストールは特に問題なく終了した。

3.2 性能評価

基本的性能評価として、地球シミュレータに移植した XcalableMP コンパイラでコンパイルした姫野ベンチマーク XcalableMP 版と、地球シミュレータの Fortran90 コンパイラ (sxmpif90) でコンパイルしたオリジナルの姫野ベンチマーク (Fortran+MPI 版) の実行性能を比較した。問題サイズは XL (1024 × 512 × 512) とし、繰り返し回数を 2000 回に固定した。また XcalableMP 版と Fortran+MPI 版のメモリレイアウトを共通にするために袖領域の調整を行った。この条件下での Fortran+MPI 版と XcalableMP 版によるホットスポットであるサブルーチン jacobi の性能評価結果を図 1 に示す。縦軸が実行性能 (GFLOPS)、横軸がノード数である。各 CPU が 4core を有しており、core あたり 1MPI プロセスを割り当てているため、最高で 256 並列までの結果となる。

32 ノード (128 並列) までの性能面での差は大きくないが、64 ノード (256 並列) では XcalableMP 版の性能と

Fortran+MPI 版の性能は明らかに乖離しており、後者が優位である。64 ノード時のピーク性能比は Fortran+MPI 版が約 22.0%、XcalableMP 版が約 17.4% であった。

ここで XcalableMP コンパイラで処理されたコードとその性能を精査したところ、上記の jacobi の最も高コストなループ部分を内部副プログラムに変形していることが判明した。図 2 に内部副プログラム化のイメージを示す。

この内部副プログラム化では、一部のローカル変数 (ここでは s0 と ss) を親プログラム内で宣言した上で、内部副プログラム内のループで用いている。これにより、当該変数は親子結合されていることとなり、グローバル変数扱いとなるため、ベクトル演算による終値をレジスタに保存する命令が発行される。この命令により、後続の命令が追い越せないことが判明した。SX-ACE では効率の良い演算を行うために命令の追い越しが必要となるため、この変数の親子結合が性能低下の原因と推定される。対策としては、当該変数を内部副プログラム内で宣言することで後続命令の追越しを有効にすることができる (図 2 右参照)。

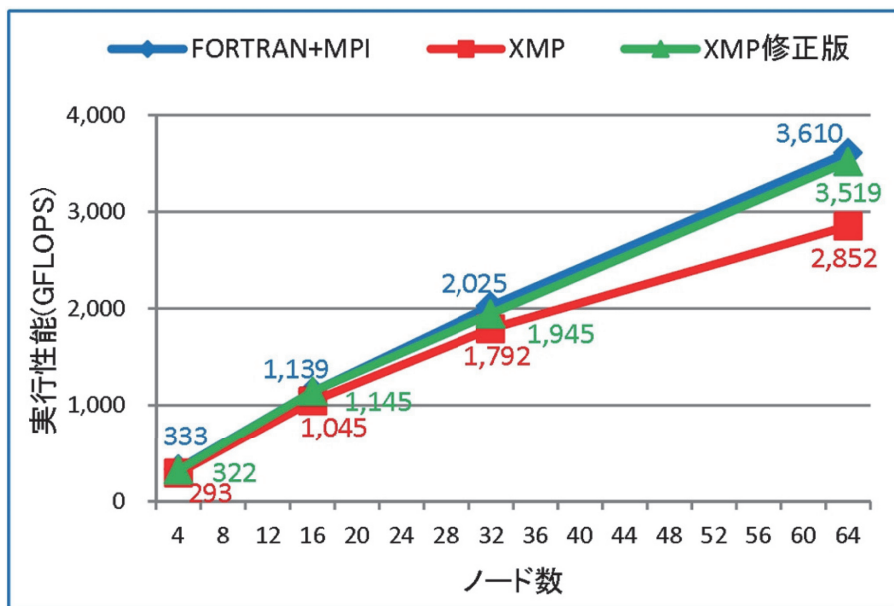


図 1 姫野ベンチマークにおける性能評価。

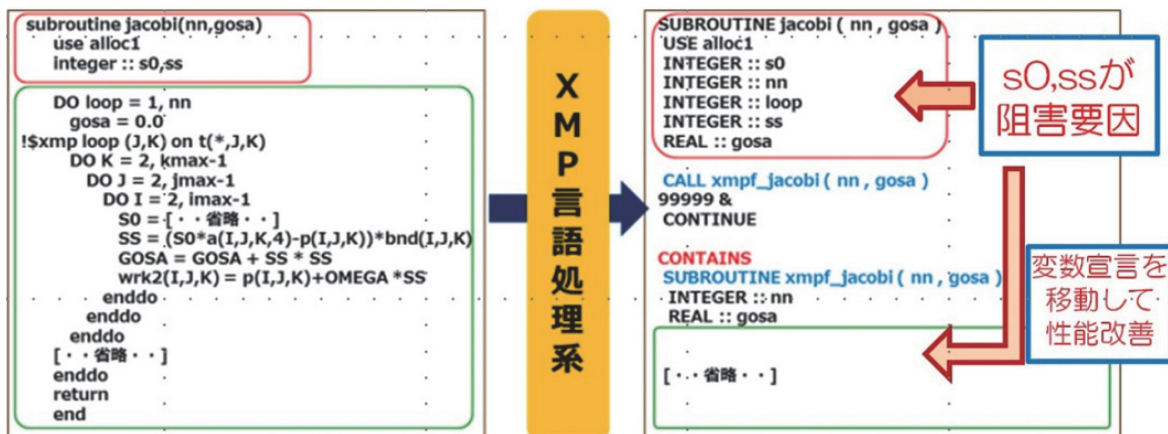


図 2 コンパイラによる内部副プログラム化とその改良。

そこで、当該変数を内部副プログラム内で宣言した場合の性能も計測した。図 1 に XMP 修正版として示すが、そのピーク性能比は 64 ノード時で約 21.5% であった。これにより XcalableMP 版でも内部副プログラム化手法を改良する事で Fortran+MPI 版とほぼ同等の演算性能が出せる事が確認できた。なお当初懸念していたベクトル処理と XcalableMP コンパイラ内部処理の齟齬による実行性能低下は確認されなかった。

次に実用的性能評価の事前調査として、オリジナルの NICAM-DC を幾つかのケースで性能評価したところ、全ケースで特定のサブルーチンがコスト上位に来た。そこで実用的性能評価では、このサブルーチンをカーネル化して性能評価する事とした。現在、カーネル化および XcalableMP 言語への書き換えを進めている。

4. おわりに

本研究では PGAS 言語 XcalableMP の地球シミュレータでの評価を行った。姫野ベンチマークによる基本的性能評価で XcalableMP 版は Fortran+MPI 版の約 8 割の性能であるが、XcalableMP コンパイラ内部で行われる内部副プログラム化方法の改良により XcalableMP 版でも Fortran+MPI 版とほぼ同じ演算性能を出せる事が確認できた。実用的性能評価としては NICAM-DC の演算カーネルについて XcalableMP 言語への書換えに取り組んだ。今後は、実用的性能評価の継続および改善された XcalableMP コンパイラの再評価、ひいては実アプリケーションそのものの実装・評価を行いたい。

謝辞

本研究の一部は文部科学省フラッグシップ 2020 プロジェクト（ポスト「京」の開発）「ポスト「京」で重点的に取り組むべき社会的・科学的課題」における重点課題④「観測ビッグデータを活用した気象と地球環境予測の高度化」（課題責任者：海洋研究開発機構 高橋桂子）の一環として行われた。また本研究の実施においては、NEC ソリューションイノベータ株式会社 山口健太氏ほか、日本電気株式会社関係各位に様々なご協力をいただいたことに感謝し、ここに記す。

文献

- [1] Bill Carlson et al., "Programming in the Partitioned Global Address Space Model", http://upc.gwu.edu/tutorials/tutorials_sc2003.pdf
- [2] Partitioned Global Address Space, <http://www.pgas.org/>
- [3] XcalableMP, <http://www.xcalablemp.org/>
- [4] Masahiro Nakao et al., "XcalableMP for Productivity and Performance in HPC Challenge Award Competition Class 2", HPC Class2 Submission, Denver, Colorado, USA, Nov. 2013.
- [5] 姫野ベンチマーク, <http://accr.riken.jp/supercom/himenobmt/download/mpi-vpp/>
- [6] 姫野ベンチマーク XcalableMP 版, <https://github.com/omni-compiler/XMP-Benchmark>
- [7] NICAM-DC, <http://scale.aics.riken.jp/nicamdc/index.html>

Study of Parallel Computational Model for Post-Peta Scale Computer System

Project Representative

Hitoshi Uehara Center for Earth Information Science and Technology, Japan Agency for Marine-Earth Science and Technology

Authors

Hitoshi Uehara^{*1}, Mitsuo Yokokawa^{*2}, Hitoshi Murai^{*3}, Ken'ichi Itakura^{*1} and Toshiyuki Asano^{*1}

*1 Center for Earth Information Science and Technology, Japan Agency for Marine-Earth Science and Technology

*2 Graduate School of System Informatics, Kobe University

*3 Advanced Institute for Computational Science, RIKEN

Experts have inferred that parallel programming language based on the partitioned global address space (PGAS) model can be a promising prospect for the development of large scale parallel programs that are required for hundreds of petaFLOPS class (post-Peta scale) computers. The purpose of this study is to evaluate the applicability of XcalableMP, which is a type of PGAS based language. In the study, we measured the performance of HIMENO benchmark coded in both Fortran90 and XcalableMP on the Earth Simulator. Based on the performance evaluation results, we propose a novel solution to improve the performance of XcalableMP code and demonstrate its effectiveness.

Keywords: Post Peta scale, parallel programming language, PGAS model, benchmark

The partitioned global address space (PGAS) model [1] is a hopeful parallel computational model for the forthcoming era of hundreds of petaFLOPS class (post Peta scale) computers. In particular, XcalableMP [2], which is a PGAS-based language, has two features: code readability and ease of rewriting from legacy Fortran/C code. Thus, XcalableMP is promising for the development of large-scale programs. The purpose of this study is to evaluate the applicability of XcalableMP on the Earth Simulator.

First, we measured the performances of HIMENO benchmarks coded in Fortran90 [3] and XcalableMP [4]. We rewrote each source code in a small number of lines to adjust the memory access condition and other conditions; we utilized XcalableMP compiler stable version 0.9.2 for compiling.

In HIMENO benchmark, the nested loop part of Jacobi's method incurs the highest computational cost. Figure 1 illustrates the measurement results of the part.

The effective performances using 64 nodes (256 MPI processes) are 22.0% and 17.4% for Fortran90 and XcalableMP code, respectively.

We have clarified that the performance degradation in case of XcalableMP is attributable to the code translation by the

XcalableMP compiler. We, therefore, proposed the solution and confirmed its effectiveness. The improved XcalableMP code achieved an effective performance of 21.5%. The performance of improved XcalableMP code is represented by "XMP rev." in Fig.1.

We have also attempted to write the computation kernel code of NICAM DC [5] and rewrite it in XcalableMP. As part of our future work, we will attempt to evaluate the performance of the computational kernel. We will also attempt to code the communication kernel in XcalableMP and evaluate its performance.

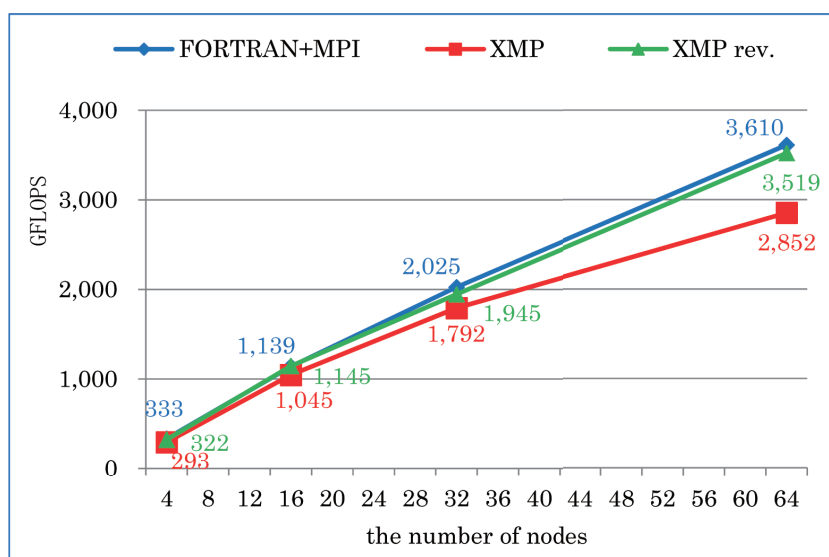


Fig. 1 HIMENO benchmark results for nested loop part of Jacobi's method

Acknowledgement

A part of this study was conducted as part of post-K priority issue 4 “Advancement of meteorological and global environmental predictions utilizing observational Big Data.” This work was supported by Mr. Kenta YAMAGUCHI and other NEC staffs, who are invaluable to the completion of this study. We are deeply grateful to them.

References

- [1] Partitioned Global Address Space, <http://www.pgas.org/>
- [2] XcalableMP, <http://www.xcalablemp.org/>
- [3] Himeno Benchmark, <http://accr.riken.jp/supercom/himenobmt/download/mpi-vpp/>
- [4] Himeno Benchmark ver. XcalableMP, <https://github.com/omni-compiler/XMP-Benchmark>
- [5] NICAM-DC, <http://scale.aics.riken.jp/nicamdc/index.html>

