

受付 ID	17a34
分野	HPCS

アクセラレータクラスタを対象とした並列言語 XcalableACC と  
メニーコアクラスタを対象とした並列言語 XcalableMP の性能評価

Evaluations of a parallel language XcalableACC for accelerated cluster  
systems and a parallel language XcalableMP for many-core cluster systems

中尾昌広

理化学研究所 計算科学研究センター

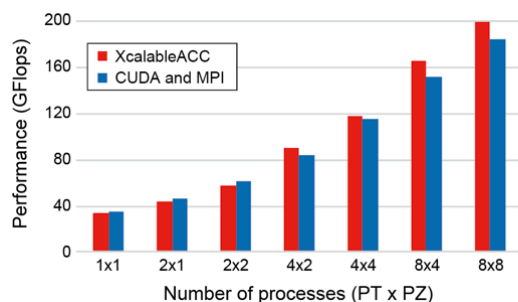
## 1 研究目的

本研究の目的は、「アクセラレータクラスタを対象としたアプリケーションを簡易に開発できる言語 XcalableACC (XACC)」および「メニーコアクラスタを対象としたアプリケーションを簡易に開発できる言語 XcalableMP (XMP)」の評価を行うことである、これらの評価は、性能と生産性の2つの軸から行う。

## 2 研究成果の内容

### 2.1. アクセラレータクラスタ用並列言語 XACC

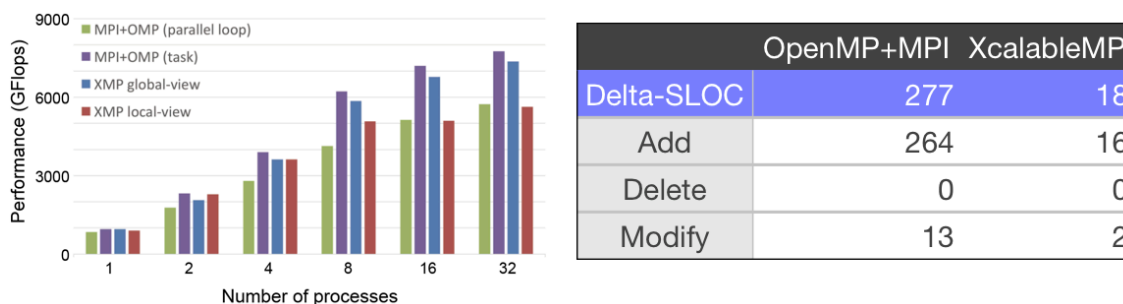
まず、HA-PACS/TCA が搭載している Tightly Coupled Accelerators (TCA) と InfiniBand を組合せたハイブリッド通信機構を開発した。その結果、単体の通信性能よりもハイブリッド通信の性能は高いことを確かめた。次に、XACC からそのハイブリッド通信を利用できる記法および処理系を開発した。格子 QCD アプリケーションを用いて性能評価を行った結果、既存のプログラミングモデルである CUDA+MPI よりもハイブリッド通信を利用した XACC は高い性能を発揮することがわかった。また、生産性の評価として、その格子 QCD アプリケーションについて、逐次コードから並列コードに変更するための行数 (Delta-SLOC) をカウントした。その結果、XACC は CUDA+MPI よりも少ない行数でアプリケーションを開発できることがわかった。



	CUDA+MPI	XcalableACC
Delta-SLOC	767	86
Add	348	80
Delete	73	0
Modify	346	6

## 2.2. メニーコアクラスタ用並列言語 XMP

メニーコアクラスタにおいて、性能と生産性を両立させるために、XMP に対して動的タスク並列機能の追加を行った。また、その機能を高い性能で実行できる XMP 処理系の開発を行った。複雑な依存関係を持つブロックレスキーベンチマークを用いて、XMP の性能評価を行った結果、既存のプログラミングモデルである OpenMP+MPI と遜色ない性能を発揮することがわかった。また、前節と同様に Delta-SLOC をカウントした結果、XMP は OpenMP+MPI よりも少ない行数でアプリケーションを開発できることがわかった。



	OpenMP+MPI	XcalableMP
Delta-SLOC	277	18
Add	264	16
Delete	0	0
Modify	13	2

## 3 学際共同利用として実施した意義

我々の研究はアクセラレータおよびメニーコアを搭載した大規模クラスタに対するプログラミング環境の向上を目的としている。そのため、Oakforest-PACS, COMA, HA-PACS/TCA といった世界トップレベルの大規模環境を必要とした。特に、XACC の TCA と InfiniBand を利用したハイブリッド通信の評価を行えるのは HA-PACS/TCA のみであるため、HA-PACS/TCA は必須であった。

## 4 今後の展望

XACC の処理系については、2018 年 3 月末にホームページに公開した (<http://omni-compiler.org>)。今後は、GPU 以外のアクセラレータを搭載したクラスタシステム (例えば、PESY-SC2 や Intel Xeon Phi など) への対応を考えている。XMP については、XMP の仕様を策定している PC クラスタコンソーシアム (<https://www.pcluster.org/>) にタスク並列の仕様を提案し、より幅広い意見を組み込みながら、新しいアプリケーションへの対応を行っていく予定である。

## 5 成果発表

5.1 学術論文：7 件 (国外)

5.2 学会発表：1 件 (国外)、6 件 (国内)

使用計算機	使用計算機 に○	配分リソース※	
		当初配分	追加配分
HA-PACS/TCA	○	1400	
COMA	○	3000	
Oakforest-PACS	○	5000	
※配分リソースについてはノード時間積をご記入ください。			