
発表概要

Java Grid Compiler

稲上克史[†] 平木 敬[†]

近年 PC の低価格化とネットワーク技術の進歩によって、ネットワークを介して複数の PC 上で分散計算を行うグリッドコンピューティングが普及してきている。しかし、グリッドプログラミングには、単一 SMP マシン上での並列プログラミングとはフレームワークが異なり、ネットワークを意識したプログラミングが必要になる。そこで我々は SMP 上で並列計算を行うプログラムを静的に分散環境で動作するプログラムに自動変換する手法を提案する。とくにターゲット言語として Java を選択した。具体的にはマルチスレッドを用いて記述された Java 並列プログラムに、自動的に適切なネットワークコミュニケーション処理を挿入し、複数ノード上で分散並列計算を行う Java プログラムに静的に変換する。とくにクラスタのようなノードやネットワーク構成が規則的な環境だけでなく、ノードアーキテクチャが多様で、動的に増減するようなヘテロジニアスな環境での動作を焦点とした。本研究によってプログラムはネットワークを意識せずに単一ノード上でのマルチスレッドプログラムを記述するだけで分散環境上でグリッド計算を行うプログラムを生成できる。また生成されるのも Java プログラムであるため、すでに普及している Java VM 上で動作し、動作に特別な環境を必要とせず、多様なプラットフォームで動作させることが可能となる。我々はこの手法を実装し、実際に並列ベンチマークプログラムを変換し、その生成されたプログラムを分散環境上動作させた。最後に実験で得られた性能や機能に関する制限などについて議論を行う。

Java Grid Compiler

KATSUSHI INAGAMI[†] and KEI HIRAKI[†]

Today, grid computing, which is computing with distributed resources through network, is a hot topic. But grid programming is very different from parallel programming on SMP single machine, because of communications between nodes. In this presentation, we suggest the methods to translate statically from parallel programs on SMP into the one working on distributed resource environment, by inserting network operations automatically. We select Java as a target language. The aim is to enable programmers to get grid program only with writing SMP parallel Java program network transparently, and enable existing Java multi-threaded programs work on distributed environment. We focus on not only the homogeneous environment such as cluster, but also heterogeneous environment consisting of various architecture nodes and member nodes may dynamically change. Our approach gives some benefits. Since the generated code is also Java program and can work on traditional Java VM widely used, need no change to runtime environment. In addition, generated code can work on multiplatform. We implement our method, translate parallel benchmark programs, and let the generated programs work on distributed environment. Finally, we examine the performance and scalability of generated programs, and discuss limitations of our method.

(平成 17 年 8 月 3 日発表)

[†] 東京大学大学院情報理工学系研究科

Department of Computer Science, The University of Tokyo