

## 発表概要

## 三次元表現空間を用いたプログラムの解析と並列化の一手法

三好健文<sup>†</sup> 杉野暢彦<sup>†</sup>

計算負荷の高いアプリケーションは、複数の計算ノードを用いる並列計算により実装されることが多い。しかし、効率の良い並列計算を行うためには、プログラム中に内在する並列度を抽出し、効果的にスケジューリングすることが必要となる。そのために、単一の計算ノードのために最適されて設計されたプログラムを、並列度の高いプログラムとしてアルゴリズムレベルで再設計することが必要となることも多い。これは、ソフトウェア開発者によって大変困難であるため、コンパイラによる自動化が望まれている。アルゴリズムレベルでのプログラムの最適化は、基本ブロックを単位とした解析だけではなく、演算レベルでの解析、すなわちプログラム中の静的な演算からなる要素と動的な処理を含む要素を統一的に評価し、定量的な評価を行う必要がある。本発表では、三次元表現空間によるプログラムの包括的な解析手法と、これを用いたプログラムの並列化の一手法を示す。まず、プログラム中で用いる計算資源、演算処理に要する時間、動的な演算の存在条件の三次元からなる表現空間を定義する。この表現空間上においてプログラム中の動的な要素と静的な要素を包括的に評価でき、さらにプログラムに対する様々な最適化が空間上の操作によって表すことができることを示す。次に、単一計算ノードを対象として表現空間と空間上の操作による既存の最適化方法を示し、その拡張・改善方法を提案する。さらに、表現空間と空間上の操作による複数の計算ノードに対する並列化の一手法を示し、その効果を示す。

## Techniques for Analysis and Parallelization of Program by Use of 3-D Representation Space

TAKEFUMI MIYOSHI<sup>†</sup> and NOBUHIKO SUGINO<sup>†</sup>

A novel automatic parallelizing method for multiple processing units based on unified representation scheme is proposed. It is common to implement them in parallel computation manner by use of multiple processing units or multiple computational nodes for computationally intensive applications. For highly efficient computation on multiple processing units, software designers redesign algorithm from the original algorithm designed for single processing unit, and this causes heavy programming load. Therefore, a compiler with automatic parallelization has been strongly desired. The conventional methods analyze and parallelize a given program in basic blocks. In order to achieve more efficient computation, global analysis of static and dynamic elements of a given program, and inter-block parallelization are necessary. In this presentation a technique of analyzing and paralleling of programs by use of three dimensional representation space and scheme. Firstly, a given program is uniquely mapped onto three dimensional representation space, which consists on time, computational resource and mode axes, where mode denotes alternative execution associated with conditional branches. By use of this representation, a give program can be analyzed in terms of various measures. Furthermore, various optimization techniques are represented by operations to the representation. In this presentation, single computational node is assumed, and optimization techniques is presented. Automatic parallelization methods for multiple computational nodes follows, where overhead in inter-node communication is accurately estimated.

(平成 17 年 8 月 3 日発表)

<sup>†</sup> 東京工業大学総合理工学研究科物理情報システム専攻  
Department of Information Processing, Tokyo Institute  
of Technology