

発表概要

タスク並列プログラムに対する動的カットオフ手法の評価

岩崎 慎太郎^{1,a)} 田浦 健次朗^{1,b)}

2016年8月10日発表

タスク並列プログラムにおいて、タスクの粒度の調整は最も重要な最適化の1つである。タスクの粒度を大きくして並列化オーバーヘッドを削減するために、“適当な基準”を満たしている場合、タスク生成を単なる関数呼び出しに置き換える、カットオフと呼ばれる最適化手法が一般的に用いられる。このカットオフの“適当な基準”は、従来はプログラマが実験的・経験的に決めていたが、プログラマの負担を減らすために、タスクの総量やタスクの深さなど、タスク並列ランタイムの持つ情報を用いてカットオフ最適化を自動化する動的カットオフ手法が複数提案されている。本研究は、従来提案されてきた動的カットオフ手法を統一的に評価することで、その長短を明らかにする。まず同一のタスク並列処理系に対してそれぞれの動的カットオフ手法を実装する。その後、ベンチマークを用いてその性能向上を測定することで、各手法の有効性を明らかにする。得られた結果から望ましい動的カットオフ手法についての考察を行う。

Evaluation of Dynamic Cut-off Techniques for Task Parallel Programs

SHINTARO IWASAKI^{1,a)} KENJIRO TAURA^{1,b)}

Presented: August 10, 2016

Controlling task granularity is one of the most important optimizations for task parallel programs; a task parallel program often suffers from a large parallelization overhead due to its extremely fine granularity even though task parallel runtime systems are designed to handle tasks efficiently. A “cut-off” is a well-known method to address this problem; it alleviates a tasking overhead by just calling serial functions tasks instead of creating tasks in a certain condition. The cut-off conditions are often specified by programmers, which decreases productivity. Numerous studies have focused on a dynamic cut-off; a technique to determine whether the task is created or sequentialized by utilizing runtime information such as total amount of tasks and current depth of a task. In this work, we evaluate the dynamic cut-off techniques studied in the previous work and characterize their performance optimization. We implemented these cut-off methods on the same task parallel runtime system, and measured the performance using the same benchmarks. In summary, we discuss the desirable dynamic cut-off strategy based on the experimental results.

¹ 東京大学情報理工学系研究科
Graduate School of Information Science and Technology,
The University of Tokyo, Bunkyo, Tokyo 113-0033, Japan

a) iwasaki@eidos.ic.i.u-tokyo.ac.jp

b) tau@eidos.ic.i.u-tokyo.ac.jp