

発表概要

入れ子関数を利用した動的負荷分散

八 杉 昌 宏^{†,††} 田 畑 悠 介[†]
 小 宮 常 康[†] 湯 淺 太 一[†]

探索問題などの樹状再帰的な細粒度並列計算においてプロセッサ間でタスクを授受して効率良く負荷分散を行うには、実行中のタスクから *lazy* に並列実行可能なタスクを分割・生成し抜き取れるようにした遅延タスク生成などの負荷分散方式が有効である。転送可能なひとまとめの仕事 (work) であるタスクの抜き出しはできるだけ呼び出しの根元から行うことで、ほぼ最小限のタスクの生成・授受で各プロセッサに仕事が十分に割り当てられる。しかし、そのためには関数の呼び出し中に通常はアクセスできない呼び出し元の変数にアクセスできる必要がある。本発表では、GNU C コンパイラなどが C 言語の拡張機能として提供する入れ子関数を利用して、呼び出し元の変数にアクセスすることで、遅延タスク生成に基づく負荷分散を行うプログラムが記述できることを示す。また、バックトラックに相当する動作の記述も可能であることを示す。ただし本記述方式では、オリジナルの遅延タスク生成とは異なり、タスクは、継続を処理するものではなく、関数中に示された並列実行可能部分を処理するものとする。本発表では、共有メモリ型並列計算機上および SMP クラスタ上で、提案する記述方式のプログラムを実行した結果についても述べる。

Dynamic Load Balancing by Using Nested Functions

MASAHIRO YASUGI,^{†,††} YUSUKE TABATA,[†] TSUNEYASU KOMIYA[†]
 and TAIICHI YUASA[†]

To realize efficient dynamic load balancing by transferring tasks among processors in tree-recursive fine-grained parallel computing such as searching problems, load balancing schemes which lazily create and extract a task by dividing the present running task, such as *Lazy Task Creation* (LTC), are effective. By extracting a task, which is a piece of work transferable to other processors, around the *root* of the invocation tree as much as possible, each processor can have sufficient amount of work with the almost minimum number of task transfers. However, it requires a way to access to caller's variables which are usually not accessible during an invocation of a function. In this presentation, we show that we can describe a program with LTC-based load balancing where caller's variables are accessed by using *nested functions* provided as an extension to C by the GNU C compiler. We also show that we can describe a behavior corresponding to *backtracking*. Although a task is created to process a *continuation* in the original LTC, in our scheme, a task is created to process the part for parallel execution specified in a function. In this presentation, we also show the result of the execution of the proposed-style program on a shared-memory parallel computer and on an SMP cluster.

(平成 14 年 1 月 30 日発表)

[†] 京都大学大学院情報学研究科通信情報システム専攻
 Department of Communications and Computer Engineering, Graduate School of Informatics, Kyoto University

^{††} 科学技術振興事業団さきがけ研究 21「情報と知」領域グループ
 "Information and Human Activity", PRESTO, Japan Science and Technology Corporation (JST)