

## プログラム解析情報可視化システム PREPS のタスクレベル並列化支援機能\*

3L-1

草野和寛 蒲池恒彦 †藤田淳 †渡辺幸光†

NEC C&amp;C 研究所† †NEC 情報システムズ‡

e-mail: {kusano,kamachi}@csl.cl.nec.co.jp, {f-jun,yukimi}@ats.nis.nec.co.jp

## 1 はじめに

現在我々はプログラム解析情報可視化システム PREPS の研究開発を行なっている [1]。PREPS は FORTRAN に HPF(High Performance Fortran) のデータ分割指示行を加えたプログラムの依存関係やデータ分割などの解析情報を、GUI を用いて利用者に提供する可視化システムである。本システムは、データ並列を利用した並列プログラムを開発する作業を支援する機能を備えている。

ここで、HPF のようなデータ並列に対して、ユーザが指定した部分並列実行の単位とするタスク並列化方式が存在する。タスク並列化は、プログラム中に含まれる比較的処理が独立した機能単位に並列処理を行なう。Fortran M[2] では、ユーザがタスク並列実行を記述するための言語インタフェースと実行時ライブラリの研究が行なわれている。また、データ並列性を記述する HPF にタスク並列を記述する仕様を加えることを、HPFF(HPF Forum) でも検討している [3]。

超並列計算機や高速ネットワークで接続された分散環境の登場により、利用可能なプロセッサが飛躍的に増大している。これを利用するためには、プログラム中存在している並列性をできるだけ利用する必要がある。タスク並列をデータ並列と組み合わせることにより、プログラム中に存在する並列性をさらに引き出すことが可能になる。そこで、本システムにタスク並列化作業を支援する機能を加える検討を行なっている。本稿では、タスク並列化を支援する PREPS の機能とその概要を述べる。

## 2 タスク並列化

## 2.1 タスク並列化支援機能

逐次プログラムをタスク並列化する作業では、まずプログラムで並列実行可能なタスクを抽出し、そのタスク間に依存関係がないことを保証する。そしてプログラムでタスクの指定とタスク間で通信が必要な通信コードの挿入を行なう。この作業手順において、タスク分割を指定する部分を除き、言語処理系がプログラムを解析して行なうことが可能である。しかし、並列化可能性の判定

には大域的解析情報が必要であるなどの理由により、タスク間の依存関係はユーザが保証することを要求している研究が多い。

本システムでは、FORTRAN(+HPF 指示行) プログラムを対象に上記のようなタスク並列化を行なう作業を支援するため、以下にあげる機能を提供する。本システムの出力はタスク並列記述を挿入した並列プログラムである。

- 対話環境でのタスク指定
- タスク単位の解析情報生成
- 並列実行可能性判定
- 静的解析情報に基づく性能予測

これらの機能を用いることで、ユーザが指定したタスク間の並列化の可能性判定や並列化できない原因の特定作業を容易に行なうことができる。さらに、静的性能予測機能を用いることで、並列化の効果をプログラムの実行をしないで見積もることが可能になる。

本システムを用いたタスク並列化では、ユーザはタスクの指定のみを行ない、並列化可能性の判定やデータ通信解析などの作業は言語処理系が行なう。これに対して、HPFF の仕様や Fortran M では、タスクが並列実行可能であることをユーザが判定して指定する必要がある。さらに、Fortran M ではタスク間で必要なデータ通信をユーザが記述する必要がある。

## 2.2 タスクとその指定方法

本システムにおいてタスクを以下のように定義する。  
タスク 基本ブロック 1 つ以上からなるプログラム片  
タスクの例として、DO ループや if-else-endif ブロックがあげられる。

タスクは本システムの対話機能のインタフェースを利用してユーザが指定する。この指定には、基本ブロック (or タスク) をノードとし、コントロールフローをエッジとした有向グラフを利用する。このグラフでタスクの指定は、タスクを構成する基本ブロックをユーザが指定することにより行なうことができる。ここで、本システムではタスクの表示とタスク間の依存解析を容易にするために、タスクの入口/出口は 1 つに制限している。ユーザが指定した基本ブロックがこの条件を満たしていない場合には、その基本ブロックの集合をタスクに指定することはできない。

\*Program information visualization system PREPS—the task-level parallelization assist function

†Kazuhiro KUSANO, Tsunehiko KAMACHI, Jun FUJITA, Yukimitsu WATANABE

‡C&C Research Laboratories, NEC Corporation

§NEC Informatec Systems

## 2.3 並列化可能性の表示

ユーザ指定のタスクが並列化可能であることを判定する機能を提供する。また、並列化できない場合には、並列化を阻害している原因の表示を行なう。並列化できない原因は、タスク間に依存関係があると判定された変数が存在する場合である。この判定において、タスクが手続き呼びだしを含む場合には、プログラムの大域的解析情報が必要となる。そこで、本システムは手続き間解析を行なう機能を持っている。並列化可能性や並列化を阻害している原因の表示は、ユーザのタスク指定操作に対応して再解析して表示を更新する。

タスク並列性情報の表示例を図1に示す。図1(a)のA,B,C

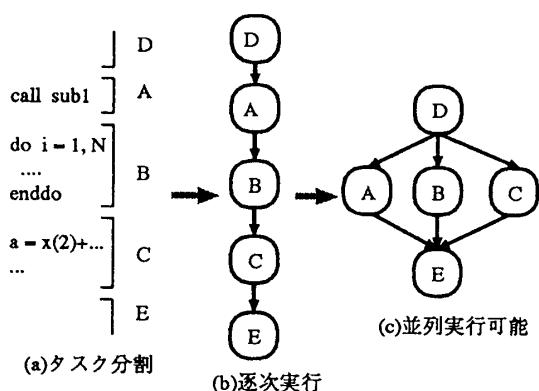


図1: タスク並列性表示

に示すタスクが並列実行可能である場合には図1(c)のように表示する。これらのタスクが並列化できない場合には図1(b)のような表示となる。この時に、並列化を阻害している依存関係が存在するエッジを選択することで、依存関係の詳細な情報を提供する。この機能を用いることで、タスク並列性を視覚的情報として直観的に理解することができ、さらにタスク間の依存関係の有無を容易に確認することができる。

## 2.4 静的性能予測機能

本システムでは、プログラムの静的な解析情報と逐次実行の実行結果を利用して、並列化したプログラムの性能を予測する機能を提供する。この機能では、図2に示すように Critical path の検出と並列プログラムの予測性能表示を行なう。なお、この予測性能の計算では、ネットワークのコンテンションによる遅延やキャッシュの影響など実行時の要因については考慮しない。また、プロセッサ間の転送性能はパラメータとしてユーザが与えるものとする。

本機能は、各タスクの実行時間とタスク間のデータ転送に必要な時間に分けて実行時間の予測を行なう。並列実行部分の処理時間は逐次実行の時間と並列処理度数よ

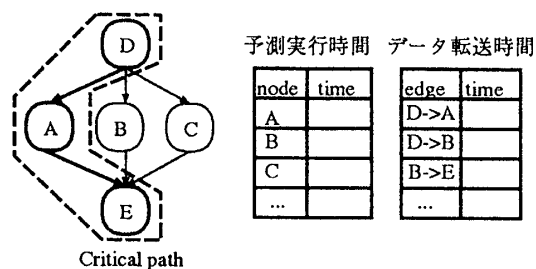


図2: 性能予測結果の表示

り計算する。この時に、並列実行タスク内で必要なデータ転送時間は、データ分割情報と計算に必要なデータから転送が必要なデータ量から求める。

次に、タスク間のデータ転送に要する時間は、タスクの計算に必要なデータは全てデータ転送が発生するものとして転送量の見積もりを行なう。つまり、データ転送は最悪のケースを想定して予測を行なっている。これは、タスクとデータのプロセッサへのマッピングによりデータ転送に必要なデータは変化するためである。そこで、本システムでは不要なデータ転送を削除するための対話機能を提供する予定である。この機能では、タスク間でデータ転送が必要と判定した変数の表示を行ない、これに対してユーザが各変数単位にデータ転送の削除を行なうことができる。本機能を用いることで、並列プログラムの実行をしないで並列化の効果を見ることが可能になるので、並列プログラムの開発サイクルを短縮することができる。

## 3 まとめ

以上、プログラム解析情報可視化システム PREPS のタスク並列化支援機能とその概要を述べた。現在、本稿に述べた機能の基本設計を終了しており、今後は各機能の詳細設計およびインプリメントを行なう。そして、本機能がタスク並列化作業において有効であることの確認を行なう予定である。

なお本研究は科学技術振興調整費総合研究「知的生産活動における創造性支援に関する基盤的研究」の一環として行なった。

## 参考文献

- [1] 草野他, プログラム解析情報可視化システム -PREPS-, 情処第49 全大予稿集 4U-05, 1994.
- [2] I. Foster, etc., Programming in Fortran M, Technical Report, ANL, 1993.
- [3] High Performance Fortran Forum, High Performance Fortran Journal of Development, CRPC-TR93300, Rice University, May, 1993.