

## 平成24年度 情報工学コース卒業研究報告要旨

枝廣 研究室	氏 名	山 田 竜 正
卒業研究題目	ソースコードからのSSAグラフを用いた静的な命令レベルの並列度解析	
<p>組込みシステム開発は大規模複雑化、制御の高度化によりモデルベース開発がデファクトスタンダードとなっている。また複雑化や高度化によりシングルプロセッサにおける演算負荷が限界に近づくに伴い、今後はマルチ・メニーコアを活用していくことも考えられている。マルチ・メニーコアを活用するためにはソフトウェアを並列化する必要がある、モデルベース開発のモデルからの並列化が望まれている。ところが、制御設計者は最終的に実行される計算機プラットフォームまで考慮に入れない場合が多く、設計されたモデルが並列化に向かない場合も多い。そのため、モデルベース開発のモデルにおける並列性解析を行い、制御設計者自身や、設計指針にフィードバックすることが重要である。</p> <p>本研究では、組込み制御のモデルベース開発において自動生成されたソースコードを静的に解析し、その並列性を見積もり、効果的な並列化を行うための情報を抽出する手法を確立することを目的としている。そのために、我々は制御・データフローグラフ (CDFG: Control and Data Flow Graph) にコア割当情報を入れた、並列実行 SSA グラフを考えた。この並列実行 SSA グラフにより、コンパイラ等から抽出された CDFG 上で並列化を考え、シミュレーションし、並列性解析などを行うことができるようになった。また、並列性解析には静的単一代入 (SSA: Static Single Assignment) を利用した。SSA 形式とは、変数の使用に対してその変数の定義の場所は一つのみとした形式のことである。SSA 形式に変換することで変数間のデータ依存関係の見通しが良くなり並列性の抽出が容易となる。</p> <p>静的解析、SSA 形式の利用のため、本研究では COINS コンパイラインフラストラクチャを利用した。ソースコードを COINS コンパイラを使用して SSA グラフを生成し、そのグラフを解析することで元のソースコードの命令レベルの並列度を静的に見積もることにした。今回使用した COINS コンパイラとは新しいコンパイル方式を容易に実験・評価・開発するために開発されたオープンソースの共通インフラストラクチャである。COINS コンパイラは SSA 形式としたコードをデータや命令をノードとする木構造で表現した SSA グラフを生成・出力する機能を備えている。これに対して、元々の SSA グラフでは制御・データの依存関係を辿ることができない条件分岐の条件判定と分岐後の処理を連結させ、またメモリアクセスを行うデータ構造間を連結させ依存関係をできるだけ再現するようグラフを変更した。</p> <p>並列度の解析は、グラフのノードを幅優先で1サイクルずつ探索しながらノードに割り当てられたコアがノードの命令を実行していく形で行われる。それぞれサイクルでの命令実行中のコア数をそのサイクルでの並列度となる。コア数や命令の実行に必要なサイクル数は任意の数に変更できる。また、コア間通信のオーバーヘッドも考慮するようにしておりその通信コストも設定することで、解析するコードのプログラムを実行するマシンに合わせた解析を行うことができる。</p> <p>本研究で作成したプログラムの実行結果を評価するため、実際に MATLAB/simulink モデルから自動生成を行った C コードを解析の対象とした。通信コストを変更しながら対象を解析したところ、通信コストの増大に伴い並列度が急速に下がっていく結果となり、コア数の多さに比例した並列度を得るためには通信コストの削減が必要であることが確認できた。本研究での並列度解析はループや関数呼び出しの解析は行っていないため、それらの考慮については今後の課題となる。</p>		